

На правах рукописи

Хамухин Анатолий Владимирович

**Высокоэффективные алгоритмы семантической обработки
видеоизображений и управления приборными комплексами
технического зрения**

05.13.01 – «Системный анализ, управление и обработка информации»
(информационно-вычислительное обеспечение)

Том 1
ДИССЕРТАЦИЯ
на соискание учёной степени
доктора технических наук

Научный консультант,
доктор технических наук,
профессор Петрикович Я.Я.

Москва — 2016

Содержание

Том 1

Введение	7
1. Алгоритмы семантического сжатия цифровых видеосигналов	20
1.1. Алгоритмы семантического сжатия как основа современных комплексов технического зрения	20
1.2. Выбор языковой среды, компиляторов и систем визуальной программной разработки алгоритмов семантического сжатия	33
1.3. Классификация алгоритмов семантической обработки цифровых информационных видеосигналов	36
1.4. Алгоритмы распознавания и идентификации видеоизображений	48
1.5. Теоретическая оценка достоверности выходных данных алгоритмов сопряжения, синхронизации и семантической классификации в системах технического зрения	57
1.6. Анализ достоинств и недостатков традиционных подходов к обработке информационных видеосигналов	61
1.7. Результаты анализа и исследования современных методик оценки достоверности алгоритмической обработки видеинформации	68
1.8. Цели и задачи диссертационной работы	69
Выводы	71
2. Разработка и функциональный анализ параллельных алгоритмов семантической обработки видеосигналов и создание теоретических основ синхронизации и сопряжения изображений	73
2.1. Разработка и описание распределённых алгоритмов семантической обработки синхронных видеосигналов от разных датчиков	73
2.2. Теоретический анализ ограничений, вытекающих из пропускной способности сети и быстродействия устройств обработки	84
2.3. Основы теории алгоритмов сопряжения видеокамер с общей зоной обзора	89
2.4. Исследование особенностей преобразования алгоритмов переноса координат изображения с видеокамер на топографический план местности	94
2.5. Оценка точности преобразований координат объектов семантического анализа при сопряжении изображений	98
2.6. Разработка алгоритмов автоматизации настройки сопряжения неподвижных и поворотных видеокамер	100
Выводы	115

3. Исследование и разработка алгоритмов синхронизации, управления и семантической обработки информации в системах технического зрения	117
3.1. Разработка алгоритмов синхронизации программных модулей обработки цифровых информационных массивов	117
3.2. Теоретическая оценка быстродействия алгоритмов синхронизации	125
3.3. Вычисление угла упреждения и разработка алгоритмов управления наведением поворотных видеокамер на движущийся объект	127
3.4. Теоретическая оценка погрешностей наведения, обусловленных временными параметрами управления поворотными камерами	131
3.5. Определение критерия идентичности объектов на изображениях видеокамер	139
3.6. Теоретическая оценка погрешности сопряжения видеокамер на основе достоверности сопоставления семантических элементов изображений .	145
3.7. Адаптация алгоритмов семантической обработки изображений для систем и приборов видеонаблюдения	147
Выводы	152
4. Разработка, исследование и анализ методик сравнения алгоритмов семантического сжатия видеоинформации	154
4.1. Сравнение алгоритмов семантической сегментации изображений	154
4.2. Сравнение алгоритмов выделения движущихся объектов	158
4.3. Анализ методик оценки точности алгоритмов классификации	167
4.4. Разработка программного обеспечения для автоматизированной оценки достоверности алгоритмов обработки видеоизображений	174
Выводы	179
5. Исследование эффективности алгоритмов семантической обработки видеоинформации при анализе стереоизображений	181
5.1. Основы теории построения и анализа алгоритмов калибровки стереоизображений и восстановления глубины	181
5.2. Исследование эффективности алгоритмов выделения движущихся объектов на основе изображений одной видеокамеры	189
5.3. Разработка алгоритмов синхронизации кадров стереокамер на основе теории графов	194
5.4. Разработка и анализ алгоритмов выделения движущихся объектов с помощью стереосистем	196

5.5. Исследование способов построения алгоритмов выделения движущихся объектов одиночными видеокамерами и стереосистемами	198
5.6. Исследование способов повышения быстродействия алгоритмов обработки стереоизображений за счёт семантического сжатия	205
Выводы	207
6. Исследование алгоритмов обработки панорамных изображений, получаемых автоматически управляемыми поворотными видеокамерами в режиме постоянного сканирования зоны обзора	209
6.1. Современные алгоритмы построения панорамных изображений из набора цифровых снимков, сделанных из одной точки в различных направлениях	209
6.2. Исследование влияния выбора информации, описывающей характерные точки изображений, на точность и вычислительную сложность алгоритмов построения панорам	211
6.3. Анализ влияния предварительной юстировки поворотного устройства на эффективность алгоритмов обработки панорамных изображений . .	222
6.4. Разработка и исследование алгоритмов семантического выделения движущихся объектов на панорамных изображениях	233
6.5. Анализ алгоритмов сопряжения панорамного изображения с планом местности	239
Выводы	242
7. Экспериментальная оценка эффективности разработанных алгоритмов сопряжения, синхронизации, управления и семантического сжатия изображений. Результаты внедрения	243
7.1. Внедрение результатов и апробация	243
7.2. Описание схемы видеосистем с приборами и блоками семантической обработки изображений	245
7.3. Сравнение эффективности систем технического зрения с различными принципами действия	248
7.4. Методика оценки точности сопровождения движущихся объектов . .	254
7.5. Экспериментальная оценка точности управления поворотными видеокамерами	256
7.6. Методика оценки достоверности сопоставления изображений объектов на сопряжённых видеокамерах	266
7.7. Экспериментальные результаты сопоставления синхронных изображений объектов	268

7.8. Экспериментальная оценка эффективности автоматизированной системы тестирования алгоритмов семантической обработки изображений	274
7.9. Результаты применения алгоритмов семантического сжатия изображений. Оценка достоверности	277
Выводы	281
Заключение	283
Список литературы	285
Том 2	
Приложение 1. Свидетельства о регистрации интеллектуальной собственности	4
Приложение 2. Протоколы испытаний	10
Протокол испытания точности наведения поворотных видеокамер на подвижные объекты	10
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Курской АЭС	13
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Бурейской ГЭС совместно с ОАО «РусГидро»	19
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» в губе Териберская совместно с ООО «Питер Газ»	22
Протокол испытания приборов подсчёта посетителей ВАРШ.463135.002-01 в Международном аэропорту «Шереметьево»	26
Протокол испытания системы технического зрения «Orwell2k-IP» в Международном аэропорту «Шереметьево»	27
Приложение 3. Акты внедрения алгоритмов семантической обработки видеоизображений и управления приборными комплексами технического зрения в серийном производстве, разработке и эксплуатации	32
Акт АО «ЭЛВИС Неотек» о внедрении алгоритмов управления видеоприборами, синхронизации и семантического сжатия в системе «Orwell2k» .	33
Акт АО «ЭЛВИС Неотек» о внедрении алгоритмов семантической обработки стерео- и моноизображений в устройствах «Statistics»(ВАРШ.468196.001) и «Statistics-3d»(ВАРШ.463135.002-01) .	35

Акт ОАО НПЦ «ЭЛВИС» о внедрении результатов диссертации в процесс проектирования архитектуры микросхем	36
Акт системного интегратора «ИнфинитиРус» об использовании результатов диссертации в устройствах ВАРШ.468196.001 и ВАРШ.463135.002-01 .	38
Акт системного интегратора «Лилана» об использовании результатов диссертации в комплексах технического зрения, предназначенных для охраны сооружений Базузского гидроузла	39
Акт ООО «АДС» о внедрении описанных в диссертации алгоритмов в системы безопасности транспортной инфраструктуры олимпийских объектов Сочи	41
Приложение 4. Перечень патентов, относящихся к системам синхронной семантической обработки изображений	43
Приложение 5. Участие в выставках приборов, систем и технологий безопасности	44

Введение

Диссертационная работа посвящена разработке высокоеффективных по быстродействию и достоверности алгоритмов семантической обработки изображений и развитию теории построения систем технического зрения, в которых применяется синхронизация процессов управления параметрами приборов, например, углами поворота поворотных платформ и углами зрения видеокамер, и процессов параллельной обработки изображений и сигналов от одного или нескольких датчиков для существенного улучшения основных характеристик комплексов технического зрения.

Актуальность проблемы. По мере развития вычислительной техники системы технического зрения используются всё шире, охватывая многие области деятельности человека [1]: от ставшей уже обыденной уборки помещений автоматами до автоматического контроля изменений ландшафтов при слежении из космоса. Системы технического зрения всё чаще применяют в автопилотах для автомобилей, для контроля качества конвейерного производства и даже в устройствах, предназначенных для игр и развлечений: при кажущейся несерёзности последнего массовая реализация подобных устройств может дать очень значительный экономический эффект для предприятия-производителя в частности и для страны-производителя в целом. Тем не менее, наиболее значимым и актуальным применением систем технического зрения, на наш взгляд, остаются охранные системы, предназначенные для предотвращения неправомерных действий. Современные технические средства, предназначенные для охраны, улучшают такие важные показатели, как своевременная реакция на нарушение, количество задействованного в охранном комплексе персонала, трудоёмкость принятия решений при выполнении необходимых действий в ответ на нарушение. В последнее время проблема эффективной охраны порядка стала особенно актуальной в связи с повышением активности террористических организаций, поэтому задача совершенствования охранных приборов является насущной. Исходя из необходимости контроля соблюдения правопорядка, в последние годы органами местного управления по всему миру установлено огромное количество видеокамер, что ещё больше повысило актуальность создания эффективных систем технического зрения и систем видеонаблюдения как распространённый вид технических охранных средств. Как показано в исследованиях [2, 3, 4], использование средств видеонаблюдения в комплексе с административными мероприятиями позволяет более эффективно использовать ресурсы охранных структур и организаций.

Приблизительно до середины-конца 90-х годов XX века понятие «система видеонаблюдения» обозначало некоторый аппаратный комплекс, состоящий из видеокамер, аппаратуры, записывающей видеосигнал, терминалов с простым управлением для просмотра видео в записи и в реальном времени, а также средств коммуника-

ции между перечисленными элементами. Такие системы принято называть видеорегистраторами. Чаще всего в видеорегистраторах используются неподвижные камеры, реже — камеры на поворотных устройствах.

Подобного типа системы технического зрения обладают двумя основными недостатками. Во-первых, на операторов, в обязанности которых входит следить за происходящим на изображении, налагается высокая нагрузка, и на практике её можно выдержать только в течение 2–3 часов, даже если камера наблюдения всего одна. Во-вторых, поиск события в видеоархиве, если время события известно не точно, может быть проведён лишь при помощи просмотра архива целиком. Часто видеорегистраторы снабжаются алгоритмами определения движения в кадре, что несколько повышает эффективность использования системы видеонаблюдения. Однако простой детектор изменения изображения бесполезен в условиях интенсивного движения, например, на транспортных магистралях или многолюдных улицах.

К текущему моменту создано множество разработок в области семантической обработки цифровых изображений, которые используются для решения проблем технического зрения. Работы [5, 6, 7, 8, 9, 10] Ярославского Л.П., Грузмана И.С., Киричука В.С., Косых В.П., Перетягина Г.И., Кузнецова А.Е, Спектора А.А, У. Претта, Ханта Б.Р. и многих других учёных позволяют улучшить качество изображений и сигналов с помощью первичной фильтрации для повышения эффективности основной семантической обработки и для увеличения информативности данных. В работах Алпатова Б.А, Блохина А.Н., Бабаяна П.В., Степашкина А.И. [11, 12, 13, 14], а также в работах [15, 16, 17, 18, 19, 20] зарубежных учёных Т. Канаде, Р. Колинза, К. Стоуфера, В. Гримсона, Д. Коминисью, Дж. Малика и других предложены различные алгоритмы выделения положения движущихся объектов в области изображений при обработке потока сигналов от видеоприборов. Подобные алгоритмы могут быть использованы для привлечения внимания оператора видеосистем к изображению видеокамеры, если на изображении происходит какое-то событие. Ещё большее влияние на развитие систем технического зрения оказали достижения в области машинного обучения и информационной семантической обработки, классификации и распознавании видеосигналов, которые основаны на теории, изложенной в работах [21, 22, 23, 24] Харкевича А.А., Ковалевского В.А., Ф. Розенблата, Бравермана Э.М., Айзermana M.A., Розоноэра Л.А. и получившей дальнейшее развитие в работах [25, 26, 27, 28, 29, 30, 31, 32] Галушкина А.И., Ивахненко А.Г., Вапника В.Н., Червоненкиса А.Я., Журавлёва Ю.И., Злобина В.К., Р. Дуда, Д. Сторка, Р. Харта и многих других. В теории распознавания образов существенно используется теория статистических решений, в развитие которой основополагающий вклад внесли Колмогоров А.Н., Дж. Нейман, К. Пирсен. Семантическая классификация изображений

позволяет создавать приборы обработки, которые не только реагируют на произвольное изменение в области изображения видеокамер, но и анализируют тип изменений. В итоге внимание операторов систем технического зрения привлекается не к каждому изменению в области наблюдения, а только к тем, которые действительно представляют интерес с позиции решаемых аналитической видеосистемой задач. В связи с проблемой распознавания и классификации изображений следует отметить большие практические успехи [33, 34, 35, 36, 37], которые в последние несколько лет были достигнуты при применении аппарата так называемых свёрточных нейронных сетей в задачах машинного обучения: достоверность классификации изображений, полученная с помощью нейронных сетей в данных работах, сопоставима с качеством классификации, выполняемой человеком. Однако алгоритмическая сложность глубоких свёрточных нейронных сетей настолько высока, что на данный момент их невозможно полноценно применять в системах реального времени, поскольку для этого нет достаточно мощных вычислительных устройств.

Процесс развития систем технического зрения неотделим от процесса развития техники, и качество современных систем определяется характеристиками оптических приборов, эффективностью обработки сигналов и быстродействием вычислительных устройств. Кроме того, большую роль играет развитие средств для разработки алгоритмов и программ. Прогресс в области быстродействия вычислительных машин и компьютеров в настоящее время позволил достичь таких показателей производительности ЭВМ универсального назначения, что воплощение многих из перечисленных методов для систем обработки видеосигналов реального времени уже не является такой же сложной технической задачей, как это было десятилетие назад, тем более появляются специализированные вычислительные системы на кристаллах и микросхемы, специально предназначенные для быстродействующей семантической обработки сигналов. Большой вклад в развитие элементной базы обработки сигналов, вычислительной техники и в теорию построения и разработки алгоритмов внесли Котельников В.А., К. Шенон, Гуляев Ю.В., Глушков В.М., Валиев К.А., Гуськов Г.Я., Преснухин Л.Н., Стемпковский А.Л., Филатов В.Н., Красников Г.Я., Бетелин В.Б., Немудров В.Г., Шахнов В.А., Корячко В.П., Бархоткин В.А, Петричкович Я.Я. и многие другие. Тем не менее, проблема быстрой обработки изображений до сих пор актуальна, поскольку многие теоретические разработки в области семантической обработки видеосигналов применимы лишь в режиме постобработки, а не в реальном времени, поскольку обработка одного кадра видеопоследовательности занимает слишком длительное время даже самыми современными вычислительными средствами.

В последнее время системы видеонаблюдения строятся по принципу объединения в единую сеть различных датчиков [15, 38], регистрирующих обстановку в зоне на-

блюдения. Заметим, что датчики — это не только видеокамеры, но и, например, датчики объёма в помещениях, датчики открытия/закрытия дверей, датчики поворота турникетов и т.п. В том числе датчиками комплексов технического зрения являются видеоприборы: стационарные камеры и камеры с возможностью управления положением оптической оси и величиной оптической силы объектива. Цель объединения видеокамер в единую систему заключается в том, чтобы выдавать общую информацию о ситуации в зоне наблюдения в виде единого целого, а не как не связанные друг с другом изображения с видеокамер. Основное назначение подобной системы — максимально упростить анализ информации оператором за счёт первичной обработки и оптимизированного с точки зрения эргономики вывода данных о событиях, происходящих в зоне действия видеоприборов системы.

Одним из важнейших типов сопряжения видеоприборов является сопряжение видеокамер на поворотных устройствах с неподвижными видеокамерами. Поворотные видеокамеры — это приборы, состоящие из объектива, камеры с чувствительной матрицей и непосредственно механического устройства, которое позволяет изменять положение оптической оси объектива и направлять угол обзора видеокамеры в нужном направлении. Современные поворотные устройства могут изменять угол оптической оси со скоростью 300°c^{-1} и более. Кроме управления положением оптической оси, поворотные устройства могут также управлять увеличением объектива, изменяя угол зрения и фокусное расстояние с помощью управления системой линз. Поворотная видеокамера с объективом, диапазон фокусного расстояния которого 10–86мм, может получать изображения высокого разрешения для объектов в радиусе до 200 метров от места установки видеокамеры. В то же время возможности по использованию современных поворотных видеокамер в большинстве случаев ограничиваются ручным управлением и наблюдением за обстановкой в поле зрения видеокамеры с помощью оператора, поскольку в режиме перемещения оптической оси автоматическое выделение новых объектов в поле зрения поворотной камеры без участия человека является сложной и не до конца решённой задачей. Для автоматического выделения движущихся объектов на данный момент надёжнее всего применять неподвижные видеокамеры, однако для того, чтобы покрыть сектор, например, с углом 360° и радиусом 200 метров неподвижными видеокамерами с фокусным расстоянием 86мм для получения разрешений, эквивалентных разрешениям изображения поворотной видеокамеры с фокусными расстояниями 10–86мм, потребуется установить несколько тысяч неподвижных видеокамер. Кроме того, фиксированный угол зрения, соответствующий фокусному расстоянию 86мм, не обязательно оптimalен для получения изображения объектов. Проблема получения изображений высокого разрешения может быть решена с помощью сопряжения двух приборов: поворотной и неподвижной видеока-

меры. Неподвижная видеокамера должна определять новые объекты, появляющиеся в области наблюдения, а поворотная видеокамера должна наводиться на эти объекты и сопровождать их в автоматическом режиме, управляя параметрами поворота и увеличения таким образом, чтобы разрешение объекта было наилучшим, и в то же время чтобы объект был в кадре, несмотря на погрешность наведения. Как показывает практика, подобный подход может в 5–10 раз сократить количество видеоприборов, требуемых для получения изображения высокого разрешения.

Связь видеокамер между собой требует настройки ряда параметров, часть из которых нужно ввести вручную, а часть автоматически. Чем более прост и более автоматизирован процесс предварительной настройки, тем экономически более оправдано применение видеосистемы.

Таким образом, основная проблематика в области систем технического зрения заключается в трудоёмкости их настройки и эксплуатации. Несмотря на значительные усовершенствования, достигнутые благодаря развитию семантической обработки изображений, многие **проблемы** остаются не решёнными:

- операторы систем видеонаблюдения вынуждены постоянно следить за массивом изображений, поэтому из-за человеческого фактора велика вероятность пропуска событий в зоне наблюдения;
- для упрощения анализа информации, поступающей от приборов в информационных комплексах, используют связь видеокамер между собой, при этом возникает проблема оптимизации времени предварительной юстировки сопряжения перед вводом системы в эксплуатацию;
- необходимость увеличения разрешающей способности изображений, получаемых от неподвижных камер в составе видеосистем, требует значительного дополнительного количества неподвижных камер, поскольку зависимость между требуемым линейным увеличением разрешения и количеством камер квадратичная, естественно, что повышение разрешающей способности путём установки дополнительных неподвижных камер приводит к повышению сложности комплексов и увеличению затрат;
- в ряде приборных комплексов технического зрения есть возможность управления поворотными видеокамерами, и с их помощью можно получить изображение с более высокой разрешающей способностью, чем на неподвижной камере, причём сектор обзора поворотных камер достигает 360° , однако для управления поворотными камерами в существующих системах требуется непосредственное участие оператора, что повышает вероятность пропуска событий в зоне наблюдения, кроме того, многие поворотные видеокамеры могут осуществлять сканирование области контроля в автоматическом режиме, но при этом эффективных алгоритмов семантической обработки панорамных изображений реального времени пока не существует;

— использование стереокамер потенциально может существенно увеличить достоверность идентификации объектов в приборных комплексах технического зрения, однако многие вопросы теории использования и обработки стереоизображений остаются также не решёнными.

Цель работы. Цель данной диссертации — повысить эффективность контроля зоны наблюдения с помощью комплексов технического зрения нового поколения за счёт разработки и практической реализации способов автоматического сопряжения и юстировки изображений видеокамер, создания и анализа алгоритмов синхронизации процессов семантической обработки информационных сигналов от сети видеоприборов и автоматического управления поворотными камерами при наведении на объекты в зоне наблюдения с последующим сопровождением или в режиме непрерывного сканирования.

Задачи работы. Для достижения поставленной цели необходимо решить следующие задачи:

1. Провести функциональный анализ процессов семантической обработки видеосигналов и разработать описание модели распределённой системы коллективной обработки изображений различными приборами технического зрения.
2. Вывести соотношения, определяющие параметры сопряжения приборов технического зрения, и создать на их основе алгоритмы сопоставления объектов в общей зоне контроля приборов и алгоритмы наведения поворотных камер на сопровождаемые объекты с адаптивным упреждением.
3. Разработать методику тестирования и контроля алгоритмов семантической обработки изображений.
4. Разработать алгоритмы семантического анализа стереоизображений и оценить эффективность их применения в системах технического зрения.
5. Разработать алгоритмы семантической обработки панорамных изображений реального времени для видеоприборов, установленных на поворотной платформе, которая движется в режиме непрерывного сканирования.

Методы исследований. Задачи диссертационной работы решены с применением теории сложности алгоритмов, методов обработки цифровых сигналов-изображений, теории вычислительных методов оптимизации, фактов проективной и сферической геометрии, теории вероятности и теории статистических решений.

Научная новизна состоит в разработке, анализе и применении в приборных комплексах и в системах технического зрения эффективных алгоритмов синхронизации процессов семантической обработки изображений и управления поворотными видеокамерами, а также в создании способов автоматизации юстировки параметров сопря-

жения приборов технического зрения. В работе развивается концепция систем видеонаблюдения, которая строится на совместной семантической обработке информации с различных видеокамер и на применении элементов теории распознавания образов. В процессе выполнения диссертационной работы впервые получены следующие новые научные результаты.

1. Впервые разработана теория параллельных вычислений для комплекса программ, осуществляющих коллективную обработку информации, поступающей от приборов технического зрения. Для данной модели создан новый сетевой протокол синхронизации данных, определены ограничения на количество сопряжённых приборов в сети комплексов технического зрения, а также сформулирована и решена проблема полуавтоматической юстировки начальных параметров сопряжения.

2. Выведены математические соотношения для сопряжения поворотных камер и приборов технического зрения, на основе которых созданы новые алгоритмы сопоставления изображений одного и того же объекта в поле зрения сопряжённых неподвижных видеокамер и алгоритмы автоматического наведения и сопровождения объектов поворотными камерами для получения изображений большего разрешения на основе адаптивного упреждения.

3. Разработана теория тестирования и контроля алгоритмов семантической обработки на основе формальной логики и математического аппарата теории множеств.

4. Предложен разработанный автором новый комбинированный алгоритм обработки моно- и стереоизображений, отслеживающий перемещение объектов с высокой достоверностью.

5. Создан новый быстродействующий параллельно-конвейерный алгоритм построения и анализа панорамных изображений в реальном времени для видеоприборов, установленных на поворотном устройстве и непрерывно сканирующих область контроля.

Практическая значимость. Разработанные в диссертации алгоритмы, способы и модели используются в следующих серийно изготавливаемых системах и приборных комплексах технического зрения:

1) системы видеонаблюдения с компьютерным зрением «Orwell2k» РАЯЖ 46652.001-ОС.ПЗ, ЛЦКБ.464412.002, ВАРШ.00001-01, ВАРШ.00003-01 (патенты РФ на полезные модели №36315 от 07.08.2003 и №36912 от 23.06.2003, патенты РФ №2265531 от 07.08.2003 и №2268497 от 23.06.2003);

2) видеодетектор «Orwell2k-Barrier», определяющий объекты в состоянии свободного полёта;

3) система подсчета зрителей в кинозалах «Orwell2k-Cinema» (патент РФ на по-

лезнью модель №47546 и на изобретение №2296434 от 14.05.2005);

4) приборы подсчёта объектов «Statistics» ВАРШ.468196.001 и «Statistics-3d» ВАРШ.463135.002-01;

5) системы контроля дорожного движения «Travio» ВАРШ.466452.001, ВАРШ.466452.002 (патенты РФ №2014104763, №2015102323, патенты РФ на полезную модель №2014104762, №2015102309, №2015102315);

которые были разработаны при непосредственном участии автора, что подтверждено рядом свидетельств о регистрации программ для ЭВМ. Таким образом, тема диссертации полностью соответствует направлению научно-производственной деятельности организации, в которой выполнена работа.

Программное обеспечение семейства «Orwell2k», реализованное на основе разработанных в диссертации алгоритмов сопряжения, управления и юстировки поворотными видеокамерами, поставляется на внутренний и зарубежный рынок в составе серийно производимых комплексов технического зрения и является конкурентоспособным продуктом на мировом уровне. **Преимущества «Orwell2k»** заключаются:

- в работоспособности при тяжёлых климатических и механических условиях эксплуатации;
- в автоматическом наведении поворотных камер на движущиеся объекты с установкой высокой разрешающей способности, при этом погрешность наведения на подвижные объекты составляет 3° , что при типичном угле зрения неподвижных камер 30° – 60° позволяет улучшить разрешающую способность на порядок по сравнению с исходной разрешающей способностью изображения от неподвижных видеокамер;
- в выводе событий на топографический план с локализацией места события;
- в применении критерия идентичности изображений объекта в поле зрения пересекающихся камер с достоверностью сопоставления 99,0%;
- в классификации всех объектов по типу «человек/машина/группа людей»;
- в существенном сокращении времени реакции системы «Orwell2k» на новые объекты, появляющиеся в зоне наблюдения, до 0,5с, что более чем на порядок превышает показатели существующих систем;
- в повышении на порядок точности подсчёта числа объектов за счёт анализа стерео-изображений по сравнению с существующим принципом обработки моноизображений и уменьшению погрешности подсчёта до 3%;
- в применении анализа панорамных изображений, что позволяет в 10 раз и более сократить количество камер, необходимых для контроля зон наблюдения.

Автор диссертации проводил исследования и разработки в рамках «Приоритетных направлений развития науки, технологий и техники РФ» и «Критических технологий РФ», утверждённых указом Президента РФ №899 от 7 июля 2011 года.

Достоверность результатов подтверждается хорошей сходимостью результатов моделирования основных параметров функционирования приборных комплексов технического зрения, разработанных с участием автора, со значениями параметров серийных образцов, полученными в ходе их промышленной эксплуатации на различных объектах.

Внедрение результатов. Результаты диссертационной работы внедрены в системах технического зрения «Orwell2k» (ВАРШ.00001-01, ВАРШ.00003-01, ЛЦКБ.464412.002), которые используются в целях обеспечения безопасности как в России, так и за рубежом, в системах контроля безопасности дорожного движения «Travio» (ВАРШ.466452.001, ВАРШ.466452.002), в приборах подсчёта объектов «Statistics» (ВАРШ.468196.001) и «Statistics-3d» (ВАРШ.463135.002-01). Данные приборные комплексы используются для охраны стратегически важных объектов, таких как аэропорты, железные дороги, ГЭС, АЭС. Некоторые из перечисленных систем применяют для мониторинга городских общественных территорий в целях обеспечения безопасности. Приборы подсчёта объектов широко используются для управления хозяйственной деятельностью предприятий, предоставляющих услуги населению. Применение систем подтверждено актами о внедрении и протоколами испытаний, проводимых на объектах внедрения.

Личный вклад. Автором выявлены проблемы повышения эффективности комплексов технического зрения и сформулированы задачи по их решению. Все материалы, изложенные в диссертации, а именно: научные положения, технические решения, математические модели, семантические алгоритмы и алгоритмы управления, методики и результаты испытаний опытных образцов — получены автором лично. Автор создал архитектуру программного обеспечения, осуществляющего семантическую обработку, и участвовал в разработке программ и комплексов технического зрения, в которых внедрены результаты диссертации. Работы, выполненные в соавторстве, посвящены конструкторскому моделированию, производственным и эксплуатационным вопросам.

На защиту выносится:

1. Впервые предложенная теория параллельного функционирования семантических обработчиков в комплексах технического зрения и способы полуавтоматической юстировки параметров сопряжения видеоприборов, позволяющие посредством синхронизации информации об анализируемых изображениях объединять в сети сопряжённые приборы технического зрения.
2. Новые алгоритмы сопряжения поворотных и неподвижных видеокамер, обеспечивающие достоверность сопоставления изображений в общей зоне контроля приборов 99,0%, а также автоматическое наведение поворотных камер с малыми погреш-

ностями $0,5^\circ$ на неподвижные объекты и 3° на подвижные объекты, позволяющие получать изображения на порядок большего разрешения, причём время автоматического наведения в 20 раз меньше времени, требуемого для ручного наведения.

3. Теория тестирования и контроля алгоритмов семантической обработки, на основе которой создана распределённая система контроля и тестирования «Супервизор», использованная в качестве программно-инструментального средства, позволившего в 20 раз сократить частоту ложных идентификаций во время анализа потока видеоизображений при неизменном уровне ошибок второго рода.

4. Новый комбинированный алгоритм семантического анализа стерео- и моноизображений, позволяющий вести подсчёт объектов с погрешностью 3%.

5. Впервые созданный алгоритм построения и семантической обработки панорамных изображений с частотой 25 кадров в секунду в реальном времени.

6. Внедрение результатов диссертации при разработке приборных комплексов технического зрения нового поколения «Orwell2k», в которых реализованы разработанные в диссертации алгоритмы и способы (свидетельство о регистрации программы №2003612604 от 28.11.2003, патенты РФ на полезные модели №36315 от 07.08.2003, №36912 от 23.06.2003, патенты РФ №2265531 от 07.08.2003, №2268497 от 23.06.2003) и которые внедрены, как уже отмечалось, на ряде важнейших объектов.

Апробация работы. Результаты диссертации докладывались на XLVI, XLVII и XLVIII научной конференции Московского физико-технического института, а также на XV, XVI и XVII конференциях молодых ученых, аспирантов и студентов по современным проблемам машиноведения в институте машиноведения им. А.А. Благонравова РАН. Информационные приборы и видеосистемы семейства «Orwell2k», в которых внедрены результаты работы, демонстрировались на 20 выставках. Алгоритмы и информационные технологии, разработанные автором диссертации и реализованные в системе, получили признание специалистов и отмечены соответствующими дипломами. В 2005 году видеосистема с компьютерным зрением «Orwell2k» удостоена II Национальной премии по безопасности «За укрепление безопасности России» («ЗУБР-2005»).

Публикации. Основное содержание диссертации отражено в 70 опубликованных работах, в том числе в 29 статьях в журналах, входящих в перечень, утверждённый ВАК. Без соавторов опубликовано 36 статей. В соавторстве получены семь патентов на изобретения, шесть свидетельств на полезную модель и шесть свидетельств о регистрации программ.

Структура и объём диссертации. Диссертация состоит из введения, семи глав, заключения, списка литературы и приложений. Работа содержит 250 страниц основ-

ного текста, 50 страниц с рисунками и таблицами, список литературы из 215 наименований и приложений на 44 страницах.

Содержание и результаты работы. Во введении даётся обоснование актуальности темы диссертации, формулируются основные цели и задачи работы, описывается практическое значение полученных результатов, а также структура диссертации и краткое содержание глав.

В главе 1 проведён обзор основных методов обработки цифровых видеосигналов, а также методов классификации изображений. Выявлены достоинства методов, позволяющие решать поставленные в работе задачи, и недостатки, которые необходимо устранить. На основе анализа существующих видеосистем определены необходимые параметры сопряжения и синхронизации изображения: оптимальная погрешность сопряжения поворотных и неподвижных видеокамер определена как $0,1^\circ$, предельная погрешность преобразования сопряжения неподвижных видеокамер, при которой ещё возможно эффективное сопоставление изображений, равна 4 пикселам, и задержка синхронизации данных не должна превышать $3 \cdot 10^{-2}$ с. Определены цели и задачи работы, заключающиеся в достижении выведенных параметров за счёт разработки высокоеффективных алгоритмов.

В главе 2 проведена разработка и функциональный анализ параллельных алгоритмов коллективной семантической обработки видеосигналов, созданы теоретические основы синхронизации и сопряжения изображений, описана схема параллельных вычислений распределённых обработчиков в комплексах технического зрения. Проведён вывод ограничений на количество камер, которые могут взаимодействовать друг с другом при использовании конкретного типа вычислительной техники. Выведены соотношения для сопряжения видеокамер: неподвижных с неподвижными, неподвижных с планом местности, и неподвижных и поворотных. Предложены способы автоматизации настройки сопряжения. Разработан алгоритм, полностью автоматизирующий настройку управления фокусировкой при сопряжении поворотных видеокамер.

В главе 3 на основе результатов предыдущей главы выполнены исследование и разработка алгоритмов синхронизации, управления и семантической обработки информации в системах технического зрения, приведены оценки их быстродействия, а также выведены соотношения для угла упреждения при управлении поворотной камерой в процессе наведения на движущиеся объекты, предложен критерий идентичности изображений объектов в зоне пересечения областей зрения двух видеокамер. Приведены оценки погрешности сопряжения поворотных и неподвижных видеокамер, а также разработан способ статистической оценки погрешности сопряжения. По результатам теоретической оценки погрешность наведения на неподвижный объект

$0,5^\circ$, на подвижный объект — 20° в исследованных условиях наведения. Рассмотрены способы применения алгоритмов распознавания в аналитических видеосистемах.

В главе 4 приводится исследование и анализ методик сравнения алгоритмов семантического сжатия видеоинформации, разработана теория тестирования алгоритмов семантического сжатия на основе теоретико-множественных операций. Показано, что методика тестирования выделения движущихся объектов должна существенным образом учитывать качество работы алгоритмов сопоставления движущихся областей изображений на последовательных кадрах. На основе предложенной теории создан прототип системы тестирования «Супервизор».

В главе 5 разработаны алгоритмы выделения движущихся объектов с помощью анализа изображений одной камеры, с помощью анализа стереоизображений, и с помощью комбинированного алгоритма. В результате точность идентификации движущихся моноизображений составила примерно 85%, точность идентификации движущихся стереоизображений составила 90%, а точность идентификации с помощью комбинированного алгоритма — 95%.

В главе 6 на основании исследования алгоритмов обработки панорамных изображений, получаемых автоматически управляемыми поворотными видеокамерами в режиме постоянного сканирования зоны обзора, создан новый быстродействующий алгоритм анализа панорамных изображений. Алгоритм основан на построении панорамы в реальном времени и применении разностной схемы для выделения движущихся изображений на неподвижной панораме. Достигнутое время обработки одного кадра 30–40мс позволяет обрабатывать изображения с частотой 25к/с, типичной для большинства видеокамер. Для панорамного изображения разработаны специальные поправки к методике сопряжения панорамы с планом местности.

В главе 7 описываются экспериментальная оценка эффективности разработанных алгоритмов сопряжения, синхронизации, управления и семантического сжатия изображений и практические результаты их внедрения. Предложены методики измерения погрешности сопряжения и достоверности результатов алгоритмов распознавания. Проведены экспериментальные измерения, в результате которых оказалось, что погрешность наведения поворотных видеокамер на неподвижный объект равна $0,5^\circ$, а погрешность наведения на подвижный объект равна 3° . Рассмотрены основные результаты работы, которые нашли практическое внедрение при разработке аналитических видеосистем нового поколения. Экспериментально подтверждена эффективность тестовой системы «Супервизор», созданной на основе теории, предложенной в главе 4, в том числе на примере анализа стереоизображений. Проведены сравнительные испытания радарной системы и систем видеонаблюдения с неподвижными видеокамерами, тепловизионными видеокамерами в режиме шагового сканирования, и

видеокамерами в режиме панорамного сканирования. Анализ результатов испытаний показал преимущества радарной системы и системы видеонаблюдения с неподвижными камерами по достоверности идентификации перед системами сканирования местности видеоприборами на поворотной платформе, и выявил направления доработок панорамной системы сканирования.

В заключении приведены основные результаты и выводы по диссертационной работе.

В приложениях включены свидетельства о регистрации программ (приложение 1), протоколы испытания приборных комплексов технического зрения на некоторых объектах государственной важности, подтверждающие экспериментальную часть работы (приложение 2), акты внедрения результатов исследования на некоторых предприятиях и объектах (приложение 3), список патентов, которые защищают интеллектуальную собственность результатов, полученных в диссертации (приложение 4) и список отраслевых выставок приборов, систем и технологий обеспечения безопасности, в которых проходили апробацию результаты работы (приложение 5).

1. Алгоритмы семантического сжатия цифровых видеосигналов

1.1. Алгоритмы семантического сжатия как основа современных комплексов технического зрения

Этапы развития систем технического зрения. Первые системы видеонаблюдения появились вместе с изобретением телевидения в 30-х годах XX-го века[49]. До 90-х годов под понятием «система видеонаблюдения» подразумевался некоторый аппаратный комплекс, состоящий из видеокамер, устройств, записывающих видеосигнал, терминалов с простым управлением для просмотра видео в записи и в реальном времени, а также средств коммуникации между перечисленными элементами.

Несмотря на не слишком большой набор предоставляемых функций, системы подобного типа очень широко применяются и по сей день, поскольку их просто проектировать, устанавливать и использовать. В ходе технического развития таких комплексов изменились лишь системы трансляции видеосигналов, а также способы сохранения информационных видеомассивов: вместо аналоговой формы сигнала применяется цифровой способ передачи, для хранения информации используются не магнитные аналоговые носители, а цифровые накопители, как правило, жёсткие диски. Кроме того, по мере совершенствования элементной базы, устройства, являющиеся ядром системы видеонаблюдения, называемые также «видеорегистраторами», становились всё более и более компактными. Но идеология, заложенная в систему, основное назначение которой заключается в поддержке охраны объектов и территорий, существенно не меняется при применении устройств рассматриваемого класса.

Ассортимент видеорегистраторов на сегодня очень велик, полный каталог продукции данного типа значительно превысил бы объём данной работы. Videоприборы данного типа применяются в разных областях, например, существуют видеорегистраторы для фиксирования событий вокруг автомобиля на носители видеосигнала[50]. Приведём краткую спецификацию одного из типичных видеорегистраторов „Sanyo DSR-3000P”[51]. Основные функции „DSR-3000P” заключаются в записи видео на встроенный жесткий диск, причём в целях резервирования и повышения надёжности запись может производится на два жёстких диска одновременно. Для экономии пространства, занимаемого архивированным видеосигналом, перед записью производится сжатие с помощью специальных алгоритмов компрессии. Предоставляется возможность просмотра архива видео в прямом и обратном направлении в обычном и ускоренном режиме. Присутствует возможность записи видео только в том случае, если на входном сигнале от видеокамеры присутствуют значительные изменения на наблюданной сцене. Таким образом, видеорегистратор „DSR-3000P” является не только устройством записи и чтения сигналов с видеокамер, в нём также реализованы неко-

торые функции обработки видеосигналов с целью сжатия видеоархива и определения движения на наблюдаемой территории. Функции сжатия видеосигнала и простого анализа движения стали типичным дополнением к обычным функциям видеорегистраторов примерно с середины 90-х годов XX-го века.

Развитие методов цифровой обработки видеосигналов началось в 70-х годах, чему значительно способствовало изобретение ПЗС-матриц[52] и повсеместного внедрения видеокамер на их основе. К середине 90-х годов появились высокопроизводительные ЭВМ, благодаря которым стало возможным применять в режиме реального времени многие из разработанных ранее методов обработки видеосигналов. Это привело к тому, что возникли системы видеонаблюдения нового типа, в которых на основе обработки видеосигналов автоматизировались функции анализа информации, ранее выполняемые операторами систем.

Одни из первых работ, посвящённые описанию интеллектуальных систем видеонаблюдения нового поколения, появились в результате исследований многих коллективов учёных[15, 38]. Остановимся подробнее на описании достижений авторов данных работ в области технических новаций в системах видеонаблюдения. Для определённости введём понятие «аналитическая система видеонаблюдения с элементами искусственного интеллекта», далее — просто «аналитическая видеосистема» или «интеллектуальная видеосистема», чтобы обозначить системы, подобные [15, 38].

Функциональная схема типичных аналитических видеосистем. На рисунке 1.1 представлена структура современной аналитической видеосистемы, включающей в себя множество видеокамер, из которых часть является неподвижно закреплёнными, часть устанавливается на поворотных устройствах, причём у поворотных видеокамер, как правило, объектив с управляемым увеличением. Видеокамеры подключаются к одной или нескольким объединённым локальной вычислительной сетью (ЛВС) ЭВМ, назначение которых обрабатывать и анализировать видеосигналы.

Видеосистема состоит из кластеров оборудования двух типов: оборудование зоны наблюдения и оборудование контрольного пункта (в случае небольших видеосистем оборудование контрольных пунктов может объединяться оборудованием зон наблюдения).

Подвижные и стационарные видеокамеры, а также ЭВМ-серверы, обрабатывающие видеосигнал с этих камер и управляющие поворотными видеокамерами, объединяются в оборудование одной зоны наблюдения по принципу территориальной близости, которая позволяет соединить серверы быстродействующей сетью, и организовать передачу сигналов видео и управления между камерами и ЭВМ. Также видеокамеры объединяются в оборудование одной зоны наблюдения, если у них есть возможность

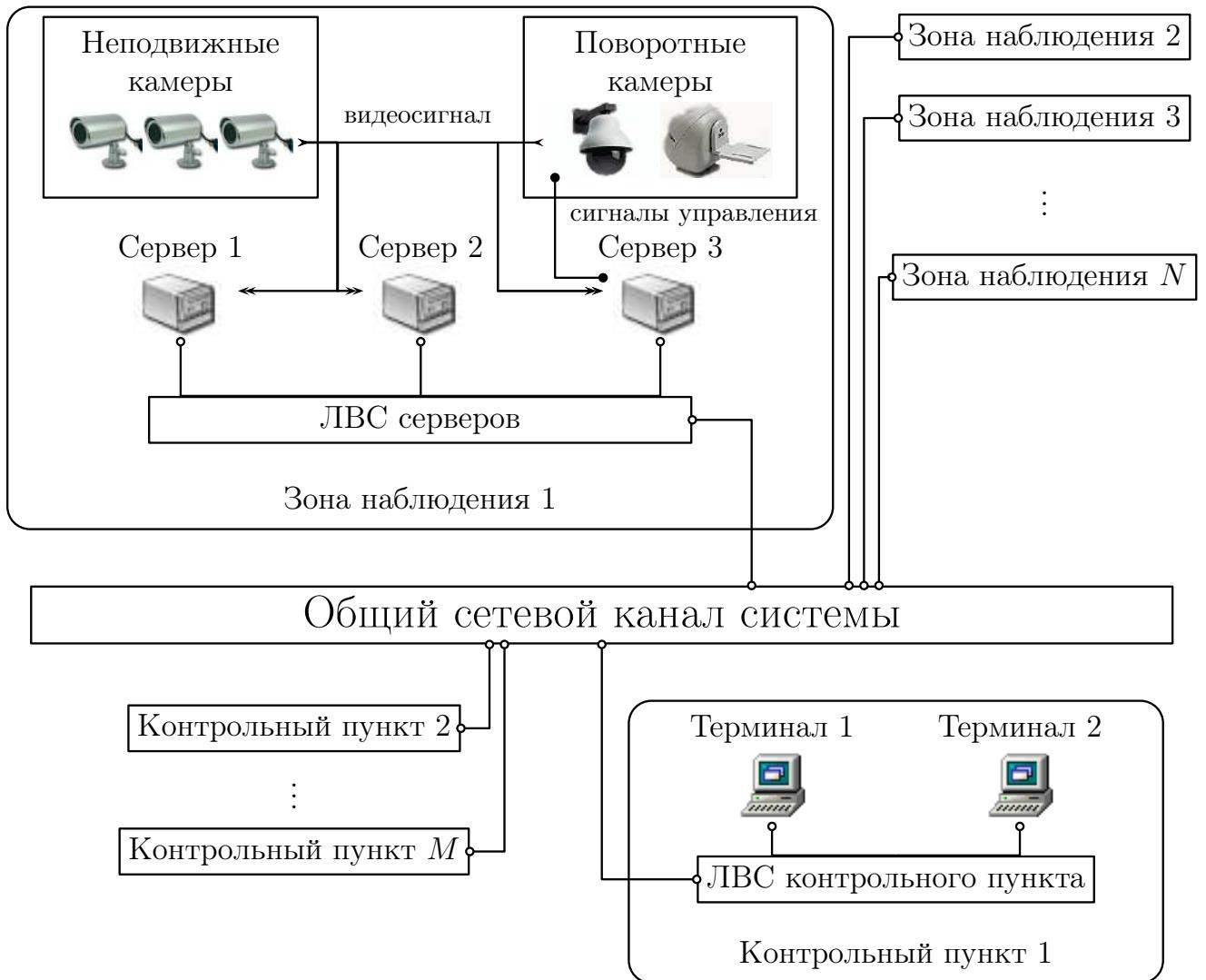


Рисунок 1.1. Общая схема аналитической видеосистемы.

совместного наблюдения за событиями, происходящими в области зрения данных видеокамер.

В контрольных пунктах с помощью терминалов видеосистемы отображается переданная по вычислительной сети информация, полученная в результате обработки видеосигналов на серверах зон наблюдения. В ряде случаев требуется удалённое наблюдение за территорией из нескольких различных контрольных пунктов. Для решения данной задачи используются ЛВС, построенные на оптоволоконных каналах связи.

Структура, представленная на рис. 1.1, может описывать не только аналитические видеосистемы, но и более просто устроенные видеорегистраторы. Рассмотрим основные понятия, описывающие отличительные характеристики аналитических видеосистем.

Основные понятия и определения для аналитических видеосистем. Одна из основных черт, отличающая аналитическую видеосистему от видеорегистратора, — это применение достаточно сложных алгоритмов и методов обработки изображений,

полученных с видеокамер с целью автоматического выявления ситуаций в поле зрения камер, представляющих интерес для операторов системы. Чаще всего автоматическое выявление ситуаций из видеосигнала обзорных камер основано на алгоритмах выделения движущихся объектов наблюдаемой сцены и алгоритмах классификации данных объектов. Чтобы формализовать понятие «алгоритм выделения движущихся объектов», определим видеосигнал как последовательность кадров $\{F_i\}_{i=1}^{\infty}$, в которой каждый кадр F_i можно рассматривать как вектор-функцию:

$$\begin{aligned} F_i : (x, y) &\rightarrow (F_i^R(x, y), F_i^G(x, y), F_i^B(x, y)) = (R, G, B), \\ x, y, R, G, B &\in \mathbb{Z} \\ 1 \leq x \leq W, 1 \leq y &\leq H, W \times H \text{ — разрешение кадра}, \\ 0 \leq R \leq 255, 0 \leq G &\leq 255, 0 \leq B \leq 255, \end{aligned} \tag{1.1}$$

то есть каждой паре (x, y) координат точки изображения сопоставляется три целых числа (R, G, B) , являющиеся компонентами цветности: красной, синей и зелёной, соответственно. Таким образом, кадр представляется тремя одномерными функциями F_i^R , F_i^G и F_i^B . Данное представление соответствует формату кадра, который принято называть «RGB24».

Определение 1.1. Алгоритм выделения движущихся объектов, который мы обозначим \mathcal{A} , является процедурой, принимающей на вход один за другим кадры последовательности F_i . Результатом процедуры является некоторая информация \mathcal{I}_i о движущихся объектах. Таким образом, при помощи алгоритма \mathcal{A} для любого натурального N можно построить соответствие

$$\{F_i\}_{i=1}^N \xrightarrow{\mathcal{A}} \mathcal{I}_N. \tag{1.2}$$

Структура информации \mathcal{I}_N зависит от цели, для которой предназначен алгоритм \mathcal{A} . Например, в видеорегистраторах алгоритмы определения движения выделяют лишь факт присутствия или отсутствия движущихся объектов в кадре, поэтому $\mathcal{I}_N = \{\delta_i\}_{i=1}^N$, где $\delta_i = 1$, если алгоритм \mathcal{A} обнаружил движение на i -м кадре, и $\delta_i = 0$ в противном случае. В [15, 38, 53] и во многих других источниках описываются алгоритмы выделения движения с более сложными выходными данными \mathcal{I}_N , содержащими в себе информацию о положении движущихся объектов на текущем и предыдущих кадрах. Формально это можно представить в виде

$$\mathcal{I}_N = \{\Omega_j\}_{j=1}^{M_N}, \Omega_j = \{\omega_i\}_{i=1}^N, \omega_i = (r_i, \chi_i), \tag{1.3}$$

где Ω_j — информация о j -м объекте из M_N объектов, обнаруженных алгоритмом \mathcal{A} на кадре N (если $M_N = 0$, то $\mathcal{I}_N = \emptyset$). Информация Ω_j состоит из набора данных об объекте ω_i на каждом i -м кадре: r_i — положение объекта, а χ_i — его свойства. Например, в качестве r_i могут быть координаты центра масс изображения объекта, либо

множество всех координат точек объекта (си卢эт), либо прямоугольная рамка, ограничивающая это множество. В качестве χ_i может быть класс объекта, полученный в процессе распознавания. Если объект отсутствует на i -м кадре, то $r_i = \varepsilon$ и $\chi_i = \varepsilon$, ε — «пустая» характеристика, специальное значение, показывающие отсутствие объекта на кадре или отказ алгоритма \mathcal{A} .

Обладая данными \mathcal{I}_N в форме (1.3), сравнительно легко строить гипотезы о тех или иных событиях: анализ траекторий объектов и их классификация позволяют с некоторой достоверностью автоматически выделять в зоне наблюдения аналитической видеосистемы события следующего типа: присутствие человека в запретной зоне

$$\exists N \exists \Omega_j \in \mathcal{I}_N \exists \omega_i = (r_i, \chi_i) \in \Omega_j : (r_i \text{ попадает в запрещённую зону}),$$

переход человека с шага на бег

$$\begin{aligned} \exists N \exists \Omega_j \in \mathcal{I}_N \exists \omega_i = (r_i, \text{«человек»}) \in \Omega_j \exists K < N : \\ (\text{скорость вдоль траектории } \{r_i\} \text{ до кадра } K \\ \text{в несколько раз меньше скорости после кадра } K), \end{aligned}$$

наличие движения в i -м кадре

$$\exists \Omega_j \in \mathcal{I}_N \exists \omega_i \in \Omega_j : (r_i \neq \varepsilon),$$

остановка объекта

$$\begin{aligned} \exists N \exists \Omega_j = \{(r_i, \chi_i)\} \in \mathcal{I}_N : (\text{цифровой сигнал модуля} \\ \text{скорости вдоль траектории } \{r_i\} \text{ меньше шума} \\ \text{цифрового сигнала положения } \{r_i\}) \end{aligned}$$

и много других ситуаций. Именно достоверное определение событий по видеосигналу расширяет возможности практического применения аналитических видеосистем в различных приложениях по сравнению с видеорегистраторами: оператор таких систем может меньше уделять внимание непосредственному контролю изображения с видеокамер, ориентируясь на автоматические подсказки о ситуации в зоне наблюдения.

Другая особенность аналитических видеосистем, рассмотренных в [15, 38], заключается во взаимодействии поворотных камер и неподвижных для получения более детальных изображений событий на наблюдаемой территории. Автору данной диссертации не удалось найти подробного описания механизма взаимодействия управления поворотными камерами и программных модулей анализа видеосигнала с неподвижными камерами, но сама идея использовать крупные изображения за счёт управления увеличением представляется исключительно важной при использовании аналитических видеосистем. Введём определение сопряжения неподвижных и поворотных камер.

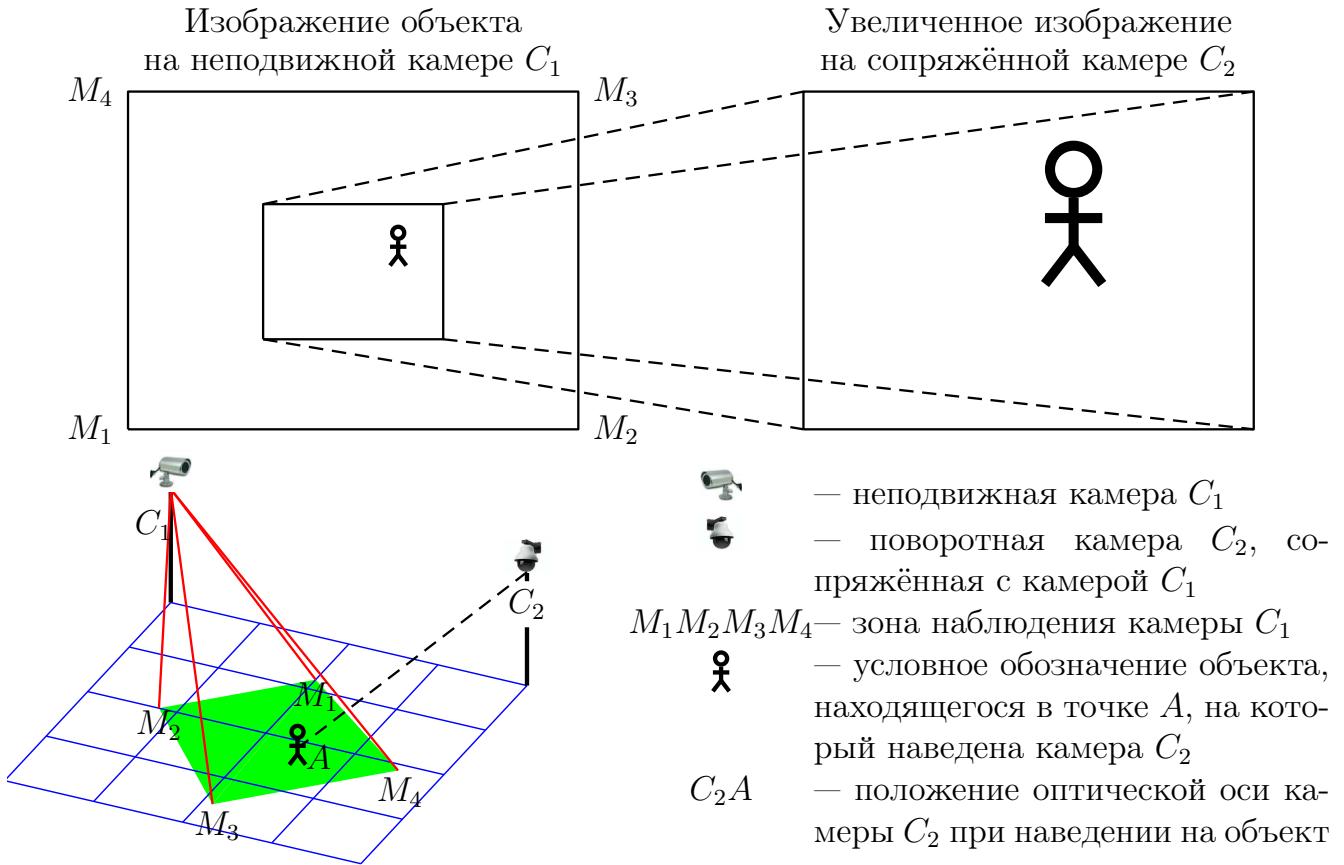


Рисунок 1.2. Схема сопряжения неподвижной и поворотной видеокамер.

Определение 1.2. Сопряжение пары неподвижной и поворотной видеокамер является операцией поиска функции, которая каждой области изображения неподвижной камеры сопоставляет такие параметры управления поворотного устройства и такие значения увеличения и фокусировки объектива видеокамеры, что при применении данных параметров изображение поворотной видеокамеры является сфокусированным изображением данной области с оптимальным увеличением.

Рисунок 1.2 иллюстрирует сопряжение неподвижной и поворотной камеры. На неподвижной видеокамере C_1 выделена область, содержащая изображение человека, который находится в территории наблюдения $M_1M_2M_3M_4$ камеры C_1 . Поворотная камера C_2 наведена на выделенную область, и показывает изображение с большим увеличением и лучшим разрешением. Ориентация оси C_2A определяется углами поворота, а увеличение — значением параметров управления объектива видеокамеры C_2 . Необходимые значения параметров управления поворотной камеры, которые задают наведение на требуемую область, согласно определению 1.2 получаются в результате специальной процедуры сопряжения видеокамер.

Кроме сопряжения неподвижных и поворотных камер, могут быть сопряжены две и более неподвижных камеры, зоны наблюдения которых пересекаются.

Определение 1.3. Сопряжение пары неподвижных камер — это поиск преобразования, переводящего координаты точки изображения одной видеокамеры в коорди-

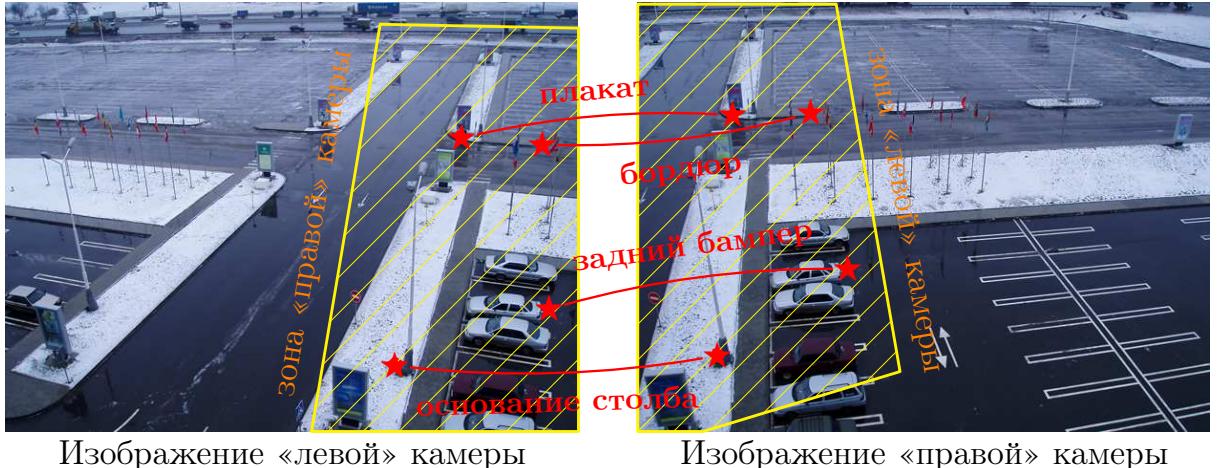


Рисунок 1.3. Сопряжение неподвижных видеокамер.

наты соответствующего изображения другой видеокамеры в случае, если обе видеокамеры направлены таким образом, что их зоны обзора пересекаются.

Пример сопряжения видеокамер представлен на рис. 1.3: на изображениях камер с общей зоной обзора отмечены две четвёрки точек, соответствующих изображению одних и тех же объектов. С помощью преобразования координат должны переводиться, например, отмеченные точки изображения левой камеры на отмеченные точки изображения правой камеры. Способ вычисления преобразования координат определяется в процессе сопряжения камер, согласно определению 1.3.

Основная цель сопряжения заключается в сопоставлении информации, получаемой при обработке видеосигналов от разных камер. Допустим, две видеокамеры поставляют синхронизированные по времени потоки кадров $\{F_i^1\}_{i=1}^\infty$ и $\{F_i^2\}_{i=1}^\infty$, соответственно. После применения алгоритма выделения и классификации движущихся объектов \mathcal{A} к первым N кадрам последовательностей получим информацию $\mathcal{I}_N^1 = \{\Omega_j^1\}$ и $\mathcal{I}_N^2 = \{\Omega_j^2\}$ об объектах на первой и второй камере в форме (1.3). Сопряжение двух видеокамер с общей зоной обзора позволяет выявить для некоторых объектов из \mathcal{I}_N^1 и \mathcal{I}_N^2 соответствие, т.е. определить, что оба объекта являются изображением одного физического объекта. Если установлено соответствие $\Omega_k^1 \leftrightarrow \Omega_m^2$, то можно говорить об «обобщённом объекте» $\{\Omega_k^1, \Omega_m^2\}$, который является совокупной информацией о перемещении одного физического объекта в области наблюдения сопряжённых видеокамер системы. Обобщённые объекты позволяют более гибко анализировать события, происходящие на наблюдаемой сцене, например, если объекты Ω_k^1 и Ω_m^2 ($\Omega_k^1 \leftrightarrow \Omega_m^2$) задействованы в ситуации, проанализированной как тревожная, то сообщение оператору должно быть выдано одно, а не два. Кроме того, можно выделять более сложные события, чем было указано ранее, например, можно выделить обобщённые объекты, побывавшие в поле зрения различных видеокамер в специально указанной последовательности.

Как наведение поворотных камер на движущиеся объекты, выделенные сопряжёнными, в смысле определения 1.2, неподвижными камерами, так и сопоставление объектов на сопряжённых, в смысле определения 1.3, требует синхронной обработки данных в параллельных модулях анализа видеосигнала, запускаемых, возможно, на различных серверах ЛВС видеосистемы. Например, в предыдущем примере для сопоставления объектов должны быть синхронизированы во времени последовательности кадров с двух видеокамер, а обработчики видеосигнала неподвижных камер должны обмениваться информацией об объектах в общей зоне наблюдения, то есть синхронизировать данные о положении объектов между собой. Модули управления поворотными видеокамерами также должны выполнять синхронизацию данных с неподвижными видеокамерами о положении и скорости объектов, на которые необходимо произвести наведение. Исходя из этих соображений, введём определение синхронизации.

Определение 1.4. *Синхронизация* — это способы обмена данными между параллельными обработчиками видеосигналов камер, сопряжённых между собой, а также алгоритмы взаимодействия обработчиков аналитической видеосистемы, например, для наведения поворотных видеокамер на объекты или для сопоставления объектов с различных неподвижных видеокамер.

Особый вид сопряжения, введённый в [15, 38] для аналитических видеосистем, — это сопряжение плана местности (топографической карты) с неподвижными камерами для того, чтобы повысить эргономические характеристики системы. Смысл сопряжения заключается в том, что поля зрения камер системы, а также выделенные с помощью анализа видеосигналов события, отображаются на изображении карты местности, содержащей территорию наблюдения. С помощью представления информации на плане местности удается облегчить восприятие оператором системы информации о происходящих событиях в зоне наблюдения. Определение сопряжения неподвижной камеры с планом местности можно свести к определению 1.3, если представить, что план местности — это изображение видеокамеры, расположенной достаточно высоко и обладающей оптической осью, перпендикулярной поверхности земли.

Понятие семантического сжатия изображений. В последние десятилетие резко возросло количество цифровых изображений, которые формируются во многих областях человеческой деятельности с помощью многочисленных устройств самого разного назначения. Многие из этих изображений сохраняются в общедоступных хранилищах мировой сети Интернет, в других случаях поток изображений сохраняется в системах с ограниченным в силу ряда обстоятельств доступом. Поскольку объём изображений становится огромен, остро стоит проблема индексации и поиска изображений. Даже поиск одинаковых изображений в общем виде является нетривиаль-



Рисунок 1.4. Пример исходного изображения 1000×750 точек (слева) и того же изображения, сжатого компрессором JPEG с большим коэффициентом.

ной задачей. При подобном поиске желательно найти не только файлы изображения, полностью, вплоть до последнего бита, совпадающие с исходным файлом, но и изображения с той же сценой, зарегистрированные в другое время или с несколько иначе произведённым кадрированием, а также желательно находить те же самые изображения с изменёнными характеристиками из-за предобработки цифровыми фильтрами.

Ещё более сложную проблему представляет собой поиск изображений по его семантическим признакам. Задачей поиска является выделение по текстовому семантическому описанию подходящих под описание изображений. Примером семантического описания изображений может служить следующий текст: изображение машин красного цвета. Для выполнения семантического поиска необходимо сначала провести операцию составления семантического индекса, которая заключается в выделении семантических признаков изображения с последующим построением сбалансированного дерева поиска из выделенных на изображении семантических характеристик. При операции выделения семантических признаков полученная информация существенно меньше по объёму, чем объём данных, необходимый для представления исходного изображения. Поэтому целесообразно ввести термин «семантическое сжатие» изображений для алгоритмов, которые выделяют отдельные семантические элементы. Поясним данный термин с помощью графических примеров.

На рис. 1.4 отображено исходное изображение городской сцены с ДТП в разрешении 1000×750 пикселей, а также то же самое изображение, сжатое компрессором JPEG с сильным коэффициентом сжатия, что приводит к заметности блочных шумовых артефактов. Исходное изображение при кодировании каналов R, G и B с помощью 8 бит в каждой точке изображения требует $2 \cdot 10^6$ байт для хранения, а изображение, преобразованное JPEG с сильным коэффициентом сжатия, требует $3 \cdot 10^4$ байт.

Рис. 1.5 является одним из примеров семантической разметки изображений, на котором выделены элементы изображения (люди, деревья, здания, дорога и др.). Каждый такой элемент представляет собой многоугольник с несколькими десятка-

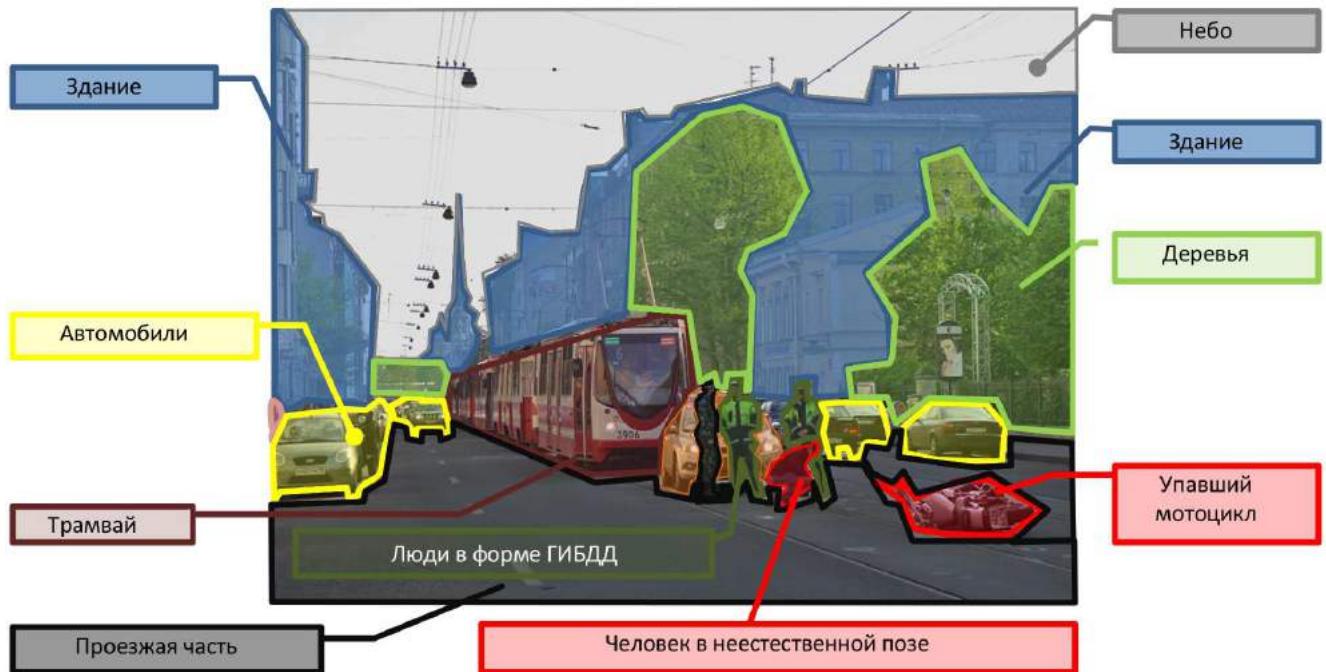


Рисунок 1.5. Пример семантической разметки изображения с помощью многоугольников

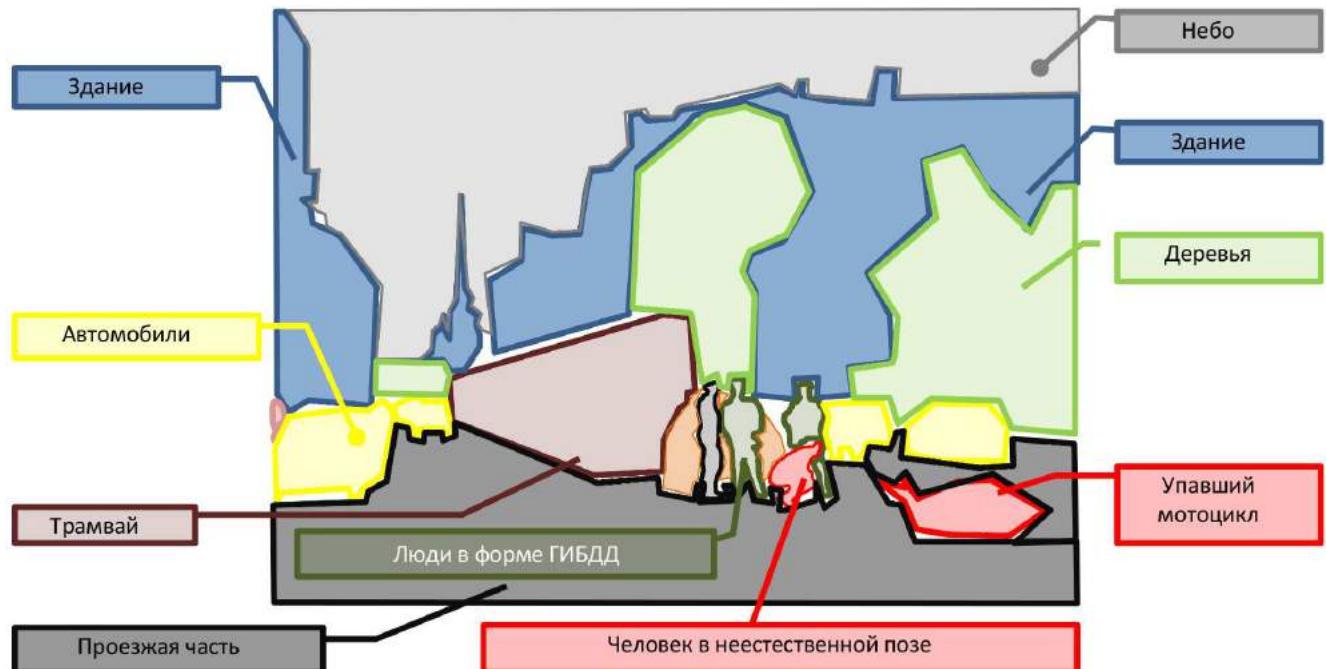


Рисунок 1.6. Иллюстрация информативности семантической разметки

ми вершин и текстовой меткой, описывающей выделенный семантический признак. На рис. 1.6 проиллюстрировано, что семантическая разметка является достаточно подробным описанием изображения, и даже без исходного изображения она даёт общее представление о том, что было изображено. Объём семантического описания с многоугольником включает в себя информацию с текстовыми метками и вершинами многоугольников, и занимает для приведённого примера около 10^3 байт.

Если в рамках анализа изображения относительное положение элементов изобра-

жения друг относительно друга не имеет значения, то исходное изображение можно описать с помощью текстового перечисления семантических элементов: здания, автомобили, трамвай, упавший мотоцикл, сотрудники ДПС, проезжая часть и т.д. Информационный объём такого описания по порядку величины 10^2 . Для описания типа наблюдаемого события, а именно: сцена ДТП, достаточно примерно 10 байт.

Представим себе, что приведённый пример изображения является одним из кадров видеопотока одной из многочисленных камер, входящих в систему ситуационного контроля крупного мегаполиса. В рамках анализа огромного объёма видеоданных, генерируемого видеокамерами подобных систем, часто необходимо для каждого изображения ограничиться лишь однобитовым триггерным семантическим описанием: есть на изображении событие «интереса» с точки зрения диспетчера, или нет. В случае необходимости разбора события данный бит можно подвергнуть «семантической декомпрессии», обратившись за расшифровкой к архиву видеоданных. Задача семантического сжатия в представленной формулировке является сложной проблемой, которая окончательно не решена. Современное состояние компьютерного зрения таково, что существует ряд известных открытых конкурсов по сегментации и классификации изображений [39, 40], в рамках которых тестируются различные подходы, самые известные из которых базируются на основе аппарата свёрточных нейронных сетей, либо на основе подхода SVM с нелинейными ядрами, либо на основе составных классификаторов с использованием гистограмм градиентов изображения и т.д. Результаты тестирования показывают, что лучшие из методов сегментации достигают точности порядка 50%. И хотя методы сегментации и классификации с каждым годом совершенствуются, их достоверность пока не достаточна для применения в промышленных масштабах. Тем не менее, существует ряд задач, для которых семантическое сжатие разработано с достаточной степенью достоверности [41, 42]. Существуют помехоустойчивые способы выделять движущиеся объекты в системах охранного видеонаблюдения, решена задача выявления возгораний на наблюданной сцене [43], предметов и вещей, оставленных без внимания, переброшенных через ограждение предметов, разработаны методы видеоконтроля транспортных средств в потоке и на парковках [44, 45], находят широкое применение системы бизнес-мониторинга [46]. Вышесказанное позволяет ввести определение семантического сжатия.

Определение 1.5. *Семантическое сжатие* — это операция преобразования множества изображений в информационное описание значительно меньшего объёма, чем исходный объём изображений, с обеспечением сохранения в сжатом описании элементов, достаточных для решения поставленной перед аналитической видеосистемой задачи.

В ряде случаев возможно построение преобразования, обратного к преобразова-

нию семантического сжатия, при котором на основе информации небольшого объёма, полученной при обработке множества видеокадров, формируется более подробная информация. Для этого необходимо обеспечивать архивирование видеокадров в процессе функционирования аналитических видеосистем. Тогда на основе операций сопряжения и синхронизации, определённых выше, возможно, например, по изображению низкого разрешения объекта с неподвижных камер восстановить либо изображение высокого разрешения этого же объекта на поворотных камерах, либо изображение в другом ракурсе, зафиксированное сопряжённой неподвижной камерой. Также по времени и месту автоматически зафиксированного события интереса восстановить видео-последовательность, содержащую изображение события, в видеоархиве. Для краткого обозначения операции восстановления информации после применения методов семантического сжатия введём следующее определение.

Определение 1.6. Преобразование, обратное к операции семантического сжатия, построенное на основе видеоархива, семантической обработки и операций синхронизации и сопряжения, называется *семантической декомпрессией*.

Таким образом, концепция семантического сжатия изображений обобщает принципы построения сложных алгоритмов обработки видеоинформации, применяемых как при индексации поиска в больших массивах видеоданных, так и при построении современных видеосистем контроля и анализа наблюдаемых сцен и событий, состоящих из большого количества видеокамер и других датчиков. Теоретическое и практическое развитие алгоритмов семантического сжатия существенно увеличивает значимость аналитических видеосистем в составе сложных экспертных систем поддержки принятия решений при контроле крупных инфраструктурных объектов.

Существующие аналитические видеосистемы. На текущий момент существует несколько десятков систем видеонаблюдения, которые в той или иной степени обладают функциями аналитических видеосистем. Отметим среди них [53, 54, 55, 56, 57]. Указанные источники являются лишь перечислением функций систем, а не научными статьями с теоретическим и экспериментальным обоснованием, тем не менее, из них тоже можно получить представление о современном состоянии отрасли.

Общая схема выполнения функций аналитических видеосистем представлена на рис. 1.7. Как следует из данной схемы, время реакции на появление нового объекта в зоне наблюдения без автоматизированных функций управления поворотными камерами и локализации местоположения объекта на плане местности составляет примерно 10с.

Источник [53] описывает реализацию алгоритмов системы „W4”, которая, судя по приведённым экспериментальным результатам, очень точно выделяет информацию в форме (1.3) для обрабатываемых последовательностей кадров, что позволяет, как

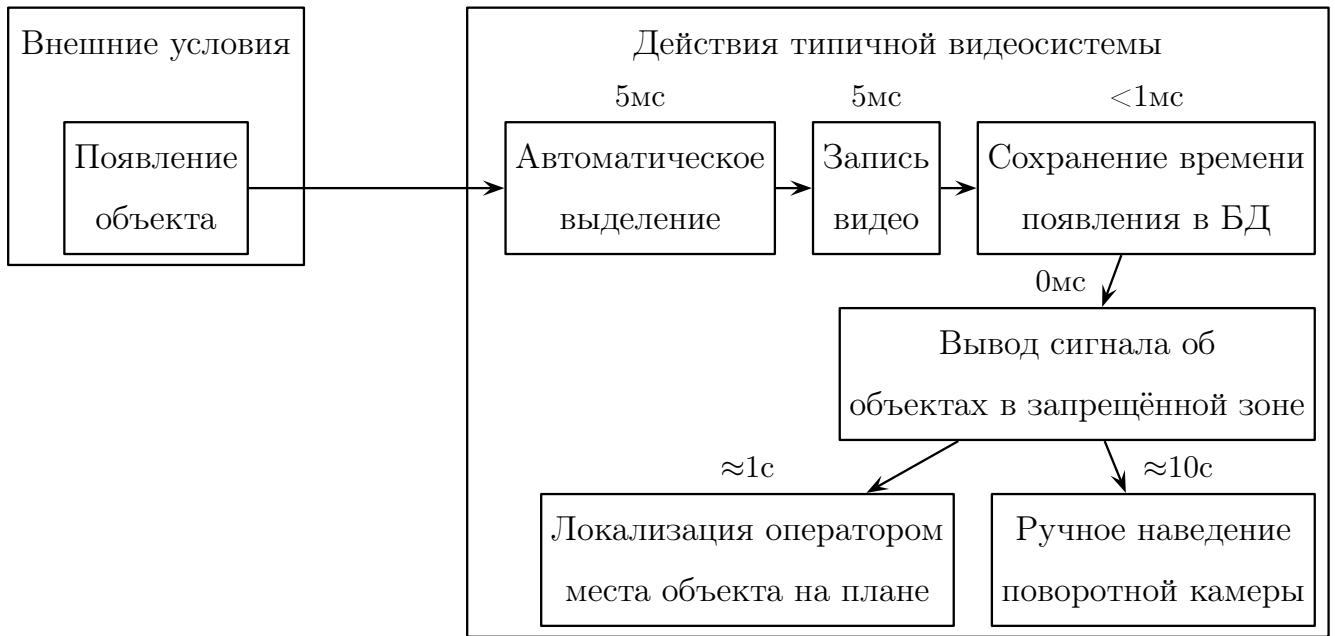


Рисунок 1.7. Выполнение функциональных блоков современных видеосистем.

уже отмечалось выше, определять широкий спектр событий автоматически.

В системе „iOmniscient”[54] декларируется автоматическое выделение из видеосигнала событий многих типов: определение оставленных предметов среди оживлённого движения людей, выделение краж, определение по видео людей, которые упали или поскользнулись, определение проникновения в запретную зону. Наряду с этим реализованы функции подсчёта количества людей, проходящих в определённом месте перед камерой. Сопряжение камер между собой и с планом местности, а также использование поворотных камер в описании [54] не отмечено.

Система „IoImage”[55] также обладает «интеллектуальными» функциями выделения движущихся объектов, определения оставленного без присмотра багажа, проникновения в запретную зону. Кроме того, в ней реализовано автоматизированное управление поворотными камерами, позволяющие следить за объектами крупным планом. Сопряжение камер между собой и с планом местности не упоминается.

В системе «Интеллект»[56] описываются функции определения автомобильных номеров и номеров железнодорожных вагонов, а также функции выделения и распознавания лиц. Описывается возможность связи видеокамер с планом помещений. Сопряжение камер между собой, а также функции слежения за объектами не упоминаются.

Инструкция по использованию системы „Goal v8”[57] содержит упоминание о возможности детектирования по видеосигналу бегущих людей, резкого возрастания числа людей на видеокамере, проникновение в запретную зону, определение оставленных или, наоборот, унесённых предметов. В системе предусмотрено автоматизированное управление поворотными камерами, и есть возможность использования внутренних

планов помещений. Среди функций системы также можно отметить возможность сопряжения модулей, анализирующих видеосигнал и выявляющих события, и, например, объёмными датчиками движения с целью сокращения числа ложных срабатываний. Сопряжение видеокамер между собой и сопряжение с планом местности в данном источнике не упомянуты.

Таким образом, существует не так много современных систем видеонаблюдения, в которых реализованы все функции, описанные в работах [15, 38] десятилетней давности. По-видимому, причины заключаются в том, что приведённые алгоритмы и методы являлись либо требовательными к ресурсам ЭВМ настолько, что промышленный вариант применения становился очень дорогим, либо качество алгоритмов не позволяло использовать системы с их реализацией из-за неприемлемости выходных характеристик по точности результатов.

1.2. Выбор языковой среды, компиляторов и систем визуальной программной разработки алгоритмов семантического сжатия

В настоящее время можно выделить два направления разработки аналитических видеосистем, которые различаются по типу оборудования, используемого в качестве вычислительных приборов системы.

Одно направление заключается в том, чтобы в качестве вычислительных устройств использовать специально разработанное для этих целей аппаратное обеспечение. Чаще всего в основе таких устройств специализированные процессоры цифровой обработки сигналов, иначе называемые DSP-процессорами (см., например, [58]). Пример аналитической видеосистемы на основе специализированных вычислителей см. в [55].

Второе направление базируется на том, что программное обеспечение аналитических видеосистем разрабатывается полностью для ЭВМ универсального назначения. Обычно для аналитических видеосистем используют ЭВМ, в основе которых лежит центральный процессор с архитектурой x86-x64, впервые разработанной компанией «Intel», а затем усовершенствованной как самой компанией «Intel», так и другими предприятиями за рубежом и в России. В современном мире наиболее распространены ЭВМ именно этой архитектуры, усовершенствованной за 3 десятилетия с момента её создания.

Преимущество специализированных устройств обработки изображений перед универсальными ЭВМ заключается в том, что, во-первых, габариты специализированного прибора чаще всего меньше габаритов универсальной ЭВМ с архитектурой x86-x64, а во-вторых, специализированные приборы разрабатываются так, что для них в большинстве случаев не требуется операционная система, и следовательно, установка и пуско-наладка систем со специализированными приборами быстрее и удобнее,

чем установка аналитических видеосистем на универсальные ЭВМ, поскольку универсальная ЭВМ сама по себе требует установки и настройки операционной системы для нужд видеосистемы. Преимущество же видеосистем на основе универсальных ЭВМ заключается в том, что доработка и модификация программного обеспечения осуществляется стандартными средствами разработки, и потому циклы разработки программного обеспечения завершаются гораздо быстрее, чем циклы разработки специализированных приборов. Кроме того, процессоры архитектуры x86-x64 постоянно совершенствуются, причём не за счёт потери совместимости с программным обеспечением. За последнее десятилетие производительность ЭВМ архитектуры x86-x64 возросла по совокупным показателям приблизительно в 10 раз. Это означает, что программное обеспечение, единожды разработанное для ЭВМ архитектуры x86-x64, может работать всё эффективнее и эффективнее без изменения кодов программ из-за того, что аппаратное обеспечение постоянно совершенствуется мировым сообществом разработчиков, чему способствует конкуренция среди производителей микросхем. Кроме того, огромное количество прикладных программ, необходимых в самых разных сферах деятельности и разработанных для данной архитектуры, исключает трудности с элементной базой: всегда найдётся производитель, выпускающий приборы для ЭВМ данной архитектуры, что не обязательно выполняется для некоторых DSP-процессоров. Поэтому для разработки аналитических видеосистем чаще применяются универсальных ЭВМ архитектуры x86-x64.

С самого начала существования архитектуры x86-x64 языковая среда «C» стала одной из самых применяемых для разработки программного обеспечения. Этому способствовала гибкость конструкций языка, которые, с одной стороны, приближены к ассемблерным инструкциям машинного кода, а с другой стороны, более понятны, чем конструкции ассемблера, что упрощает разработку программ. Самые распространённые на сегодняшний день операционные системы: системы компании «Microsoft» и «Unix»-подобные системы написаны в большей степени на языке «C». Поскольку для аналитических видеосистем нужна операционная система, и поскольку при решении большинства задач при разработке систем видеонаблюдения более эффективно пользоваться существующими операционными системами, чем разрабатывать новые, то для создания оптимального кода с точки зрения скорости разработки и эффективного использования системных ресурсов удобнее всего использовать язык «C».

Несмотря на то, что язык «C» значительно проще ассемблерного языка, у него есть ряд недостатков, затрудняющих разработку сложных систем, каковыми являются аналитические видеосистемы. В нём отсутствует строгая проверка типов данных, при решении ряда задач требуется прибегать к запутанным синтаксическим конструк-

циям, затрудняющим поддержку кода, в сложных системах код сильно разрастается по объёму и становится трудным для понимания, затруднена коллективная разработка систем из-за ряда синтаксических особенностей, например из-за невозможности контролировать пересечения в пространстве имён функций и макросов у разных разработчиков.

Практически всех этих недостатков лишён язык «C++», который, с одной стороны, почти полностью включает в себя синтаксис языка «C», и поэтому в нём нисколько не уменьшились возможности по эффективному использованию ресурсов операционной системы, а с другой стороны, в «C++» реализованы синтаксические конструкции, позволяющие в полной мере применять принципы объектно-ориентированных подходов к программированию и разработке [59].

Подробное описание языка «C++» можно найти в [61], ниже перечисляются только основные достоинства языка, которые делают его практически незаменимым при разработке крупных программных проектов, в том числе аналитических видеосистем.

1. В 1998 году принят стандарт ISO для языка «C++» [60], и с этого момента производители компиляторов этого языка стремятся ему следовать. Поэтому упрощается перенос исходного кода из одной операционной системы в другую.

2. Стандарт [60] постоянно совершенствуется с целью приведения данного стандарта в соответствие с современными методологиями разработки программного обеспечения с использованием шаблонов проектирования, доказавшим свою эффективность. Официально стандарт был пересмотрен в 2011 и 2014 году, таким образом он поддерживается в актуальном состоянии, при этом сохраняется совместимость с новым стандартом для программных кодов, написанных в соответствии с предыдущими версиями стандарта.

3. В «C++» поддерживаются классы объектов и шаблоны, что позволяет упростить разработку за счёт более эффективного кода и более эффективных методов проектирования.

4. В языке введена поддержка различных пространств имён, поэтому коллектив разработчиков может более эффективно решать конфликты, возникающие при одинаковых названиях функций, методов и классов.

5. Многие синтаксические особенности языка «C++» нацелены на то, чтобы выявить ошибку на этапе компиляции, ещё до окончательной отладки программы, что повышает скорость завершения циклов разработки программного обеспечения.

6. Несмотря на то, что язык «C++» не создавался специально для решения математических задач (в отличие, например, от языка «FORTRAN»), за время существования языка разработано множество библиотек и модулей, упрощающие создание

математического программного обеспечения.

7. Многие современные компиляторы «C++», например, компилятор компании «Intel», широко используют автоматическую оптимизацию кода для различных современных расширений архитектуры процессора x86-x64, позволяющие за время выполнения одной команды процессора выполнить множество однотипных операций с данными с помощью так называемых SIMD-инструкций (Single Instruction — Multiple Data), что особенно важно в программах обработки видеосигналов.

Таким образом, язык «C++», в отличие от других языков высокого уровня, позволяет эффективно использовать ресурсы ЭВМ посредством операционной системы, и при этом снабжён удобными средствами объектно-ориентированной разработки программного обеспечением. Поэтому при разработке аналитических видеосистем чаще всего применяется именно этот язык программирования.

1.3. Классификация алгоритмов семантической обработки цифровых информационных видеосигналов

В системах видеонаблюдения, подобных [15, 38, 53, 54, 55, 56, 57], в той или иной используется методы семантической обработки видеосигналов. Среди них можно выделить несколько основных категорий: методы предварительной обработки видеосигналов, методы сопоставления изображений, выделение объектов из видеосигналов с последующей идентификацией, методы сопряжения видеокамер и методы автоматического управления поворотными видеокамерами. Не претендуя на полноту обзора, приведём несколько часто используемых методов в каждой из этих категорий.

Предварительная обработка видеосигналов. Перед анализом видеосигнала часто возникает необходимость подвергнуть изображение фильтрации, например, для упрощения извлечения информации на следующих этапах анализа, либо для понижения шума изображения, чтобы повысить качество функционирования анализирующих алгоритмов. Приведём некоторые часто применяемые алгоритмы фильтрации изображения, описанные в [5, 48, 10, 7, 62, 6].

Фильтры выделения границ используются для векторизации изображения и для выделения дополнительных признаков. Пусть F — кадр в формате (1.1), а F^* — таким образом преобразованный из F кадр, что область определения F^* (разрешение кадра F^*) совпадает с исходным, а в координатах граничных точек изображения F на кадре F^* яркость значительно превышает яркость в других точках F^* .

Наиболее часто для выделения границ используют фильтр Собела, описанный, например, в [10]. Формулы преобразования $F \rightarrow F^*$ выглядят следующим образом:

$$\forall C \in \{R, G, B\} \quad \forall(x, y) \in [2, W - 1] \times [2, H - 1] : F^{*C}(x, y) = \sqrt{a_C^2 + b_C^2}, \quad (1.4)$$

$$\begin{aligned} a_C &= (F^C(x+1, y-1) + 2F^C(x+1, y) + F^C(x+1, y+1)) - \\ &\quad -(F^C(x-1, y-1) + 2F^C(x-1, y) + F^C(x-1, y+1)), \\ b_C &= (F^C(x-1, y+1) + 2F^C(x, y+1) + F^C(x+1, y+1)) - \\ &\quad -(F^C(x-1, y-1) + 2F^C(x, y-1) + F^C(x+1, y-1)). \end{aligned}$$

Из формулы (1.4) следует, что в каждой точке (x, y) функции F^{*R}, F^{*G}, F^{*B} являются евклидовой нормой оценки градиентов функций F^R, F^G, F^B (векторы (a_R, b_R) , (a_G, b_G) , (a_B, b_B)), соответственно. Отсюда мы наблюдаем выделение границ на изображении F^* .

Повышение контрастности изображения F может быть довольно сложным (см. [6]), однако чаще всего используют преобразование кадра F в более контрастное изображение F^* простого типа:

$$\forall C \in \{R, G, B\} : F^{*C} = \frac{F_{max}^* - F_{min}^*}{F_{max} - F_{min}} (F^C - F_{min}) + F_{min}^*, \quad (1.5)$$

где $F_{min} = \min_{C, x, y} F_C(x, y)$, $F_{max} = \max_{C, x, y} F_C(x, y)$, а F_{min}^* и F_{max}^* — параметры контрастирования. Обычно выбирают предельные параметры $F_{min}^* = 0$ и $F_{max}^* = 255$.

Повышение контрастности позволяет улучшить визуальное восприятие изображения человеком, однако при этом не улучшаются характеристики, необходимые при дальнейшей обработке. Например, для алгоритма выделения особых точек объектов требуется улучшить качество изображения таким образом, чтобы его стало возможным обрабатывать: необходимо снижать зашумлённость изображения с помехой.

Фильтр Винера — одно из эффективных преобразований для снижения соотношения сигнал-шум (см. [48, 6]). Для того, чтобы ввести фильтр Винера, нам потребуются дополнительные обозначения. Пусть \tilde{F} — «идеальное» изображение, на котором отсутствует шум, а исходное изображение F является некоторымискажением идеального изображения, $F = \tilde{F} + N$, где N — это некоторый шум. Для идеального, исходного и преобразованного изображения, а также шума введём обозначения

$$\begin{aligned} x_j^i &= \tilde{F}_C(i \bmod W, j \bmod H), & y_j^i &= F_C(i \bmod W, j \bmod H), \\ \hat{x}_j^i &= F_C^*(i \bmod W, j \bmod H), & n_j^i &= N_C(i \bmod W, j \bmod H), \end{aligned} \quad (1.6)$$

где C обозначает одну из компонент цветности $\{R, G, B\}$ (фильтрация производится по каждому каналу цветности отдельно), а под операцией « \bmod » понимается остаток от деления левого числа на правое. Таким образом, $x_j^i, y_j^i, \hat{x}_j^i, n_j^i$ — это, соответственно, периодические продолжения функций $\tilde{F}_C, F_C, F_C^*, N_C$ прямоугольника $[1, W] \times [1, H]$ на всю двумерную целочисленную плоскость. Оценку \hat{x} сигнала без шума будем искать в виде

$$\hat{x}_j^i = \sum_{m, n} a_n^m \cdot y_{j-n}^{i-m}, \quad (1.7)$$

где суммирование идёт по $[1, W] \times [1, H]$. Выражение (1.7) является формулой линейного фильтра, коэффициенты a_n^m , $m \in [1, W]$, $n \in [1, H]$ называются импульсной характеристикой линейного фильтра. Для фильтрации Винера необходимо предположить, что x и y — это некоторые случайные величины, и выбрать импульсные характеристики a_m^n таким образом, что значение математического ожидания квадрата ошибки $\varepsilon = x - \hat{x}$ для оценки сигнала \hat{x} минимально:

$$\{a_n^m\} = \arg \min M\varepsilon^2 = \arg \min M\left(x - \sum a_n^m \cdot y_{i-n}^{j-m}\right)^2. \quad (1.8)$$

Пусть $B_{xy}(k, l) = M(x_j^i \cdot y_{j-l}^{i-k})$ и $B_y(k - m, l - n) = M(y_{j-l}^{i-k} \cdot y_{j-n}^{i-m})$, тогда $M\varepsilon^2 = M((x_j^i)^2) - 2 \sum a_n^m B_{xy}(m, n) + \sum a_n^m a_k^l B_y(k - m, l - n)$ — квадратичная функция относительно $\{a_j^i\}$, поэтому решение задачи (1.8) удовлетворяет уравнениям Винера-Хопфа, полученным после дифференцирования по коэффициентам a_j^i :

$$\begin{cases} B_{xy}(k, l) = \sum_{m, n} a_n^m B_y(k - m, l - n), \\ k \in [1, W], l \in [1, H]. \end{cases} \quad (1.9)$$

Для решения уравнений (1.9) применяется быстрое преобразование Фурье для двумерного случая (см. [71]). Поскольку правая часть уравнений является свёрткой, то

$$G_{xy}(i, j) = A_j^i \cdot G_y(i, j) \Rightarrow A_j^i = \frac{G_{xy}(i, j)}{G_y(i, j)}, \quad (1.10)$$

где A_j^i , G_{xy} , G_y — гармоники преобразования Фурье для, соответственно, a_j^i , $B_{xy}(i, j)$, $B_y(i, j)$. Поэтому, выполнив обратное преобразование Фурье для гармоник A_j^i , получим искомые характеристики $\{a_j^i\}$.

Обучаемые фильтры повышения разрешения изображений. Задача повышения разрешения изображения формулируется следующим образом: обладая изображением \tilde{F} с разрешением $W_F \times H_F$ некоторой сцены, необходимо получить изображение \tilde{S} с разрешением $W_S \times H_S$, намного превышающим изображение исходного кадра \tilde{F} . В данной формулировке у задачи не определены исходные данные, которых было бы достаточно для решения, поэтому вводятся дополнительные предположения о наблюдаемой сцене.

Одним из простых способов повысить разрешение является интерполяция с помощью двухмерных кубических сплайнов значений интенсивности кадра \tilde{S} в точках, в которые не попадают точки кадра \tilde{F} после масштабирования сетки $W_F \times H_F$ на сетку $W_S \times H_S$. Впервые для повышения разрешения подобную интерполяцию было предложено использовать в [63], в этой работе введён термин бикубическая интерполяция для данного метода обработки изображения.

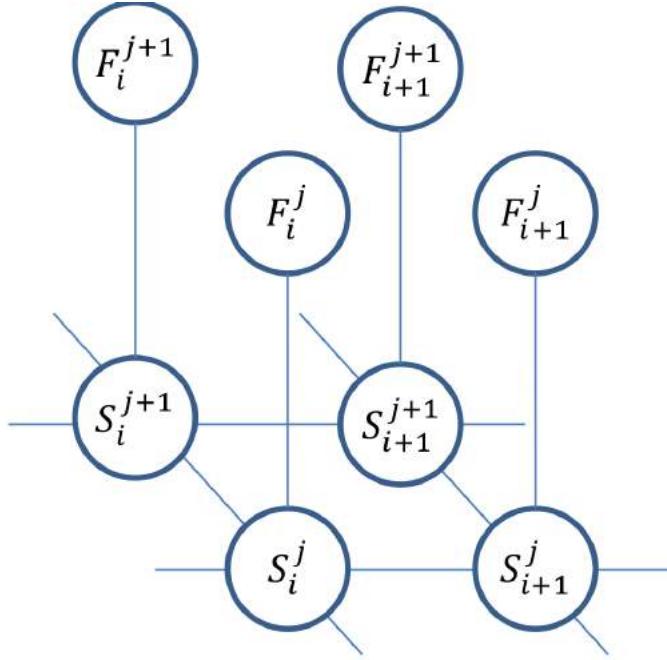


Рисунок 1.8. Пример подграфа марковского случайного поля, используемого для построения изображения с высоким разрешением на основе изображения с малым разрешением.

Недостатком бикубической интерполяции является то, что результирующее изображение не обладает высокой чёткостью текстур, данный метод сглаживает граничные переходы, что является свойством сплайнов. Существуют методы восстановления текстур высокого разрешения на основе заранее составленных вероятностных моделей. Для описания подобных моделей используется понятие марковского случайного поля, которое, в отличие от марковской цепи, представляет собой граф $G = (V, E)$, в котором вершины V являются случайными величинами, а рёбра E связывают зависимые случайные величины и не связывают независимые случайные величины. Понятие случайного марковского поля введено в работе [64], с его помощью которой обобщается математический аппарат, лежащий в основе моделей ряда физических процессов и явлений. Например, марковским случайным полем описывается двухмерная [65] и одномерная [65] модель Изинга взаимодействия магнитных диполей ферромагнетиков, предложенная физиком Вильямом Ленцем [67].

В работе [68] для повышения разрешения в четыре раза ($W_S = 2W_F$, $H_S = 2H_F$) в качестве элементов марковского случайного поля предлагается использовать отдельные небольшие части изображения \hat{F} , полученного из исходного изображения \tilde{F} методом повышения разрешения с помощью интерполяции, например, бикубической до разрешения $W_S \times H_S$, и искомого изображения \tilde{S} в форме векторов интенсивностей в квадратных окрестностях размера $N \times N$. Введём обозначения для данных векторов:

$$\begin{aligned}
S_i^j &= [\tilde{S}(i - \lfloor N/2 \rfloor, j - \lfloor N/2 \rfloor), \dots, \tilde{S}(i + \lceil N/2 \rceil - 1, j - \lfloor N/2 \rfloor), \\
&\quad \tilde{S}(i - \lfloor N/2 \rfloor, j - \lfloor N/2 \rfloor + 1), \dots, \tilde{S}(i + \lceil N/2 \rceil - 1, j - \lfloor N/2 \rfloor + 1), \\
&\quad \vdots \\
&\quad \tilde{S}(i - \lfloor N/2 \rfloor, j + \lceil N/2 \rceil - 1), \dots, \tilde{S}(i + \lceil N/2 \rceil - 1, j + \lceil N/2 \rceil - 1)], \\
F_i^j &= [\hat{F}(i - \lfloor N/2 \rfloor, j - \lfloor N/2 \rfloor), \dots, \hat{F}(i + \lceil N/2 \rceil - 1, j - \lfloor N/2 \rfloor), \\
&\quad \hat{F}(i - \lfloor N/2 \rfloor, j - \lfloor N/2 \rfloor + 1), \dots, \hat{F}(i + \lceil N/2 \rceil - 1, j - \lfloor N/2 \rfloor + 1), \\
&\quad \vdots \\
&\quad \hat{F}(i - \lfloor N/2 \rfloor, j + \lceil N/2 \rceil - 1), \dots, \hat{F}(i + \lceil N/2 \rceil - 1, j + \lceil N/2 \rceil - 1)]. \quad (1.11)
\end{aligned}$$

На рис. 1.8 показано, как между вершинами марковского случайного поля вводятся связи: между вершинами $\{S_i^j\}$ и $\{F_i^j\}$ связь существует тогда и только тогда, когда пары индексов i и j совпадают; между вершинами $\{F_i^j\}$ связь вводится тогда и только тогда, когда лишь один из индексов i или j отличается на 1.

В работе [69] показано, что общая вероятность состояния системы, изображенной на рис. 1.8, определяется произведением парных потенциалов, соответствующих кликам графа (кликой графа называется любой полный подграф исходного графа):

$$p(\{S_i^j\}, \{F_i^j\}) = \prod \Omega_i^j(S_i^j, F_i^j) \cdot \prod \Psi_i^j(F_i^j, F_{i+1}^j) \cdot \prod \Phi_i^j(F_i^j, F_i^{j+1}),$$

причём, как следует из [70], функции Ω_i^j , Ψ_i^j , Φ_i^j имеют форму потенциала Гибса $1/Z \cdot e^{-U(x,y)/T}$, где Z — нормировочный коэффициент, U — функция энергии, а T — условная температура, параметр, характеризующий состояние системы, а x , y — значения случайных величин марковского поля.

Наиболее востребованы методы повышения разрешения в современном цифровом телевидении, однако их можно использовать для повышения эффективности идентификации типов изображений.

Методы сопоставления изображений. Предположим, заданы два кадра F и \tilde{F} . На кадре \tilde{F} выделены точки изображения в прямоугольнике $\tilde{\mathcal{W}} = [i_0, i_{max}] \times [j_0, j_{max}]$, который принято называть «окном» ($i_0, i_{max} \in [1, W]$, $j_0, j_{max} \in [1, H]$). Для простоты изложения будем считать, что $i_{max} = i_0 + N - 1$ и $j_{max} = j_0 + N - 1$, то есть окно $\tilde{\mathcal{W}}$ — квадратное со стороной N точек.

Задача сопоставления изображений формулируется следующим образом: необходимо на кадре F найти такое квадратное окно $\mathcal{W} = [i, i + N - 1] \times [j, j + N - 1]$, $i \in [1, W - N + 1]$, $j \in [1, H - N + 1]$, в котором изображение наиболее похоже на изображение в окне $\tilde{\mathcal{W}}$ кадра \tilde{F} . В качестве меры близости изображений F и \tilde{F} в окнах \mathcal{W}

и $\tilde{\mathcal{W}}$ чаще всего используют либо сумму квадратов отклонений (квадрат евклидовой нормы)

$$S(x - y) = \sum (x_n^m - y_{n-j_0+j}^{m-i_0+i})^2, \quad (1.12)$$

либо корреляцию

$$\rho(x, y) = \frac{\sum ((x_n^m - \bar{x}) \cdot (y_{n-j_0+j}^{m-i_0+i} - \bar{y}))}{\sqrt{\sum (x_n^m - \bar{x})^2} \cdot \sqrt{\sum (y_{n-j_0+j}^{m-i_0+i} - \bar{y})^2}}. \quad (1.13)$$

В (1.12) и в (1.13) суммирование идёт по координатам $(m, n) \in \tilde{\mathcal{W}}$, обозначения сигналов x_j^i , y_j^i введены в формулах (1.6). Средние значения сигналов \bar{x} и \bar{y} для формулы (1.13) вычисляются, соответственно, в окнах $\tilde{\mathcal{W}}$ и \mathcal{W} .

Характеристика (1.12) более чувствительна к глобальным перепадам яркости на кадре, зато для задачи поиска

$$\min_{i,j \in [-M,M]} S(x - y) \quad (1.14)$$

«оптимального окна» \mathcal{W} в окрестности окна $\tilde{\mathcal{W}}$ (по любой из координат точки окна \mathcal{W} удалены от окна $\tilde{\mathcal{W}}$ не более, чем на M пикселов) существует алгоритм, позволяющий отыскать $\arg \min S(x - y)$ за $O(N^2 \log_2 N)$ операций [72]. Прямой перебор при решении задачи (1.14) возможен за $O(M^2 \cdot N^2)$ операций, что при значениях M , сравнимых со значениями N , существенно больше $O(N^2 \log_2 N)$.

Быстрый алгоритм решения задачи (1.14) основан на том, что

$$S(x - y) = \sum (x_n^m)^2 - 2 \sum (x_n^m \cdot y_{n-j_0+j}^{m-i_0+i}) + \sum (y_{n-j_0+j}^{m-i_0+i})^2,$$

где сумма $\sum (x_n^m)^2$ — постоянна относительно (i, j) , сумму $\sum (y_{n-j_0+j}^{m-i_0+i})^2$ мы предполагаем постоянной относительно (i, j) (данное предположение часто выполняется на практике), и поэтому, чтобы найти решение для (1.14), необходимо найти

$$\max_{i,j \in [-M,M]} \sum (x_n^m \cdot y_{n-j_0+j}^{m-i_0+i}). \quad (1.15)$$

Обозначим прямое и обратное преобразование Фурье, применяемое к значениям изображения в окне $\tilde{\mathcal{W}}$, как, соответственно, \mathcal{F} и \mathcal{F}^{-1} . Известно, что если обозначить $r_{ij} = \sum (x_n^m \cdot \tilde{y}_{n-j_0+j}^{m-i_0+i})$, где \tilde{y}_n^m — это периодическое продолжение y_n^m с квадрата $\tilde{\mathcal{W}}$ на всю целочисленную плоскость, то с помощью быстрого преобразования Фурье можно вычислить сразу все значения сумм попарных произведений за $O(N^2 \log_2 N)$ (см. [71]):

$$r_{ij} = \mathcal{F}^{-1}(\mathcal{F}(x) \cdot \overline{\mathcal{F}(y)}),$$

где $\overline{\mathcal{F}(y)}$ — операция комплексного сопряжения. При значениях M меньших, чем примерно $N/3$, можно пренебречь отличием r_{ij} от $\sum (x_n^m \cdot y_{n-j_0+j}^{m-i_0+i})$, поэтому задача поиска минимума (1.14) сводится к нахождению $\min r_{ij}$.

Развитие данного метода поиска соответствующих друг другу изображений можно увидеть в работах [73, 74]. В [73] вычисляется коэффициент корреляции $\rho(x, y)$ (см. формулу (1.13)), а в [74] — норма $S(x-y)$ по формуле (1.12), причём не делается никаких предположений, влияющих на точность решения, и в то же время используется быстрое преобразование Фурье для ускорения поиска максимума коэффициента корреляции при сравнении окон \mathcal{W} и $\tilde{\mathcal{W}}$ (идея применения преобразования Фурье схожа с идеей поиска решения для (1.14)). Асимптотика описанных алгоритмов $O((N+M)^2 \log_2(N+M))$.

В [75] описан наиболее быстрый из известных автору алгоритмов поиска минимума выражения $S(x-y)$ (задача (1.14)). Алгоритм имеет вычислительную сложность $O(N^2 \log_2 M)$, и основан на эвристическом подходе.

Чтобы найти вектор смещения (δ_x, δ_y) окна \mathcal{W} относительно $\tilde{\mathcal{W}}$, будем искать

$$\min_{\delta_x, \delta_y \in [-M, M]} \sum_{(i,j) \in \tilde{\mathcal{W}}} \left(F(i + \delta_x, j + \delta_y) - \tilde{F}(i, j) \right)^2, \quad (1.16)$$

что эквивалентно (1.12)). Разности $F(i + \delta_x, j + \delta_y) - \tilde{F}(i, j)$ можно приблизить дифференциалом

$$F(i + \delta_x, j + \delta_y) - \tilde{F}(i, j) = F(i, j) - \tilde{F}(i, j) + \frac{\partial F}{\partial x}(i, j) \cdot \delta_x + \frac{\partial F}{\partial y}(i, j) \cdot \delta_y,$$

где аппроксимация производных вычисляется по формулам

$$\frac{\partial F}{\partial x}(i, j) = \frac{F(i+1, j) - F(i, j)}{1}, \quad \frac{\partial F}{\partial y}(i, j) = \frac{F(i, j+1) - F(i, j)}{1}.$$

Перейдя к обозначениям (1.6), получим, что (1.16) в дифференциальном приближении записывается как

$$\min \sum \left(y_j^i - x_j^i + (y_j^{i+1} - y_j^i) \cdot \delta_x + (y_{j+1}^i - y_j^i) \cdot \delta_y \right)^2,$$

откуда, после дифференцирования и решения системы линейных уравнений из двух неизвестных, можно получить решение

$$\begin{cases} \delta_x = \frac{\sum ((x_j^i - y_j^i)(y_j^{i+1} - y_j^i)) \sum (y_{j+1}^i - y_j^i)^2 - \sum ((y_j^{i+1} - y_j^i)(y_{j+1}^i - y_j^i)) \sum ((x_j^i - y_j^i)(y_{j+1}^i - y_j^i))}{(\sum (y_j^{i+1} - y_j^i)^2)(\sum (y_{j+1}^i - y_j^i)^2) - \sum ((y_j^{i+1} - y_j^i)(y_{j+1}^i - y_j^i))}, \\ \delta_y = \frac{\sum ((x_j^i - y_j^i)(y_{j+1}^i - y_j^i)) \sum (y_j^{i+1} - y_j^i)^2 - \sum ((y_j^{i+1} - y_j^i)(y_{j+1}^i - y_j^i)) \sum ((x_j^i - y_j^i)(y_{j+1}^i - y_j^i))}{(\sum (y_j^{i+1} - y_j^i)^2)(\sum (y_{j+1}^i - y_j^i)^2) - \sum ((y_j^{i+1} - y_j^i)(y_{j+1}^i - y_j^i))}. \end{cases} \quad (1.17)$$

Поиск смещения с помощью дифференциального приближения уместен лишь при соблюдении условий $|\delta_x| < 1$, $|\delta_y| < 1$. Если требуется найти вектор смещения $(\delta_x, \delta_y) \in [-M, M]^2$, то необходимо использовать итерационный алгоритм поиска, заключающийся в нижеперечисленных шагах.

1. Из изображений F и \tilde{F} формируем конечную последовательность изображений $\{F_i\}_{i=0}^K$ и $\{\tilde{F}_i\}_{i=0}^K$, где $K = \lceil \log_2 M \rceil$ (округление вверх). Последовательность формируем по следующим рекуррентным соотношениям:

$$F_0 = F, \quad \tilde{F}_0 = \tilde{F}, \quad F_k = \mathbf{r}(F_{k-1}), \quad \tilde{F}_k = \mathbf{r}(\tilde{F}_{k-1}), \quad k = 1, \dots, K,$$

где \mathbf{r} является оператором понижения разрешения кадра, функцию кадра $F(x, y)$, определённую на прямоугольнике $[1, W] \times [1, H]$, \mathbf{r} переводит в функцию $F'(x, y)$, определённую на $[1, \lfloor W/2 \rfloor] \times [1, \lfloor H/2 \rfloor]$, такую, что

$$F'(x, y) = \frac{F(2x, 2y) + F(2x+1, 2y) + F(2x, 2y+1) + F(2x+1, 2y+1)}{4}.$$

2. Пусть окно

$$\tilde{\mathcal{W}}_K = [\lfloor i_0/2^K \rfloor, \lfloor (i_0 + N)/2^K \rfloor] \times [\lfloor j_0/2^K \rfloor, \lfloor (j_0 + N)/2^K \rfloor].$$

Данное окно соответствует исходному окну $\tilde{\mathcal{W}}$ на кадре пониженного разрешения \tilde{F}_K . Найдем смещение (δ_x^K, δ_y^K) окна \mathcal{W}_K на кадре F_K относительно окна $\tilde{\mathcal{W}}_K$ кадра \tilde{F}_K , воспользовавшись формулами (1.17). Использование этих формул правомерно, поскольку благодаря снижению размерности смещения по обоим координатам не может превышать единицы, если искомое исходное смещение действительно меньше M . Обозначим

$$i_K = i_0/2^K + \delta_x^K, \quad j_K = j_0/2^K + \delta_y^K.$$

Отметим, что i_K и j_K не обязательно целые.

3. Последовательно для каждого $k = K - 1, K - 2, \dots, 1$ выполним следующую операцию: для окна

$$\tilde{\mathcal{W}}_k = [\lfloor 2i_{k+1} \rfloor, \lfloor 2i_{k+1} \rfloor + \lfloor N/2^k \rfloor] \times [\lfloor 2j_{k+1} \rfloor, \lfloor 2j_{k+1} \rfloor + \lfloor N/2^k \rfloor]$$

на кадре \tilde{F}_k найдём смещение (δ_x^k, δ_y^k) соответствующего ему окна \mathcal{W}_k на кадре F_k по формулам (1.17) и положим $i_k = 2i_{k+1} + \delta_x^k$, $j_k = 2j_{k+1} + \delta_y^k$. За счёт итерационного «восхождения» от пониженного разрешения к более высокому и из-за поправок к положению окон $\tilde{\mathcal{W}}_k$, смещение на каждой итерации не должно превышать единицы по любой из двух координат.

4. Окончательно, $(\delta_x, \delta_y) = (i_0 - 2i_1 - \delta_x^1, j_0 - 2j_1 - \delta_y^1)$ — смещение окна \mathcal{W} относительно $\tilde{\mathcal{W}}$, причём результат не обязательно целый, то есть данный алгоритм может потенциально выделять смещения меньшие, чем одна точка изображения.

Операция снижения размерности имеет сложность $O((M + N)^2)$ для требуемой области кадра. Сложность операции расчёта смещения по формуле (1.17) оценивается как $O(N^2)$. Поскольку эта операция выполняется $\lceil \log_2 M \rceil$, то сложность всего алгоритма оценивается как $O(N^2 \log_2 M)$ при достаточно больших M .

Необходимость сопоставления изображений возникает во многих приложениях: при вычислении оптического потока, при необходимости компенсировать глобальное движение в кадре при плохом креплении видеокамеры к опоре и нестабильном изображении, при попытке отследить перемещение объекта, который показывается достаточно крупным планом и в других задачах. Кроме параллельного переноса, иногда возникает необходимость отслеживать более сложное перемещение изображений. Для такой задачи тоже разработан ряд решений, см., например, [76].

Выделение движущихся объектов из видеосигнала. Как правило, если изображение с видеокамеры статично, то никаких событий, которые могут заинтересовать оператора системы видеонаблюдения, в области зрения рассматриваемой видеокамеры не происходит. Однако простые алгоритмы определения наличия движения в кадре не всегда обеспечивают применимость средств видеонаблюдения в некоторых приложениях, поэтому огромное количество работ посвящено детализации анализа изменений в кадре видеосигнала.

Целый класс алгоритмов разработан для того, чтобы отслеживать перемещение объектов на изображении с помощью выделения различных характеристик, присущих лишь данному объекту: форма контура, цветовые и текстурные параметры. В работе [16] предложен алгоритм выделения перемещения объектов на изображении видеокамеры при помощи сопоставления гистограмм яркости точек объекта на различных кадрах последовательности видеоизображений.

Для анализа изображений сцен с насыщенным движением (скопление людей, оживлённая автомобильная трасса) разработаны алгоритмы выделения, группировки и сопоставления на соседних кадрах видеопоследовательности специальных особых точек движущихся объектов, являющихся угловыми точками контуров [17, 18].

В работе [19] предложен способ выделения движения автомобилей, основанный на сопоставлении и поиске трёхмерной модели контуров автомобилей среди выделенных двумерных контуров кадра.

С точки зрения классификации выделенных объектов наиболее подходящий алгоритм выделения движущихся объектов описан в [38]. Предложенный способ выделения движущихся объектов называется методом адаптивного вычитания фона. Чтобы получить характеристическую функцию движения точек $\omega_i(x, y)$ для кадра F_i последовательности $\{F_i\}$ ($\omega_i(x, y) = 0$, если точка (x, y) движется, и $\omega_i(x, y) = 1$, если точка (x, y) не движется), используются следующие рекуррентные формулы:

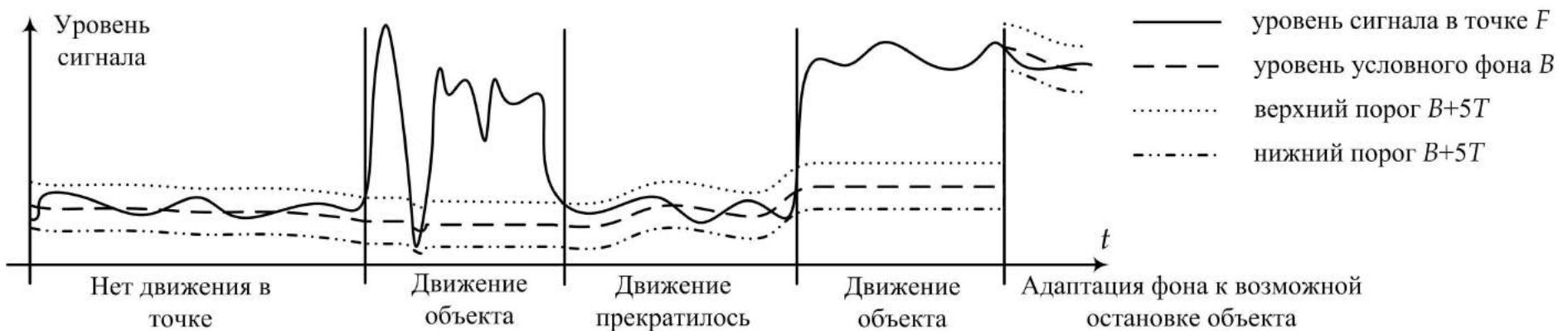
$$\omega_1 = 0, \quad B_1(x, y) = F_1, \quad T_1(x, y) = 7,$$

$$\begin{aligned}
B_i(x, y) &= \begin{cases} \alpha F_i(x, y) + (1 - \alpha) B_{i-1}(x, y), & \omega_{i-1}(x, y) = 0, \\ B_{i-1}(x, y), & \omega_{i-1}(x, y) = 1, \end{cases} \\
T_i(x, y) &= \begin{cases} \alpha |B_i(x, y) - F_i(x, y)| + (1 - \alpha) T_{i-1}(x, y), & \omega_{i-1}(x, y) = 0, \\ T_{i-1}(x, y), & \omega_{i-1}(x, y) = 1, \end{cases} \\
\omega_i(x, y) &= \begin{cases} 1, & \text{если } \|F_i(x, y) - B_i(x, y)\|_\infty > 5 \cdot T_i \\ 0 & \text{в противном случае,} \end{cases} \tag{1.18}
\end{aligned}$$

где B_i и T_i — оценка, соответственно, математического ожидания (фона) и дисперсии точек изображения на i -м кадре последовательности, α — коэффициент обновления (обычно $0.01 < \alpha < 0.1$), под нормой $\|\cdot\|_\infty$ понимается максимальное из трёх абсолютных значений разности для трёх компонент цветности. Разбив множество движущихся точек на i -м кадре, соответствующих функции ω_i , на непересекающиеся части без соседних точек, получим подмножество точек, соответствующих различным движущимся изображениям объектов. Каждая область является каким-то объектом, который, вероятно, присутствовал на предыдущих кадрах последовательности изображений. Задача сопоставления новых областей уже существующим и выделенным объектам сводится к проблеме поиска оптимальных паросочетаний. На рис. 1.9 проиллюстрированы основные этапы работы алгоритма выделения движущихся изображений.

Выделение движущихся объектов с помощью формул (1.18) не устойчиво по отношению к следующим помехам: движение теней от объектов, качание веток деревьев, волновые колебания поверхности воды, резкое изменение освещенности сцены из-за переменной облачности. В некоторых работах метод адаптивного вычитания фона развивается с целью уменьшить влияние данных помех. Например, в [20, 77] предлагается использовать модель суммы нескольких нормальных распределений для фона в одной точке, тем самым уменьшается влияние перепадов яркости, связанных с переменной облачностью и некоторыми помехами от отброшенных объектами теней. В [78] используется модель Ламберта для отражения света, чтобы устраниТЬ влияние перепадов яркости от теней объектов и облаков. Анализ цветового состава точек для отделения теней и объектов также приводит к фильтрации помех, связанных с изменением освещённости сцены и устраняет искажение силуэтов объектов [79, 80, 81]. Наконец, в [82] достигается фильтрация шумов от движения с колебательным характером (ветки деревьев, их тени, волны на воде) за счёт применения формул (1.18) не к исходному изображению, а к полю векторов смещения каждой точки, рассчитанному итерационно по формуле (1.17).

Таким образом, ряд методов позволяет с некоторой точностью получить исходные данные о движущихся объектах в области зрения камеры. Эти исходные данные могут



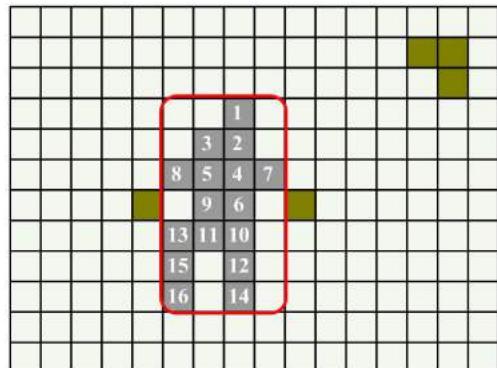
$$\omega_1 = 0, \quad B_1(x, y) = F_1, \quad T_1(x, y) = 7,$$

$$T_i(x, y) = \begin{cases} \alpha |B_i(x, y) - F_i(x, y)| + (1 - \alpha)T_{i-1}(x, y), & \omega_{i-1}(x, y) = 0, \\ T_{i-1}(x, y), & \omega_{i-1}(x, y) = 1, \end{cases}$$

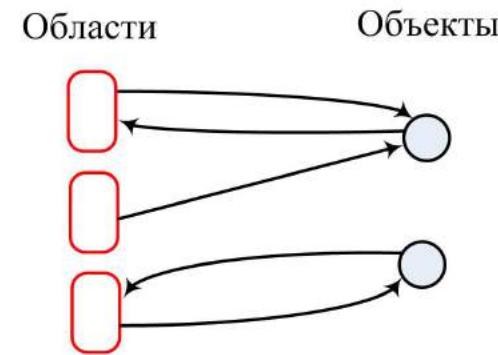
$$B_i(x, y) = \begin{cases} \alpha F_i(x, y) + (1 - \alpha)B_{i-1}(x, y), & \omega_{i-1}(x, y) = 0, \\ B_{i-1}(x, y), & \omega_{i-1}(x, y) = 1, \end{cases}$$

$$\omega_i(x, y) = \begin{cases} 1, & \text{если } \|F_i(x, y) - B_i(x, y)\|_\infty > 5 \cdot T_i \\ 0 & \text{в противном случае} \end{cases}$$

а) выделение движущейся точки изображения (алгоритм вычитания фона)



б) выделение движущихся областей



в) сопоставление объектов и областей

Рисунок 1.9. Иллюстрация работы алгоритмов выделения движущихся изображений на основе алгоритма вычитания фона.

использоваться при классификации и анализе событий в аналитических видеосистемах.

Методы сопряжения видеокамер. Одна из моделей зрения, как технического, так и биологического, представлена в работе [83]. С помощью данной модели можно ввести сопряжение различных ракурсов наблюдения видеокамер. Видеокамеры могут быть сопряжены как с физическим пространством, и тогда для каждой точки изображения существуют оценочные значения трёхмерных координат, так и друг с другом, тогда координатам каждой точки кадра одной камеры соответствуют какие-то координаты изображения с сопряжённой камеры, и наоборот.

Например, в работе [84] используется сопоставление изображения видеокамеры с физическими трёхмерными координатами для улучшения точности предсказания положения на последующих кадрах движущихся объектов, выделенных в текущем кадре. Расчёт коэффициентов проективного преобразования проводится на основе предположения, что координаты четырёх точек физического пространства наблюдаемой сцены нам известны.

В [85] для решения аналогичной задачи используются сложные цифровые карты поверхности земли с указанием перепадов высот, и угловые координаты оптической оси видеокамер в совокупности с тремя координатами.

Автоматическое сопряжение видеокамер между собой представлено в [86]: сопряжение изображений производится на основе предварительной калибровки камер, а также на основе сопоставления координат изображений движущихся объектов в случае, если в области зрения сопрягаемых видеокамерах движется только один объект. Схожий метод автоматического сопряжения представлен в [87]. Сопряжение видеокамер основано на выделении прямых-границ изображения общей зоны зрения. Полученные результаты используются для согласованного выделения движущихся объектов в общей зоне зрения видеокамер.

В работах [88, 89, 90] представлено использование сопряжения трёх и более видеокамер для получения трёхмерных координат объектов в приложении к видеонаблюдению во внутренних помещениях. После оценки трёхмерные координаты переводятся простейшим параллельным переносом на план помещений.

Системы автоматического управления поворотными видеокамерами. Большинство поворотных устройств, на которые устанавливаются видеокамеры, предназначены для выполнения простейших функций управления: повернуть оптическую ось камеры с помощью ручного манипулятора, уменьшить или увеличить угол обзора, сфокусировать изображение. Некоторые поворотные видеокамеры позволяют запоминать текущее состояние параметров поворотного устройства, чтобы затем вернуться к этому состоянию по определённой команде управления.

Идея автоматизированного управления поворотными видеокамерами описана, например, в [91]. Однако публикаций с описанием систем, использующих подобную идеологию, не очень много.

Известна система из трёх поворотных камер, установленных в разных точках лабораторного помещения, которая отслеживает перемещения вошедшего в лабораторию человека, причём все три камеры согласованы между собой таким образом, чтобы оптические оси по возможности пересекались в одной точке. Поэтому, все три камеры следят за перемещением человека, наводя на него оптическую ось, даже в том случае, когда одна из видеокамер «не видит» объект[92]. В работе описан алгоритм слежения за объектом наблюдения (на первом этапе он основан на [20], в процессе движения камеры слежение происходит по алгоритму [16]), но не рассмотрены вопросы точности и задержек управления.

Несмотря на отсутствие подробного описания в открытой литературе, функция автоматического наведения поворотных видеокамер на объекты наблюдения — одна из важных функций аналитических видеосистем, что следует из проспектов [55, 57] и подобных материалов.

В завершение обзора методов обработки цифровых видеосигналов, приведём структурную схему предложенной классификации, см. рис. 1.10.

1.4. Алгоритмы распознавания и идентификации видеоизображений

Классификация (распознавание) выделенных объектов — одна из основных особенностей аналитических видеосистем. В работах [15, 38] предложено использовать классификацию для определения категорий движущихся объектов («человек», «группа людей», «седан», «пикап», «грузовик», «автобус») и цвета, при этом использовались известные алгоритмы классификации. Рассмотрим основные методы распознавания, которые предлагаются в литературе для классификации.

Основные понятия и определения. Прежде, чем перейти к обзору методов распознавания, введём необходимые для этого определения и обозначения. В целом вводимые понятия соответствуют аналогичной терминологии в работах [29, 47, 94].

Пусть дано множество объектов M , называемое множеством допустимых объектов (здесь и далее, если специально не оговорено противное, речь идёт о конечных множествах, которые, возможно, содержат большое количество элементов). Множество M разбито на c подмножеств K_1, \dots, K_c ($M = \bigcup_{i=1}^c K_i$), называемых классами. Для любого объекта S , $S \in M$, определена некоторая информация $I(S)$. Будем называть информацию $I(S)$ стандартной, если $I(S) = (x_1, \dots, x_n)$ — вектор с вещественными компонентами, которые мы назовём признаками объектов, соответственно n -мерное вещественное пространство R^n в этом случае будем называть пространством признаков. Также для каждого объекта $S \in M$ может быть определён информационный

Методы обработки видеосигналов

<p>Предварительная обработка:</p> <ul style="list-style-type: none"> – выделение границ[10]; – выравнивание контрастности[6]; – фильтрация Винера-Хопфа[48, 6]. 	<p>Сравнение изображений:</p> <ul style="list-style-type: none"> – поиск максимальной корреляции[71, 73]; – минимизация Евклидовой нормы[74]; – итерационный поиск[75]; – поиск поворотов и увеличения[76]. 	<p>Выделение движения:</p> <ul style="list-style-type: none"> – сравнение гистограмм яркости областей[16]; – выделение перемещающихся особых точек[17, 18]; – моделирование и поиск движущихся форм на изображениях[19]; – рекурсивная оценка параметров движения в точке[15]; – подавление помех[20, 77, 78, 79, 80, 81, 82].
<p>Управление видеокамерами:</p> <ul style="list-style-type: none"> – наведение трёх поворотных видеокамер в одну точку[92]; – видеосистемы с синхронизацией и сопряжением поворотных и неподвижных видеокамер[55, 57]. 	<p>Сопряжение изображений:</p> <ul style="list-style-type: none"> – сопоставление с физическим пространством[84]; – сопоставление с цифровой картой высот[85]; – сопряжение изображений видеокамер[86, 87]. 	<p>Классификация изображений:</p> <p>см. таблицу 1.1</p>

Рисунок 1.10. Классификация методов обработки видеосигналов.

вектор $\alpha(S) = (\alpha_1, \dots, \alpha_c)$, состоящий из c компонент. Компоненты α_j принадлежат множеству $\{0, \Delta, 1\}$, значению $\alpha_j = 1$ соответствует предикат $P_j(S) \equiv S \in K_j$, значению $\alpha_j = 0$ — предикат-отрицание $\overline{P_j(S)} \equiv S \notin K_j$, $\alpha_j = \Delta$ означает, что неизвестно, выполняется предикат $P_j(S)$ или нет. Информационный вектор $\tilde{\alpha}(S)$, состоящий только из нулей и единиц, назовём истинным, если все соответствующие ему предикаты выполняются.

Выделим из множества M подмножество L , и обозначим $K'_i = L \cap K_i$ ($L = \bigcup_{i=1}^c K'_i$). Мы будем считать, что разбиение L на классы K'_i , $i \in \overline{1, c}$, нам известно, т.е. для каждого $S \in L$ нам известен истинный информационный вектор $\tilde{\alpha}(S)$. Назовём множество L обучающим. Далее мы будем считать, что обучающее множество L состоит из l элементов S_1, \dots, S_l , называемых обучающими примерами. i -ый компонент истинного информационного вектора $\tilde{\alpha}(S_j)$ обозначим как $\tilde{\alpha}_{ij}$.

Информацию $I(S)$ мы можем определить на некотором множестве $\mathcal{M} \subset M$ при помощи простого перечисления: $I(\mathcal{M}) = (I(S_1), \dots, I(S_k))$, где $\{S_1, \dots, S_k\} = \mathcal{M}$. Опре-

делим также $I(\mathcal{M}_1, \dots, \mathcal{M}_s)$ как перечисление $(I(\mathcal{M}_1), \dots, I(\mathcal{M}_s))$ для подмножеств $\mathcal{M}_i \subset M$, $i \in \overline{1, s}$. Информацию о классах $I_0(K'_1, \dots, K'_c)$ мы назовём обучающим информационным набором. Подчеркнём, что смысл перечисления I_0 отличается от смысла перечисления $I(L)$, поскольку в последнем не присутствует информация о разбиении объектов на классы.

Определив объекты $I_0(K'_1, \dots, K'_c)$, и $I(S)$, $S \in M$, мы можем ввести понятие алгоритма распознавания A :

$$A(I_0(K'_1, \dots, K'_c), I(S)) = \alpha^A(S), \quad (1.19)$$

где $\alpha^A(S)$ — некоторый информационный вектор, который алгоритм A сопоставляет объекту S , обладая информацией I_0 и $I(S)$. Задача распознавания заключается в том, чтобы построить такой алгоритм A , что для всех объектов $S \in M$ истинные информационные векторы $\tilde{\alpha}(S)$ практически не отличаются от векторов $\alpha^A(S)$, которые выдаёт алгоритм. Конкретные способы построения алгоритмов распознавания весьма разнообразны и описаны во множестве работ, например, в [93, 25, 28, 29, 47, 94, 95, 30, 96, 97, 32]. Далее рассмотрены наиболее часто применяемые способы.

Квадратичный дискриминантный анализ. Условимся считать, что вектор $I(S) = (x_1, \dots, x_n) = x$ является случайной величиной. Предположим также, что внутри каждого класса K_i признаки объектов распределены по нормальному закону, т.е. если $I(S) = x$, то $x \in N(m_i, R_i)$ при условии, что $S \in K_i$ (обозначим плотность данного распределения $\omega_i(x)$). Этой информации достаточно, чтобы посчитать по формуле Байеса условную вероятность для объекта S ($I(S) = x$) принадлежать классу K_i :

$$P(S \in K_i | I(S) = x) = \frac{P(I(S) = x | S \in K_i) P(S \in K_i)}{P(I(S) = x)}. \quad (1.20)$$

Если событие $I(S) = x$ интерпретировать как событие $x \in \Omega$, где Ω — некоторая малая окрестность признака x , то мы получим, что $P(I(S) = x | S \in K_i) = \int_{\Omega} \omega_i(x) dx \approx \omega_i(x) \mu \Omega$, где $\mu \Omega$ — мера множества Ω . Формула (1.20) преобразуется к виду

$$P(S \in K_i | I(S) = x) = \omega_i(x) \cdot \frac{P(S \in K_i) \mu \Omega}{P(I(S) = x)}. \quad (1.21)$$

Допустим, что объекты каждого класса встречаются с одинаковой частотой, тогда $P(S \in K_i)$ не зависит от класса K_i , и значит, согласно формуле (1.21) вероятность $P(S \in K_i | I(S) = x)$ максимальна для того класса K_i , у которого плотность распределения ω_i больше всех в точке x .

Плотность распределения $\omega_i(x)$ вычисляется по формуле

$$\omega_i(x) = \frac{1}{(2\pi)^{n/2} (\det R)^{1/2}} \exp \left(-\frac{1}{2} (x - m_i)^T R^{-1} (x - m_i) \right). \quad (1.22)$$

Алгоритм $A(I_0, I(S))$ строится следующим образом: i -ая компонента вектора $\alpha^A(S)$, соответствующая максимуму $\max_{i=1, \dots, c} \omega_i(I(S))$, устанавливается равной единице, а все остальные компоненты — нулю. Поскольку значения m_i , R_i не известны изначально, необходимо их оценить, исходя из информации $I_0(K'_1, \dots, K'_c)$:

$$m_i = \frac{1}{|K'_i|} \sum_{S \in K'_i} I(S), \quad (1.23)$$

$$R_i = \frac{1}{|K'_i| - 1} \sum_{S \in K'_i} (I(S) - m_i)(I(S) - m_i)^T, \quad (1.24)$$

где $|K'_i|$ — количество элементов в множестве K'_i .

Формулы (1.23) и (1.24) соответствуют подготовке начальных данных для алгоритма A , иначе говоря, обучению алгоритма. На этапе обучения требуется выполнить порядка $O(l)$ операций (напомним, что $l = |L|$). После подготовки начальных данных в основном процессе работы алгоритму распознавания A требуется всего $O(c)$ операций, чтобы вычислить $\alpha^A(S)$ на основании формул (1.22).

Линейный дискриминантный анализ. Линейный дискриминантный анализ не является способом построения алгоритма распознавания A . Суть линейного дискриминантного анализа — уменьшить размерность пространства признаков R^n при помощи линейной проекции в пространство R^m . Собственно, любое преобразование вида

$$y = V^T x, \quad (1.25)$$

где V — матрица размерности $n \times m$, уменьшает размерность пространства признаков с n до m , нас будут интересовать такие матрицы V , при которых преобразование (1.25) оптимально в определённом смысле.

Введём матрицу разброса внутри классов

$$W = \sum_i^c \sum_{S \in K'_i} (I(S) - m_i)(I(S) - m_i)^T, \quad (1.26)$$

где m_i считается по формуле (1.23), а также матрицу разброса между классами

$$B = \sum_i^c |K'_i| (m_i - m)(m_i - m)^T, \quad (1.27)$$

где $m = \frac{1}{l} \sum_{S \in L} I(S) = \frac{1}{l} \sum_{i=1}^c |K'_i| m_i$. Матрицы W и B задают эллипсоиды разброса внутри классов и между классами, соответственно. После проекции при помощи матрицы V эти эллипсоиды перейдут, соответственно, в $V^T W V$ и $V^T B V$. Отношение произведений квадратов полуосей этих эллипсоидов есть отношение определителей

$$J(V) = \frac{\det V^T B V}{\det V^T W V}, \quad (1.28)$$

поскольку определитель является произведением собственных значений матрицы в данном случае. Нас будут интересовать матрицы V , доставляющие максимум функционалу (1.28). Согласно [30, 98], столбцы оптимальной матрицы V соответствуют максимальным собственным значениям уравнения

$$(B - \lambda_i W)v_i = 0. \quad (1.29)$$

Поиск решений для (1.29) описывается следующим алгоритмом:

1. С помощью процесса ортогонализации Грамма – Шмидта находим матрицу Q такую, что $Q^T W Q = E$. Домножив уравнение (1.29) на матрицу Q^T слева, и сделав замену переменных $v_i = Qv'_i$, получим равносильное уравнение

$$(Q^T B Q - \lambda_i E)v'_i = 0.$$

2. Находим собственные значения λ_i и собственные векторы v'_i для последнего уравнения при помощи метода вращений.

3. Производим обратную замену по формулам $v_i = Qv'_i$ и составляем матрицу V .

В силу свойств матриц W и B , которые следуют из способа их вычисления (1.26) и (1.27), размерность нового пространства для признаков y не может превышать $c - 1$, т.к. уравнение (1.29) имеет не более $c - 1$ собственных решений. Когда классификация происходит по небольшому количеству классов, это может привести к потере исходной информации при применении линейного дискриминантного анализа.

Сокращение размерности по описанному методу добавляет к основному процессу обучения $O(l)$ операций при подсчете по формулам (1.26) и (1.27), а также $O(n^2)$ операций (в среднем) при решении уравнения (1.29). В процессе работы при распознавании добавляется незначительная операция расчёта подготовленного преобразования (1.25).

Правило k ближайших соседей. Обозначим за $V(S)$ выборку из k ближайших по отношению к объекту S в признаковом пространстве элементов множества $L \setminus S$, т.е.

$$V(S) = (S_1, \dots, S_k), \text{ где } S_j = \arg \min_{S' \in L \setminus \{S, S_1, \dots, S_{j-1}\}} \rho(S, S'),$$

$\rho(S, S')$ — расстояние между $I(S)$ и $I(S')$ в признаковом пространстве R^n .

Алгоритм $A(I_0, I(S))$ определяет, что $S \in K_i$ (иначе говоря, результат алгоритма $\alpha_{ij}^A(S) = \delta_{ij}$, где δ_{ij} — символ Кронекера), если в выборке $V(S)$ наибольшее число элементов из класса K_i . В случае, если максимум количества элементов в выборке $V(S)$ достигается для нескольких классов одновременно, алгоритм A выдаёт неопределённый результат (Δ, \dots, Δ) .

Если следовать указанной схеме напрямую, то обучение у данного алгоритма вообще отсутствует, а при работе требуется $O(k \cdot l)$ операций, чтобы выдать результат

распознавания, что очень расточительно. Можно использовать так называемое kD -дерево при поиске выборки $V(S)$. Тогда на стадии обучения потребуется $O(l)$ операций для создания kD -дерева, и при распознавании каждого объекта будет выполняться в среднем $O(k \log_2 l)$ операций. В наихудшем случае время поиска выборки $V(S)$ с помощью kD -дерева увеличивается до $O(k \cdot l^{2/3})$ операций.

Нейросеть. Нейросетевым алгоритмам посвящено много литературы. В частности, в [99, 25, 31] очень подробно рассмотрены вопросы обучения и применения нейросетей. В данном разделе нейросетевой подход описывается лишь с позиции применения нейросетей в качестве алгоритма распознавания в аналитических видеосистемах.

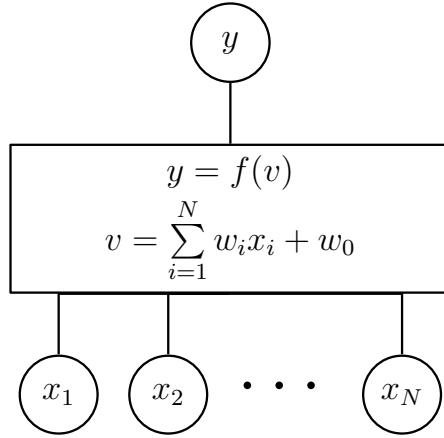


Рисунок 1.11. Схема одного нейрона.

Нейроном является функция вида $y = f(\sum_{i=1}^N w_i x_i + w_0)$. Функция $f(v)$ называется пороговой, а коэффициенты w_0, w_1, \dots, w_N — весами нейрона. Аргументы $x_i, i \in \overline{1, N}$, называются входами нейрона, а значение y — выходом. На рис. 1.11 представлено схематическое изображение нейрона.

Нейросеть — это граф специального вида, вершинами которого являются нейроны. Предполагается, что веса у всех нейронов разные, а пороговая функция одинакова. Нейросеть разбивается на несколько уровней, занумерованных по порядку, начиная с единицы. Каждый нейрон принадлежит одному из уровней. На вход нейросети подается N -мерный вектор, который является входом для всех нейронов первого уровня. Выходы каждого уровня, кроме последнего, являются входами для нейронов следующего уровня. Выходы последнего уровня называются выходом нейросети, который является вектором с размерностью, равной количеству нейронов на последнем уровне. Т.о., если x — вектор входов, y — вектор выходов, то описанная выше нейросеть однозначно соответствует функции $y = Y(x, \bar{w})$, где \bar{w} — набор весов всех нейронов, входящих в нейросеть, а Y — довольно сложная композиция линейных функций с коэффициентами \bar{w} и пороговой функцией f .

Пример нейросети с тремя уровнями, тремя входами и двумя выходами схема-

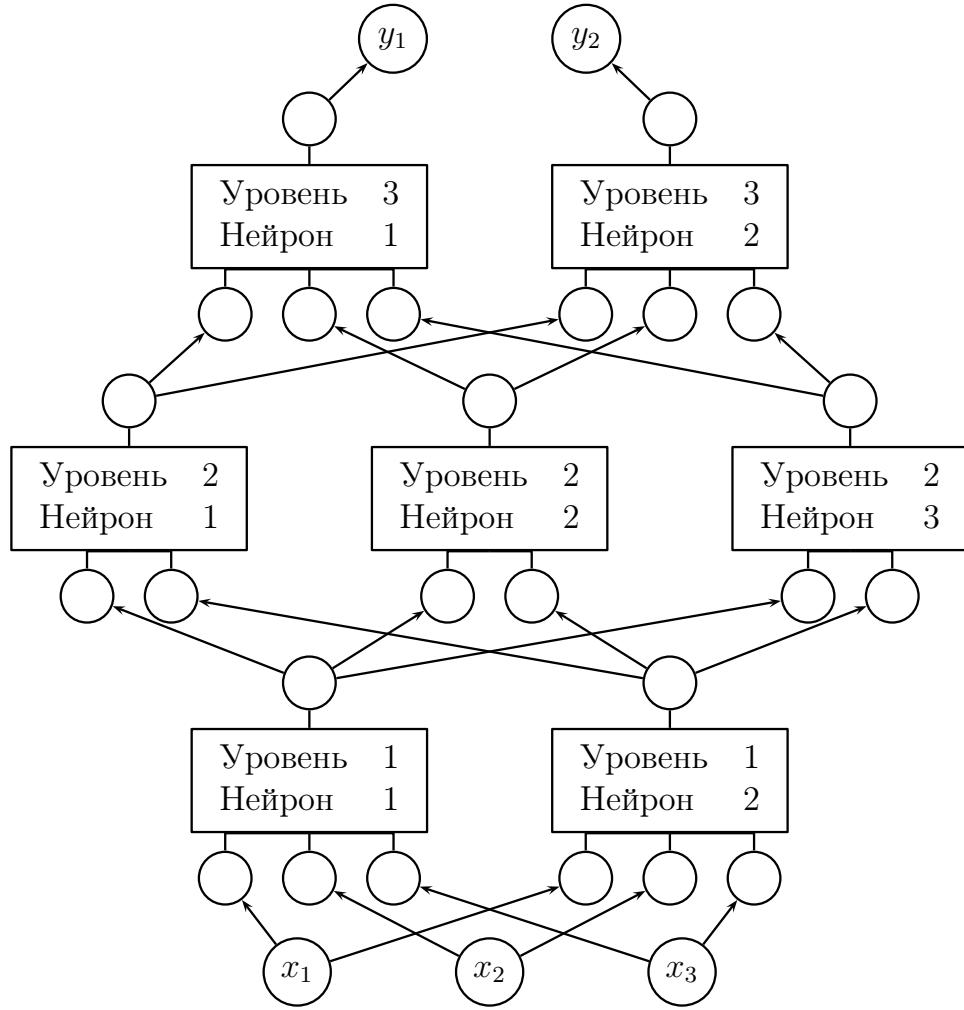


Рисунок 1.12. Пример нейросети с тремя уровнями.

тически представлен на рис. 1.12. В аналитическом виде формулы такой нейросети записываются следующим образом:

$$y_1 = f\left(\sum_{k=1}^3 w_k^{31} f\left(\sum_{j=1}^2 w_j^{2k} f\left(\sum_{i=1}^3 w_i^{1j} x_i + w_0^{1j}\right) + w_0^{2k}\right) + w_0^{31}\right),$$

$$y_2 = f\left(\sum_{k=1}^3 w_k^{32} f\left(\sum_{j=1}^2 w_j^{2k} f\left(\sum_{i=1}^3 w_i^{1j} x_i + w_0^{1j}\right) + w_0^{2k}\right) + w_0^{32}\right),$$

где \$w_i^{jk}\$ — \$i\$-ый вес \$k\$-го нейрона на \$j\$-ом уровне.

Рассмотрим пороговую функцию

$$f(v) = \begin{cases} 1, & \text{если } v \geq 0, \\ 0, & \text{если } v < 0. \end{cases} \quad (1.30)$$

Область значений функции \$Y(x, \bar{w})\$ нейросети с пороговой функцией (1.30) представляет из себя конечный набор векторов, компоненты которых равны 0 или 1. Прообраз каждого из этих векторов будет множеством, составленным из комбинации объединений и пересечений полупространств, границы которых задают нейроны первого

уровня. Меняя конфигурацию нейросети и веса нейронов, можно добиться такого разбиения пространства признаков, что значение алгоритма $A(I_0, I(S)) = \alpha^A(S)$ будет значением нейросети на признаках $x = I(S)$. Чтобы построить такую нейросеть или, иначе говоря, обучить нейросетевой алгоритм, требуется выполнить нижеперечисленные действия.

Сначала выбирается конфигурация нейросети, т.е. количество уровней и количество нейронов на каждом уровне. Фиксируется лишь число входов на нижнем уровне, которое соответствует количеству признаков в стандартной информации $I(S)$, и число нейронов на верхнем уровне, которое соответствует количеству классов c в обучающей информации I_0 . Следует отметить, что слишком малое количество нейронов в сети приводит к недостатку степеней свободы, что не позволяет довести процесс обучения до удовлетворительного уровня; слишком большое количество нейронов, наоборот, приводит к излишку степеней свободы, и нейросеть специально настраивается на информацию I_0 , иными словами, переобучается.

После выбора конфигурации мы, используя истинные информационные векторы $\tilde{\alpha}(S)$, составляем функцию

$$E(\bar{w}) = \sum_{S \in L} (Y(I(S), \bar{w}) - \tilde{\alpha}(S))^2 \quad (1.31)$$

и находим веса $\bar{w}^* = \arg \min E(\bar{w})$ методом градиентного спуска. Для того, чтобы метод градиентного спуска можно было применить, функция (1.30) заменяется похожей функцией

$$f(v) = \frac{1}{2} \left(\frac{v}{1 + |v|} + 1 \right). \quad (1.32)$$

Градиент функции $E(\bar{w})$ в заданной точке считается с помощью специального метода обратного распространения ошибки (см. [99]). Начальное приближение \bar{w}_0 задаётся случайно, каждая компонента выбирается из равномерного распределения между $-1,5$ и $1,5$. Поскольку пороговая функция (1.32) выдаёт уже не только значения 0 и 1, при окончательном расчёте ответа $\alpha^A(S)$ по результату $y = Y(I(S), \bar{w}^*)$ используется формула

$$\alpha_i^A(S) = \begin{cases} 1, & \text{если } 0,5 + \delta < y_i, \\ 0, & \text{если } 0,5 - \delta > y_i, \\ \Delta, & \text{в противном случае.} \end{cases}$$

Порог δ экспериментально подобран и равен 0,05.

На этапе обучения, которое полностью состоит из минимизации функции (1.31), требуется порядка $O(L \cdot n \cdot I)$ операций, где n — количество нейронов сети, I — количество итераций. Для распознавания требуется всего $O(n)$ операций.

Для получения приемлемых результатов обычно требуется $I \sim 10^5$ и много более итераций, поэтому нейросеть обучается долго, иногда в суток. Однако в качестве

достоинств нейросети следует отметить быстрый способ вычислений на этапе распознавания и широкие возможности по решению непараметрических многомодальных задач распознавания (см. [96]). По этой причине нейросетевой подход применяется чаще остальных в аналитических видеосистемах. Кроме того, в последние годы удалось создать нейросети сложной структуры с большим числом нейронов и связей, которые эффективно решают задачу распознавания и семантической обработки изображений, см. [33, 34, 35, 36, 37], но при этом для расчёта прямого выхода данных нейросетей только для одной подобласти изображения необходимо выполнить порядка миллиарда сложений и умножений, что не позволяет на текущий момент использовать эти методы в системах реального времени.

Простейшие эвристики. Иногда природа признаков объектов настолько прозрачна, что не требуется использовать сложные алгоритмы классификации при построении алгоритма A . Например, в простом случае алгоритм $A(I_0(K'_1, K'_2, K'_3), I(S))$ для объекта S со стандартной информацией $I(S) = (x_1, x_2)$ мог бы быть следующим:

$$\alpha^A(S) = \begin{cases} (1, 0, 0), \text{ т.е. } (S \in K_1), \text{ если } x_1 < a, \\ (0, 1, 0), \text{ т.е. } (S \in K_2), \text{ если } x_1 \geq a \text{ и } x_2 < b, \\ (0, 0, 1), \text{ т.е. } (S \in K_3), \text{ если } x_1 \geq a \text{ и } x_2 \geq b, \end{cases}$$

где a и b — некоторые пороги, которые иногда можно выбрать «из общих соображений», практически без обучения, хотя на практике всё же при выборе порогов используется статистика $I_0(K'_1, K'_2, K'_3)$.

Критерии выбора алгоритма классификации. Характеристики рассмотренных выше алгоритмов классификации сведены в таблицу 1.1. Как правило, доля ошибок классификации зависит не от применяемого алгоритма, а от параметров $I(S)$, выбранных для обучения — чем сильнее параметры дискриминируют классы K_j , тем лучше результат. Конечно, при этом следует учитывать, что, например, квадратичный дискриминантный анализ предполагает одномодальные распределения $I(S)$ внутри каждого класса K_j , а методы типа нейросетевой классификации и алгоритм выбора по k ближайшим соседям не накладывают таких ограничений. Могут быть и другие ограничения на входные данные алгоритмов классификации, которые должны выполняться при их применении. Часто ограничения одномодальности и другие можно обойти с помощью преобразования в пространстве признаков, в противном случае нужно выбрать альтернативный алгоритм классификации.

Главными критериями выбора метода классификации является сложность на этапе обучения и на этапе распознавания. В аналитических видеосистемах обычно не принимают во внимание сложность алгоритма классификации на этапе обучения, поскольку этот этап проводится предварительно до запуска системы. В штатном режиме работы системы в реальном времени модули классификации должны занимать

Алгоритмы	Много- модаль- ность	Число операций при обучении	Число опе- раций клас- сификации
Квадратичный дискриминантный анализ	нет	$O(L + cn^3)$	$O(cn^2)$
Линейный дискриминантный анализ	нет	$O(L + n^3)$	$O(L^{1,5})$
Метод k ближайших соседей	да	$O(L)$	$O(L^{1,5})$
Классификация с помощью нейросети	да	$O(LNI)$, N — число нейронов, I — число итераций	$O(N)$
Эвристики	—	—	$O(1)$

Таблица 1.1. Характеристика алгоритмов классификации.

как можно меньше вычислительных ресурсов, то есть обладать как можно меньшей сложностью на этапе классификации.

Исходя из этих условий и учитывая приведённые выше оценки сложности, чаще всего при классификации объектов в аналитических видеосистемах используется нейросетевой подход, при котором количество операций, требуемых на классификацию, постоянно для всех объектов и пропорционально числу всех входов у нейронов сети.

1.5. Теоретическая оценка достоверности выходных данных алгоритмов сопряжения, синхронизации и семантической классификации в системах технического зрения

Основное назначение аналитических видеосистем — это достоверное представление данных для оператора о ситуации в зоне наблюдения видеокамер, входящих в состав системы. Для достоверной оценки обстановки требуются надёжные способы анализа событий, которые позволили бы выполнить ряд требований к аналитической видеосистеме.

1. Наблюдаемый объект, отмеченный алгоритмом выделения движения (см. определение 1.1), находящийся в зоне наблюдения двух неподвижных камер, должен быть отмечен в программе наблюдения, запущенной на терминале оператора, только один раз.

2. Как можно больше наблюдаемых объектов должны попасть в поле зрения камер на поворотном устройстве за счёт наведения на объект, причём увеличение изображения поворотной видеокамеры должно определяться на основе задач наблюдения.

3. Движущийся объект должен быть классифицирован как принадлежащий к каким-либо категориям, например, «люди» или «машины», чтобы была возможность не выводить события некоторого типа и облегчить восприятие обстановки оператором видеосистемы.

Для выполнения сопоставления данных об объекте, определенного с помощью выделения движения на двух разных камерах, прежде всего требуется точное преобразование координат точек изображения с одной камеры на другую и наоборот, то есть требуется довольно точно выполнить операцию сопряжения двух неподвижных камер (см. определение 1.3). Пусть в процессе такого сопряжения мы получили функцию преобразования координат

$$(x', y') = f(x, y),$$

которая сопоставляет координатам изображения (x, y) одной камеры координаты изображения (x', y') другой камеры. В силу возможных неточностей метода сопряжения, функция f определяет координаты (x', y') с некоторой ошибкой. Допустим также, что «идеальное» преобразование сопряжения, у которого нет ошибок, задаётся функцией f' :

$$(x', y') = f'(x, y).$$

В качестве оценки погрешности преобразования можно воспользоваться следующей нормой:

$$\Delta_f = \max_{(x,y)} \|f(x, y) - f'(x, y)\|, \quad (1.33)$$

где под знаком « \max » стоит обозначение евклидовой нормы, т.е. корень из суммы квадратов разностей координат x' и y' оцениваемого и «идеального» преобразования. Величина погрешности Δ_f , таким образом, измеряется в точках изображения камеры, и поэтому является *безразмерной*.

Чем меньше погрешность сопряжения неподвижных камер Δ_f , тем меньшего размера объекты из общей зоны наблюдения могут быть сопоставлены. Поскольку для большинства алгоритмов выделения движения предельный минимальный размер объекта, который может быть надёжно определён, равен 4 пикселам изображения, то основное ограничение, накладываемое на точность сопряжения, задаётся соотношением

$$\Delta_f < 4. \quad (1.34)$$

Синхронизация, понимаемая в смысле определения 1.4, также влияет на сопоставление данных об объектах: при запаздывании обмена данными между обработчиками видеосигнала, которые анализируют синхронизированные с разных камер данные об объекте, уменьшается вероятность правильного определения идентичности. Необходимая скорость синхронизации, т.е. время обмена данными между обработчиками,

определяется скоростью движения объекта, однако существует время синхронизации, уменьшать которое уже не целесообразно. При частоте обработки 25–30 кадров в секунду (именно столько передают обычные видеокамеры), не имеет смысла добиваться времени синхронизации меньшего, чем

$$1/30\text{сек} \approx 3 \cdot 10^{-2}\text{сек},$$

поскольку обновление данных об объекте всё равно происходит с частотой, равной частоте анализа кадров видеопотока.

При сопряжении поворотной и неподвижной видеокамер, согласно определению 1.2, требуется найти соответствие между координатами изображения (x, y) неподвижной камеры и углами поворота (α, β) оптической оси вращающейся камеры в некоторой полярной системе координат с началом в точке крепления камеры (дополнительно определяются параметры увеличения объектива в соответствии с угловыми размерами объекта). Таким образом, сопряжение неподвижной и поворотной видеокамер — это поиск функции преобразования τ :

$$(\alpha, \beta) = \tau(x, y).$$

Допустимые отклонения значений функции $\tau(x, y)$, полученной при сопряжении, от «идеальной» функции $\tau'(x, y)$, не могут задаваться такими же простыми ограничениями, как ограничения в формулах (1.33) и (1.34), потому что максимальное угловое значение погрешности определяется расстоянием от поворотной камеры до объекта, на который производится наведение. Если производится наблюдение за человеком, находящимся, например, на расстоянии 400 метров от видеокамеры, то, считая ширину плеч человека примерно равной половине метра, получим, что горизонтальный угловой размер человека будет $0,5/400 \approx 0,07^\circ$, и для того, чтобы получить изображение лица хотя бы в половину ширины кадра видеокамеры, нельзя при сопряжении ошибаться более, чем на указанный угол $0,07^\circ$. Чем меньшего разрешения изображения объектов нужно получать на поворотной камере и чем ближе объект находится к точке закрепления видеокамеры, тем большее значение погрешности является допустимым, так что значение $0,07^\circ$ можно считать оптимальной погрешностью.

В обычных условиях погрешность сопряжения больше $0,07^\circ$. Большинство современных поворотных устройств с видеокамерами, применяемых в видеосистемах, в силу электромеханических ограничений имеют неустранимую погрешность наведения оптической оси, равную $0,1^\circ$, по каждому из углов α, β . Поэтому в большинстве случаев погрешность сопряжения поворотной и неподвижной видеокамер можно считать неулучшаемой, если

$$\Delta_\tau = \max_{(x,y)} \|\tau(x, y) - \tau'(x, y)\| < 0,1^\circ.$$

Синхронизация данных о положении объектов между обработчиками изображений неподвижной камеры и программными модулями управления поворотной камерой также влияет на точность наведения на движущиеся объекты. Как и в случае сопряжения двух неподвижных камер, время синхронизации данных менее 30мс не имеет смысла улучшать. С другой стороны, задержки синхронизации вносят также вклад в величину времени предсказания положения объекта, которое складывается из времени поворота камеры при наведении на объект и времени задержки синхронизации. Таким образом, чем больше задержка, тем больше время экстраполяции положения объекта относительно текущего, тем больше погрешность экстраполяции. Время поворота камеры измеряется значениями от 0,1с до нескольких секунд, поэтому и время синхронизации не должно по возможности превышать десятые доли секунды, чтобы не вносить дополнительный вклад в погрешность по сравнению с той погрешностью, которую вносит время поворота камеры на объект.

Классификация объектов по категориям может быть использована для фильтрации событий, происходящих в зоне наблюдения, при выводе на терминал оператора. Однако для того, чтобы сохранялась достоверность представления данных о событиях на выходном терминале видеосистемы, алгоритмы классификации должны выдавать результаты, соответственно, с высокой достоверностью, т.е. классифицировать объекты с определённой долей ошибок, которая не превышает требуемое для конкретных задач наблюдения значение. Во многих случаях считается достаточным, если количество ошибочных классификаций не превышает 5% от общего количества идентифицированных объектов, однако для ряда приложений доля ошибок должна быть значительно меньше.

Прежде всего требования к частоте ошибок классификации определяются плотностью потока событий, которые анализирует аналитическая видеосистема. Если в зоне наблюдения происходит всего 100 событий в день, то 5% ошибочно классифицированных событий означают всего лишь 5 событий, на которые оператор должен обратить внимание в течение всего дня. Для систем с большим количеством видеокамер, анализирующих события в оживлённых местах, 5% ошибочных классификаций могут обозначать для оператора системы необходимость анализировать 5 событий в секунду, и поэтому такая достоверность классификации не является удовлетворительной.

Определив параметры, описывающие качество сопряжения, синхронизации и классификации, опишем основные цели и задачи диссертационной работы, которые как раз заключаются в повышении достоверности анализа информации в видеосистемах за счёт улучшения данных параметров.

1.6. Анализ достоинств и недостатков традиционных подходов к обработке информационных видеосигналов

Как следует из анализа обзора, сделанного в предыдущих параграфах, при разработке аналитических видеосистем возникает множество проблем и задач, и для каждой проблемы или задачи существует некоторое количество готовых решений: либо теоретических, проверенных на практике, либо сугубо практических, не подкрепленных публикациями теоретического характера.

В таблице 1.2 кратко перечислены основные достоинства и недостатки рассматриваемых методов. Раскроем содержание этой таблицы более подробно.

Предварительная фильтрация позволяет повысить качество изображений, измеряемое как объективными характеристиками, например соотношением сигнал/шум, так и субъективным восприятием. В последнем случае для снижения субъективности проводится опрос группы людей, и субъективная оценка усредняется.

Кроме высоких требований к вычислительным ресурсам (количество операций любого преобразования фильтрации не менее количества точек изображения), к основным недостаткам предварительной обработки следует отнести неопределенность выбора параметров преобразования изображения для большинства методов. Фильтрация изображения F является некоторым преобразованием \mathcal{F} , сопоставляющим исходному кадру некоторый кадр F' :

$$F' = \mathcal{F}(F, \alpha),$$

где α — параметры преобразования. Качество изображения F' существенно зависит от выбора параметров α , однако, для многих фильтров не существует метода выбора оптимального значения параметра. Например, фильтрация Винера-Хопфа (см. п. 1.3) параметрически зависит от корреляции исходного изображения и шума в каждой точке. Чем выше предполагаемый шум, тем сильнее фильтр «сглаживает» изображение. Выбор степени сглаживания можно произвести лишь эвристическими методами.

Методы поиска и сравнения фрагментов изображений, основанные на формулах (1.14) и (1.17), при современном развитии ЭВМ позволяют сравнительно быстро, в течение десятка миллисекунд устанавливать смещение изображений либо сопоставлять различные изображения. Данные методы позволяют решить широкий круг задач: автоматически «шивать» (сопрягать) два изображения с общей зоной обзора с одинаковым увеличением в стереоизображении, определять смещение крупных объектов на последовательности видеоизображений, проводить сопоставление изображений для сравнения, компенсировать глобальное движение при перемещении видеокамеры в пространстве для выделения относительного движения объектов в поле зрения. Главный недостаток идеи вычислять смещения по изображению — это чувствительность

Методы	Достоинства	Недостатки
Фильтрация перед обработкой	+ выделение новой информации + улучшение характеристик изображения	– проблема выбора параметров фильтрации – число операций $O(H \cdot W)$ и выше (см. формулу-определение (1.1))
Поиск и сравнение фрагментов изображений	+ автоматическое определение смещений, необходимое во многих приложениях + отказ от полного перебора при сравнении в пользу скорости выполнения	– низкая устойчивость к помехам изображений – надёжность сопоставления снижается при увеличении «степеней свободы» искажения сравниваемых изображений
Выделение движения	+ рекурсивный способ оценки характеристик изображения ускоряет процесс выделения движения + адаптивный выбор порогов	– низкая устойчивость к помехам изображений средней и низкой частоты
Сопряжение изображений	+ при сопряжении повышаются эргономические показатели терминалов аналитических видеосистем + сопряжение повышает качество распознающих систем	– при сопряжении используется приближённая форма преобразования, что снижает точность – в аналитических видеосистемах не применяется автоматизированное сопряжение – не исследовано в полной мере сопряжение топографического плана с изображением
Определение типов изображений	+ существует много разнообразных методов классификации, позволяющих провести разбиение на классы на основе анализа пространства признаков изображений	– процесс формирования набора признаков не формализован – при распознавании в аналитических видеосистемах не применяется сопряжение изображений

Таблица 1.2. Характеристика достоинств и недостатков методов обработки цифровых видеосигналов.

к помехам различного характера: как радиотехническим шумам видеосигнала, так и помехам, связанным с естественными явлениями: изменение освещённости наблюдаемой сцены при переменной облачности, перемещение посторонних объектов в зоне наблюдения и другие. В силу увеличения количества помех при преобразованиях типа поворот и изменение масштаба надёжность сопоставления также снижается при изменении увеличения и при повороте сравниваемых изображений.

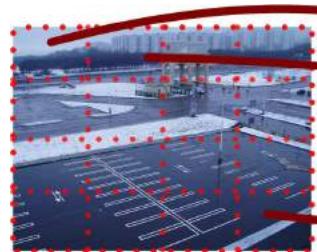
Методы выделения движущихся изображений, основанные на рекуррентных формулах (1.18), позволяют определить движение объектов в реальном времени, однако,

как и в случае сравнения изображений, несмотря на адаптивный выбор порога T и другие способы снижения влияния естественных и технических помех, полностью избавиться от ложного определения движения в кадре не удаётся ни одним из методов. В силу природы оценок характеристик изображения по формулам (1.18), наиболее часто ложное определение движения вызывают помехи средних и низких частот, распределённых во времени и пространстве (высокочастотные помехи отсеиваются за счёт адаптивного порога). Неточное выделение информации о положении объекта Ω (см. (1.3)), обусловленное некорректной работой алгоритма выделения движения, впоследствии приводит к ошибкам классификации типа объекта.

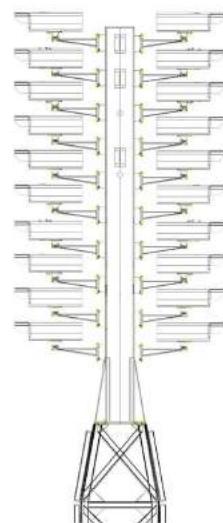
Различные типы *сопряжения изображений* значительно повышают качество восприятия информации оператором аналитической видеосистемы. Среди типов сопряжения изображений можно отметить сопряжение поворотных и неподвижных видеокамер, сопряжение изображений неподвижных камер между собой и сопряжение изображений неподвижных видеокамер с изображением плана местности.

Например, без использования поворотных видеокамер в аналитических видеосистемах для того, чтобы увеличить разрешение изображения на одной из видеокамер в S раз только с помощью других неподвижных камер, необходимо не менее S^2 видеокамер с фокусным расстоянием объективов в примерно S раз большим, чем на исходной видеокамере. Пример увеличения разрешения в 4 раза с помощью неподвижных видеокамер изображён на рис. 1.13, и из этого примера видно, что такое решение приводит к резкому увеличению количества оборудования, что сильно усложняет как монтаж, так и техническое обслуживание видеосистем. Кроме того, при контроле обстановке в зоне наблюдения видеокамер потребуется следить не за одним изображением, а за 16 изображениями одновременно, и нагрузка операторов системы возрастёт. Поворотная видеокамера, в которой реализовано управление увеличением и которая сопряжена с обзорной видеокамерой, позволила бы как сократить количество видео-приборов, необходимое для повышения разрешений изображений в зоне наблюдения, так и выделять из потока событий в зоне наблюдения только необходимые для контроля оператора.

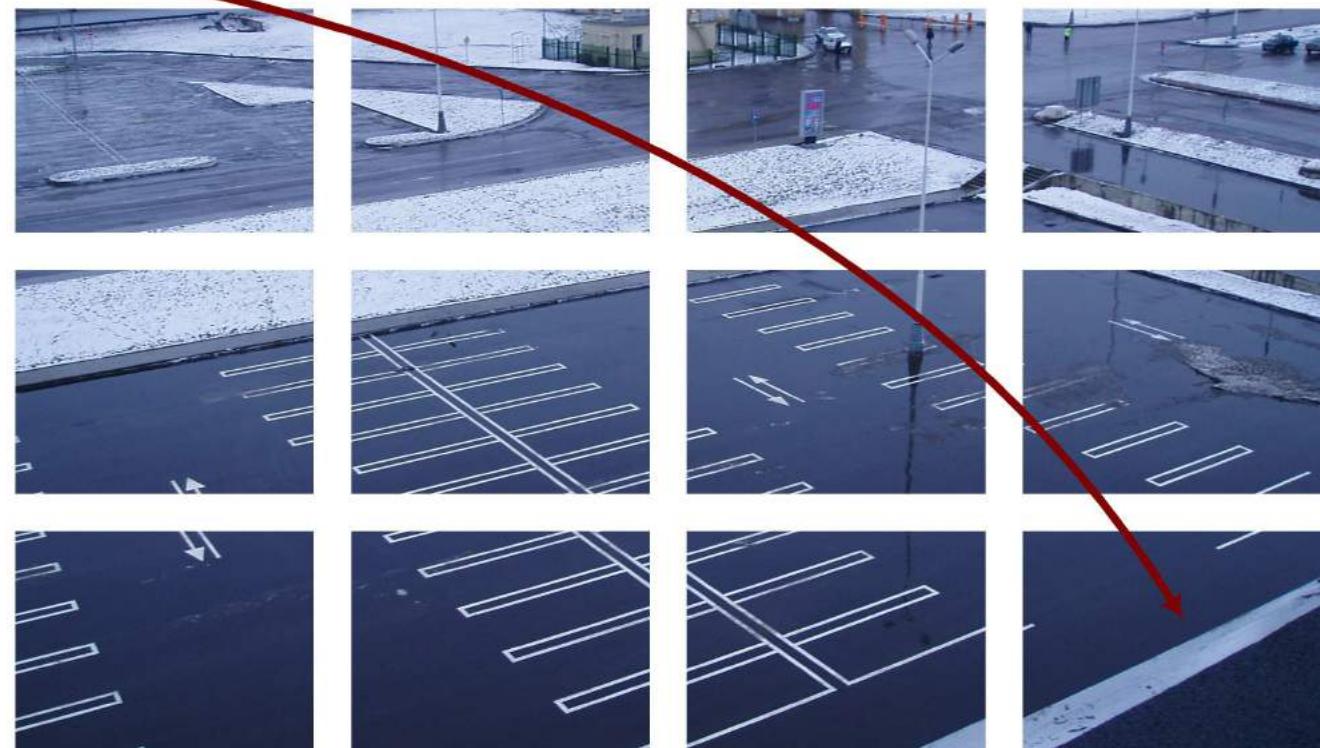
Использование сопряжения неподвижных видеокамер между собой позволяет избежать дублирования при передаче информационных сообщений об объектах в зоне пересечения областей видимости сопряжённых видеокамер. Пример дублирования информации в терминале оператора изображён на рис. 1.14. При эксплуатации систем, в которых количество видеокамер велико, исключение дублирования сообщений с помощью сопряжения позволяет уменьшить количество случаев, требующих реакции оператора системы. Таким образом, повышается автоматизация обработки видеоизображений, и для обработки нештатных ситуаций в зоне видеонаблюдения требуется



а) Изображение обзорной камеры



б) Внешний вид приборной конструкции из 16 видеокамер



в) Изображение 16 обзорных видеокамер, разрешение увеличено в 4 раза

Рисунок 1.13. Пример увеличения разрешения изображений наблюдаемых объектов за счёт увеличения количества видеокамер.

меньшее количество операторов аналитической видеосистемы.

Если видеокамеры сопряжены с топографическим планом местности, то возможен альтернативный способ визуального представления информации об обстановке в области зрения видеокамер, см. рис. 1.15. Вместо того, чтобы непрерывно анализировать изображения массива видеокамер, количество которых может быть очень велико, операторы системы могут наблюдать только за планом местности, на котором отображаются любые объекты и события, выделенные аналитической видеосистемой. В тех случаях, когда число событий невелико, например, менее одного события за 10 минут, один оператор с помощью сопряжения может контролировать ситуацию в зоне наблюдения большого количества видеокамер, даже если это количество превышает сотни единиц. Поскольку один оператор не в состоянии наблюдать за обстановкой более, чем на 20 камерах, то преимущества видеосистем с применением сопряжения с планом местности очевидны: для наблюдения за сотней видеокамер требуется не менее пяти операторов, даже если информационный поток о событиях не велик, а для наблюдения за планом местности достаточно одного человека.

Сопряжение изображений может применяться также в стереосистемах для оценки расстояния до наблюдаемых объектов, и при построении панорамных изображений. При использовании стереосистем повышается достоверность анализа событий на наблюдаемой сцене, поскольку в анализ включается информация об изменении расстояния до объектов. Панорамные изображения упрощают работу операторов видеосистем при контроле событий в поле зрения поворотных видеокамер, работающих в режиме сканирования.

Таким образом, видеосистемы, использующие сопряжение изображений, повышают надёжность классификации типов объектов, уменьшают количество сигналов о «новых» событиях, на самом деле новыми не являющихся, появляется возможность детализировать изображение событий системы, улучшается эргономика управления выводом видеокамер и событий с помощью привязки к плану местности. Сопряжение изображений в стереосистемах повышает достоверность детектирования событий засчёт дополнительной информации, а сопряжение изображений на панораме во время сканирования сцены с помощью видеокамеры на поворотном устройстве улучшает восприятие информации операторами видеосистем. Тем не менее, существующие видеосистемы не используют сопряжение видеокамер в полной мере, а в доступных автору литературных источниках не освещены вопросы автоматической настройки сопряжения синхронных изображений поворотных и неподвижных камер, не используются формулы проективного преобразования в прямом виде и не определяются способы привязки камер к плану местности.

Классификация изображений по типу обычно проводится с помощью одного из

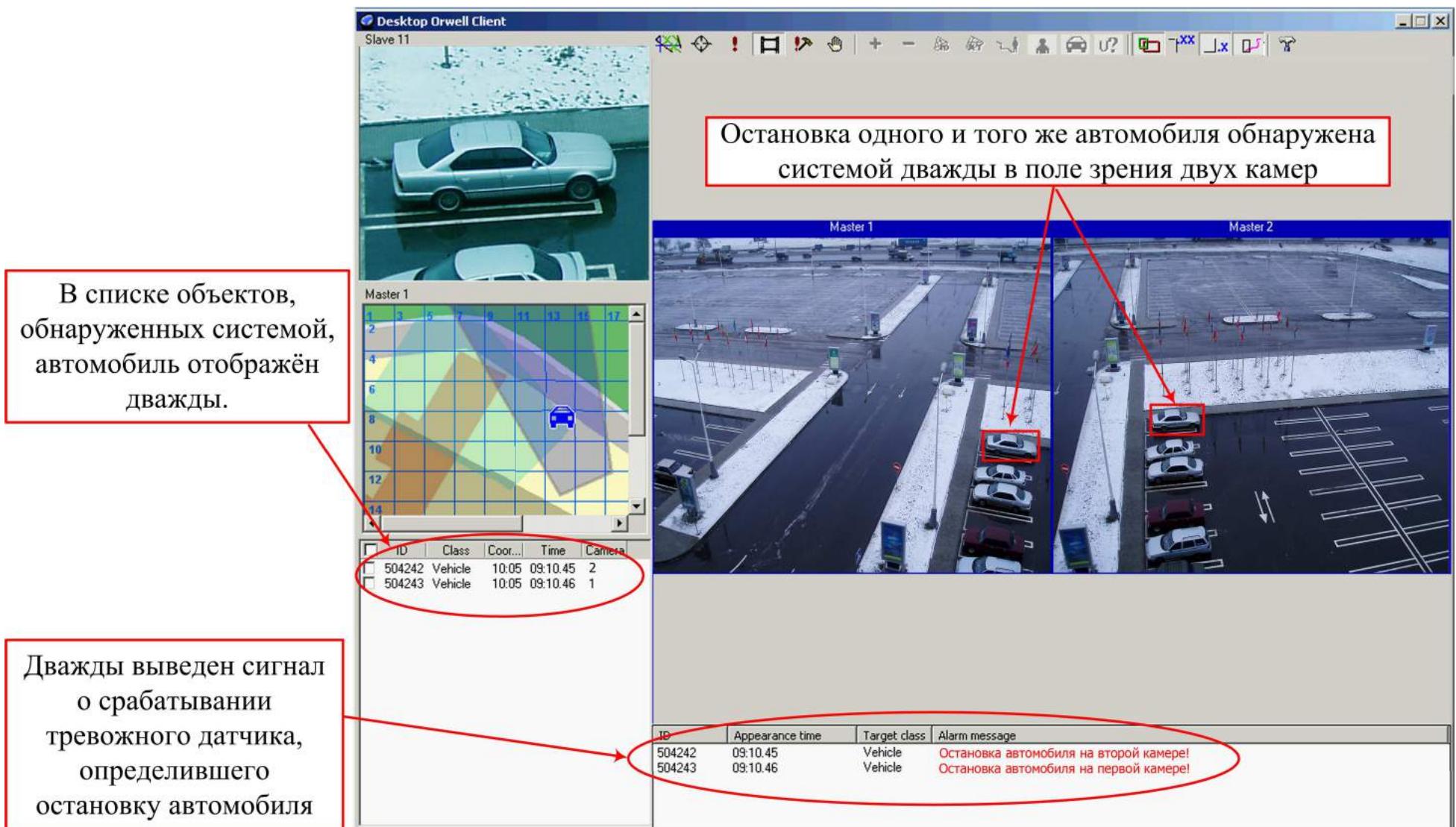
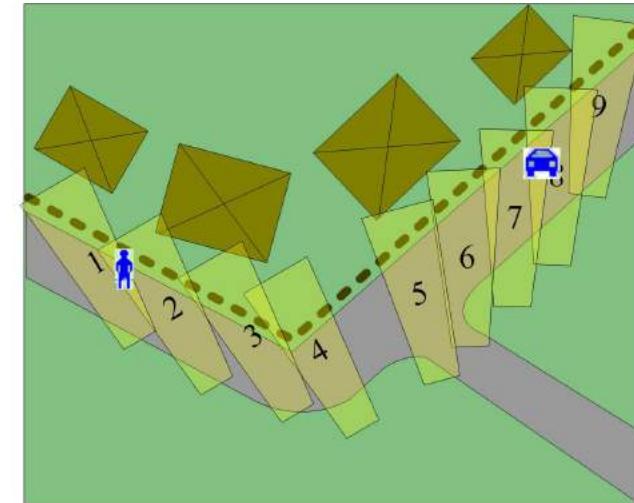


Рисунок 1.14. Дублирование информационных сообщений при отсутствии сопряжения между неподвижными видеокамерами.



а) Визуальное представление наблюдаемой территории
в виде массива изображений



- зона наблюдения видеокамеры
- охраняемый периметр
- условное обозначение дома
- условное обозначение дороги
- обозначение объектов,
выделенных системой

б) Отображение событий на наблюдаемой территории
с помощью сопряжения с планом местности

Рисунок 1.15. Визуализация событий в области контроля видеосистемы, применяющей сопряжение камер с планом местности.

алгоритмов, рассмотренных в п. 1.4. Точность результатов рассмотренных алгоритмов классификации всецело зависит от распределения свойств объектов в пространстве признаков. Не существует универсального и надёжного способа синтеза оптимального набора признаков: каждый раз эта задача решается на основе интуитивных предложений с последующим экспериментальным подтверждением или опровержением. В публикациях не содержится описания экспериментов по синтезу набора признаков для классификации объектов по видеосигналу на основе сопряжения изображений видеокамер.

1.7. Результаты анализа и исследования современных методик оценки достоверности алгоритмической обработки видеинформации

Современные технологии разработки программного обеспечения предполагают автоматизированное тестирование программных систем перед сдачей в эксплуатацию. Автоматизация позволяет более полно покрыть тестами все способы использования программ, а также позволяет избежать ошибок, допускаемых при тестировании вручную.

Алгоритмы обработки видеинформации — это тоже программы, которые необходимо тестировать, причём трудоёмкость ручного тестирования программ компьютерного зрения с алгоритмами семантического сжатия, как правило, выше, чем трудоёмкость тестирования обычных пользовательских программ, что обусловлено необходимостью наблюдения за большим потоком видеинформации во время тестирования. Поэтому актуальность автоматизации тестирования для алгоритмов компьютерного зрения очень высока.

Для автоматизации тестирования необходимы критерии оценки качества работы алгоритмов семантического сжатия, которые могут быть рассчитаны автоматически. На данный момент распространённым является подход, когда изображения хранятся в качестве исходных данных для алгоритмов обработки в форме отдельных файлов или видеофайлов, и вместе с этими данными содержится информация, описывающая эталонный результат, или, иначе говоря, разметку исходных данных. В зависимости от требуемого результата работы алгоритма, разметкой может быть следующая информация.

1. Категории объектов, присутствующие на изображении, например см. [122].
2. Множество точек, которые являются точками изображения объекта. Примеры поточечной разметки также есть в [122]. Как правило, границы объектов задают с помощью кривых или многозвездных многоугольников, встречается также поточечная разметка, в которой принадлежность точки к тому или иному типу изображения определяется функцией-маской, определённой на множестве точек изображения, на-

пример, см. [123].

3. Разметка объектов может быть загружена до минимального прямоугольника с вертикальными и горизонтальными сторонами, включающего в себя все точки изображения объекта. Такая разметка применяется из-за простоты её подготовки вручную по сравнению с поточечной разметкой.

4. На видеопоследовательности в разметке может быть информация не только о положении того или иного объекта в форме ограничивающего прямоугольника или контура, но и о том, как объект перемещался между кадрами, т.е. о траектории объектов, см. [124].

Данная информация используется для расчёта следующих показателей работы алгоритма.

1. Если алгоритм в результате выдаёт положение объекта в форме множества точек A , а в разметке шаблонное расположение объекта задано с помощью множества точек T , то точность автоматического выделения позиции обычно оценивается в литературе как

$$\varrho(A, T) = \frac{|A \cap T|}{|A \cup T|}.$$

При полном совпадении результата с шаблоном $\varrho = 1$, в противном случае это значение меньше 1.

2. Ошибка второго рода оценивается как отношение количества объектов из разметки, которые алгоритм не выделил на изображениях, к общему количеству размеченных объектов. В работе [122] критерий фиксации объекта разметки алгоритмом заключается в том, что $\varrho(A, T) > 0.5$.

3. Ошибка первого рода оценивается как доля объектов в результатах работы алгоритмов, для которых не нашлось пары в разметке, от общего числа выданных алгоритмом объектов.

4. В работе [123] применяются поточечные оценки ошибок первого и второго рода.

Практика показывает, что перечисленных инструментов не достаточно для полноценной оценки качества работы алгоритмов семантического сжатия. В частности, оценка качества сопровождения объектов алгоритмом вдоль его траектории почти не отражается в приведённых выше оценках. Поэтому необходимо разрабатывать новые методики тестирования для алгоритмов сопровождения.

1.8. Цели и задачи диссертационной работы

Учитывая недостатки методов обработки видеоизображений, используемых в аналитических видеосистемах, основываясь на анализе современного состояния в области

систем информационной обработки видеосигналов и управления поворотными видеокамерами, в диссертационной работе поставлены следующие цели и задачи.

1. Проанализировать функции компонент распределённой модели параллельной обработки сигналов от различных видеокамер и разработать способы синхронизации вычислительных процессов.

Поскольку задержки при передаче сообщений синхронизации между параллельными процессами обработки изображений влияют на точность управления поворотными камерами и точность сопоставления событий, необходимо разработать модель взаимодействия распределённых модулей и проанализировать возможные времена реакции системы на внешние события как теоретически, так и экспериментально. Необходимо также достичь показателя времени синхронизации, не превышающего значения примерно 100мс.

2. Вывести соотношения и исследовать их точность для сопряжения изображений неподвижных видеокамер между собой, а также с топографической картой местности, при неизвестных параметрах ориентации видеокамер в пространстве и неизвестных характеристиках объективов.

Обычно при сопряжении изображений используют преобразования координат вида

$$\begin{aligned}x' &= Ax + By + Cxy + D, \\y' &= Ex + Fy + Gxy + H,\end{aligned}$$

вместо более точных дробно-линейных формул проективного преобразования (здесь A, B, C, D, F, G, H — параметры преобразования), чтобы воспользоваться линейными методами оптимизации при подборе преобразований. В данной работе планируется использовать проективное преобразование с целью улучшения точности. Необходимость разработки метода сопоставления точек изображения и плана местности при неизвестных характеристиках объектива видеокамеры и её расположения в пространстве обусловлена тем, что сложность измерения данных параметров достаточно высока и требует лабораторных условий, что не приемлемо при развёртывании видеосистемы в обычных эксплуатационных условиях. Предполагается, что точность сопряжения, оцениваемая по формуле (1.33), не будет превышать 4 точек.

3. Разработать способы автоматизированного сопряжения поворотных видеокамер с неподвижными.

Существуют видеосистемы, в которых используется сопряжение поворотных и неподвижных видеокамер, однако настройка этого сопряжения, по-видимому, проводится вручную. Автоматизация процесса сопряжения минимизирует ошибки, связанные с неправильным сопоставлением изображений при ручном управлении, и зна-

чительно уменьшает время первоначального развёртывания видеосистемы. В данной работе планируется также оценить, насколько погрешность предложенного метода сопряжения превышает $0,1^\circ$.

4. Исследовать способы применения стереоизображений в системах технического зрения. Разработать меры по повышению эффективности работы тех или иных элементов видеосистем с помощью откалиброванных парных видеокамер.

5. Провести анализ возможности создать алгоритм построения панорамных изображений в системах реального времени и на основе анализа провести разработку данного алгоритма.

Построение панорамных изображений в режиме реального времени позволит существенно уменьшить требуемое для защиты объектов количество видеокамер за счёт применения поворотных устройств. Кроме того, это позволит улучшить эргономику программного обеспечения охранных видеосистем, в составе которых используются поворотные камеры, что, в свою очередь, повысит эффективность применения видеокамер на поворотных платформах.

6. Разработать новые методики оценки качества работы алгоритмов семантического сжатия. Исследовать результаты применения алгоритмов классификации объектов для подсистем распознавания образов, основанных на параметризации данных обработки видеосигналов, в том числе с сопряжённых видеокамер.

Цель исследования — экспериментальная проверка эффективности использования параметров определённого вида при классификации изображений объектов по типам. В дополнение к параметрам, предложенным в работе [15] и аналогичным, предполагается использовать параметры, полученные с помощью наведения поворотной камеры на объекты наблюдения, и проверить, приведёт ли это к улучшению качества классификации. В конечном итоге необходимо оценить достоверность классификации, сравнить полученные результаты с 5%-м порогом допустимых ошибок.

При положительных результатах исследования указанного выше круга вопросов значительно повышается достоверность результатов работы аналитических видеосистем, а область применения таких систем значительно расширяется.

Выводы

1. Проведён обзор и анализ параметров современных приборных комплексов технического зрения, введено понятие «аналитическая видеосистема» и описаны его характеристики. Приведена оценка времени реакции аналитических видеосистем с ручным управлением, которая примерно равна 10с.

2. Введено понятие алгоритмов семантического сжатия, которое обобщает прин-

ципы работы современных систем компьютерного зрения, призванных выделять из видеопотока очень большого объёма небольшую, но достаточную для анализа событий часть.

3. Исследованы и классифицированы современные методы семантической обработки цифровых видеосигналов.

4. Выявлено, что существующие комплексы технического зрения не используют в полной мере методы сопряжения видеокамер и приборов, а в доступных автору литературных источниках не освещены вопросы автоматической настройки сопряжения синхронных изображений, не используются формулы проективного преобразования в прямом виде и не определяются способы привязки камер к плану местности.

5. Анализ показал, что необходимо уменьшить количество видеокамер в системах технического зрения, упростить анализ событий и уменьшить количество операторов.

6. На основе анализа недостатков известных видеосистем и методов обработки видеосигналов поставлена цель диссертационной работы: разработать, проанализировать и внедрить более эффективные способы и алгоритмы семантического сжатия информации, обработки сопряжённых видеосигналов и управления поворотными камерами при создании интеллектуальных автоматических приборных комплексов технического зрения нового поколения.

2. Разработка и функциональный анализ параллельных алгоритмов семантической обработки видеосигналов и создание теоретических основ синхронизации и сопряжения изображений

2.1. Разработка и описание распределённых алгоритмов семантической обработки синхронных видеосигналов от разных датчиков

Как правило, аналитические видеосистемы являются сложными комплексами, состоящими из устройств различного типа, предназначенных для восприятия, передачи и анализа видеосигналов, в том числе вычислительных устройств (ЭВМ). Видеосистемы, построенные на с применением методов семантической обработки видеинформации, также включают в себя программные модули, запускаемые на вычислительных устройствах системы.

Прежде, чем перейти к описанию алгоритмов сопряжения, синхронизации и семантического сжатия, рассмотрим сначала структурное описание устройств и программных модулей аналитической видеосистемы для того, чтобы определить место данных алгоритмов в обозначенной структуре. Определение понятий «сопряжение», «синхронизация» и «семантическое сжатие» см. в п. 1.1.

Состав и основные характеристики устройств аналитических видеосистем. Из рисунка 1.1 следует, что аналитическая видеосистема может состоять из множества объединённых локальной вычислительной сетью(ЛВС) ЭВМ, к которым, в свою очередь подключается разветвлённая сеть из стационарных видеокамер и подвижных видеокамер, установленных на поворотных устройствах.

Схема, изображённая на рис. 1.1, отражает строение системы в общем виде. В самом простом случае видеосистема может состоять из одной ЭВМ и нескольких подключенных к ней видеокамер. Данная ЭВМ будет и сервером для аналитических обработчиков видеосигналов, и терминалом оператора, необходимость в сетевых каналах в этом случае отпадает.

Рассмотрим основные типы устройств системы по отдельности.

Сетевые каналы. ЛВС используется в аналитических видеосистемах для решения следующих задач: синхронизации данных для различных обработчиков видеосигнала, расположенных на разных ЭВМ, передачи видеосигнала в сжатом цифровом виде и сигналов о срабатывании фильтров событий на терминальные ЭВМ.

Пропускная способность сети определяет максимальное количество устройств, которые могут быть подключены в систему и работать при этом в синхронизированной и сопряжённой между собой связке, поскольку чем больше видеокамер задействовано в сопряжении, тем больше данных приходится передавать по сети между обработ-

чиками видеокамер для синхронизации. В большей степени пропускная способность сети ограничивает количество терминалов, которые могут одновременно подключиться к серверам-поставщикам обработанного видеосигнала, поскольку поток цифровых видеоданных даже в сжатом виде занимает значительную часть передаваемой по сетевым каналам системы информации.

Далее мы будем считать, что пропускная способность сети равна 100Мбит/с (ста мегабитам в секунду), поскольку наиболее доступные устройства, позволяющие организовать сравнительно быструю передачу данных в ЛВС, являются Ethernet-совместимыми устройствами с указанной пропускной способностью.

Основные (неподвижные) видеокамеры. Введем термин мастер-камера или, более коротко, мастер для обозначения неподвижных видеокамер.

Определение 2.1. *Мастер-камеры* — это видеокамеры, которые являются источником основной информации для анализа ситуаций и событий в зоне наблюдения с помощью алгоритма определения движения (см. определение 1.1).

Конструкция камеры, и тип передаваемого сигнала не имеют решающего значения для функционирования алгоритмов обработчика видеосигналов: качество работы алгоритмов выделения движения определяется характеристиками изображения, количеством кадров, передаваемых от видеокамеры к обработчику за секунду, разрешением изображения, светочувствительностью (способностью работать в тёмное время суток). Исходя из этих параметров выбираются видеокамеры для подключения в аналитическую видеосистему.

Объективы для камер выбираются из аналогичных соображений, но дополнительно накладывается ограничение на фокусное расстояние объектива. Ограничение формируется из расстояния, на котором находится зона наблюдения относительно точки закрепления камеры: чем дальше находится объект, тем большее фокусное расстояние необходимо для обеспечения приемлемого углового разрешения объектов, за которыми производится наблюдение. Например, для наблюдения за зоной, дальний край которой находится на расстоянии 300 метров, требуется объектив с фокусным расстоянием около 30мм (цифра зависит от размера ПЗС-матрицы), если изображение с камеры поступает в формате 352×288 точек на кадр: в этом случае разрешение изображения человека среднего роста по линейным размерам будет от 2 точек по ширине до 10 точек по высоте на дальних расстояниях.

На данный момент наиболее часто используют цифровые видеокамеры с повышенной чувствительностью (иногда в области инфракрасного спектра) с дальнейшей аналоговой передачей сигнала в формате PAL или NTSC.

Вспомогательные (поворотные) камеры. Для поворотных видеокамер введём термин слейв-камера или слейв.

Определение 2.2. Слейв-камера — это видеокамера, установленная на поворотном устройстве, объектив которой поддерживает управление параметрами увеличения и резкости.

Терминология мастер-камера, слейв-камера заимствована из работ [15, 38]. Из перевода термина с английского языка следует, что поворотные камеры выполняют вспомогательные функции для мастеров, предоставляя возможность детализации изображения событий, которые обнаруживаются обработчики мастер-камер. Каждая из слейв-камер обслуживает какой-то набор мастеров системы, с которыми она со-пряжена.

Требования в вопросах передачи видеосигнала для слейв-камер в целом повторяют требования для мастеров из предыдущего пункта за исключением того, что фокусное расстояние слейв-камеры должно по возможности превышать фокусное расстояние мастеров в несколько раз, чтобы детальное изображение события со слейва было лучшего качества и с большим разрешением, чем на мастере, который данное событие зафиксировал.

Чтобы поворотную камеру возможно было использовать в системе, её управление должно удовлетворять ряду свойств. Эти свойства подробно описаны в п. 2.6.

Передача сигналов видео и управления. Наиболее распространённый способ передачи видеосигнала от поворотных и неподвижных видеокамер до обрабатывающих ЭВМ — либо при помощи коаксиального кабеля, либо по витой паре. Дальность передачи по коаксиальному кабелю ограничена несколькими сотнями метров, при этом сигнал в большой степени подвержен искажениям из-за внешних помех, если расстояние передачи более 10 метров. Дальность передачи по витой паре — 2км.

Сигналы управления поворотными устройствами и объективами с переменными параметрами чаще всего передаются по линиям последовательного интерфейса RS-485, дальность передачи по которым не превышает несколько сотен метров. Скорость передачи сигнала зависит от типа поворотной камеры и её настройки, наиболее распространены скорости 4800кбит/с, 9600кбит/с и 19200кбит/с.

Именно расстояния, на которых надежно работает передача сигналов видео и управления, определяют то, что компоненты одной «зоны наблюдения» (см. рис. 1.1) расположены не далее, чем в одном километре друг от друга.

Серверы и терминалы. Введём понятие «сервер» и «терминал» для аналитических видеосистем.

Определение 2.3. Сервер аналитической видеосистемы — это вычислительное устройство, предназначенное для цифрового преобразования и семантического сжатия видеосигналов мастер-камер и слейв-камер, для запуска программных модулей управления поворотными устройствами слейв-камер, а также для других программ-

ных модулей, не предназначенных для визуализации данных операторам видеосистемы.

Определение 2.4. *Терминал аналитической видеосистемы* — это ЭВМ, предназначенная для визуализации анализа обстановки в зонах наблюдения и снабжённая устройствами ввода-вывода, предполагающих взаимодействие с оператором системы.

Любая ЭВМ, способная выполнять операции обработки видеопотока достаточно быстро, может быть сервером аналитической видеосистемы. Количество видеокамер, которое может быть подключено к одному серверу, определяется вычислительной мощностью ЭВМ. Для сохранения архива видео и событий ЭВМ должна обладать внешним носителем информации со скоростью записи и чтения данных, достаточной для сохранения потока видео в реальном времени.

Поскольку программные модули, запускаемые на терминалах аналитических видеосистем, должны принимать сжатый цифровой видеосигнал от многочисленных обработчиков мастер-камер и слейв-камер, и в реальном времени выполнять процедуру декомпрессии видеосигналов, то требования к вычислительной мощности терминальных ЭВМ также достаточно велики.

Вычислительная мощность современных ЭВМ определяется многими параметрами, из которых главные — это тактовая частота центрального процессора и скорость обмена данными между центральным процессором и ОЗУ. Данные два параметра определяют среднее количество операций в секунду для алгоритмов обработки видеосигнала. Чтобы далее приводить оценки вычислительной сложности не только в единицах количества операций, но и во времени, мы введём «эталонные» показатели скорости вычисления, взяв за основу ЭВМ с частотой операций примерно 10^9 команд в секунду и скорость обмена $4 \cdot 10^8$ 32-разрядных слов в секунду между оперативной памятью и процессором.

Определение 2.5. *Эталонная вычислительная мощность ЭВМ* — это вычислительная мощность при частоте операций 3 ГГц и частоте обмена между ОЗУ и ЦП 1,6 ГГц при 64-разряднойшине данных.

Выбор данных параметров в качестве эталонных обусловлен широким распространением элементной базы, из которой состоит ЭВМ с указанными параметрами, а также тем, что приведённая мощность вычислительных ресурсов заведомо достаточна для выполнения нескольких модулей обработки видеосигнала в реальном времени одновременно на одной ЭВМ.

Структура программных модулей аналитической видеосистемы. На рис. 2.1 представлены программные модули, которые составляют математическое обеспечение аналитических видеосистем. Обоснование такой структуры заключается в том, что из назначения и функций аналитической видеосистемы вытекает необхо-

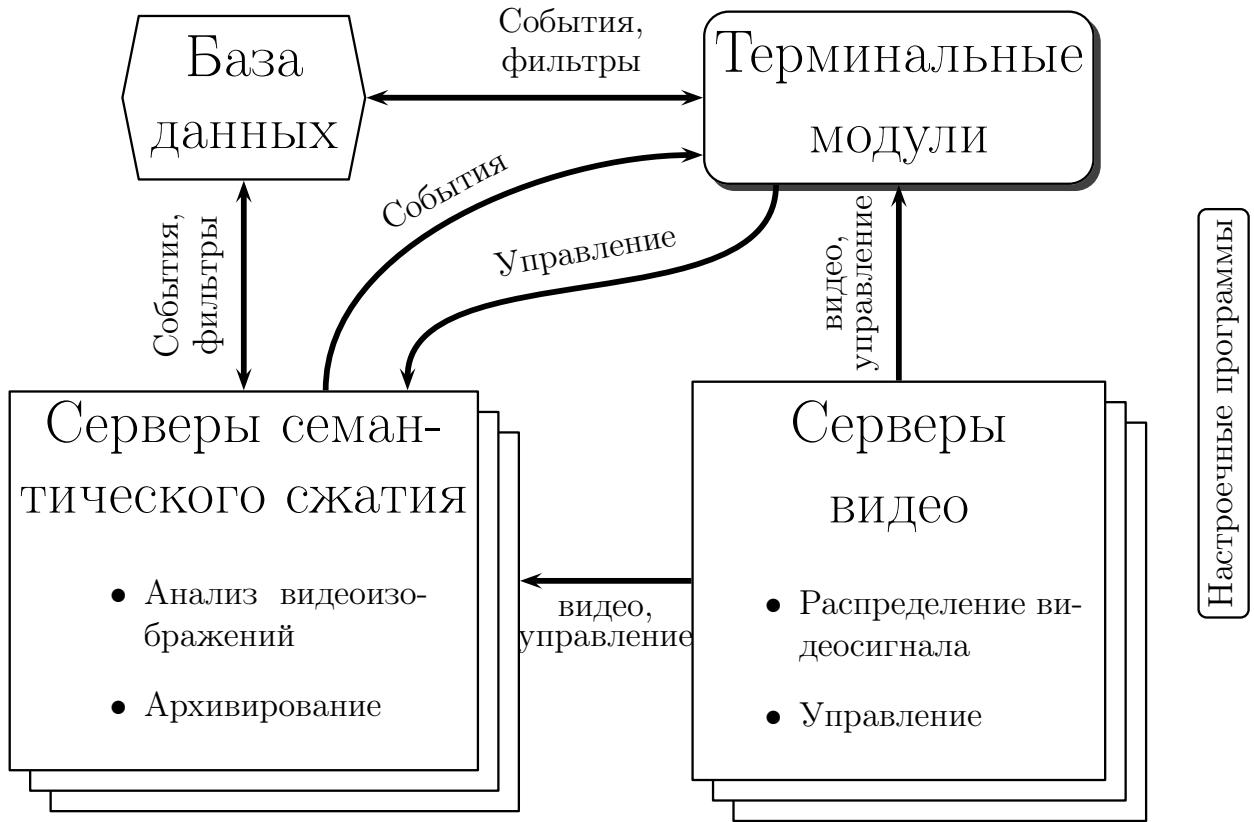


Рисунок 2.1. Схема программных модулей видеосистем с применением семантической обработки видеосигналов.

димость наличия следующих модулей: программы анализа видеинформации (сервер распознавания), программы контроля наблюдаемой обстановки на терминале (терминальный модуль), модуль, реализующий передачу видеосигнала по сетевым каналам от видеокамер к терминальным модулям и модулям анализа видеосигнала (сервер видео), база данных для сохранения информации о событиях системы, и ряд настроочных программ, выполняющих предварительное сопряжение видеокамер перед началом эксплуатации систем. Перечислим основные функции для данного перечня программ.

Сервер видео — это программа, выполняющая функции ввода-вывода видеосигналов от камер и сигналов управления для поворотных устройств в аналитической видеосистеме.

Типичная операционная система обычно не позволяет работать сразу нескольким модулям с одним устройством ввода-вывода видео или устройством передачи сигналов управления. Основное назначение сервера видео — получить доступ к данным устройствам и позволить всем остальным модулям системы получать от этих устройств данные и управлять ими, обеспечивая при этом одновременный доступ к видеосигналу и последовательный доступ к управлению поворотными устройствами.

Различные поворотные устройства видеокамер значительно отличаются друг от друга по протоколу передачи сигналов управления на низком уровне. Второе назначе-

ние сервера видео — предоставить остальным модулям системы универсальный протокол управления поворотными устройствами, который не зависит от типа поворотной видеокамеры. Команды «универсального» протокола сервер видео должен преобразовывать в специфичные команды для конкретного поворотного устройства. При таком подходе разработка программных модулей управления поворотной камерой не зависит от типа поворотного устройства, все специфичные функции поворотных устройств обрабатываются в сервере видео.

База данных аналитической видеосистемы хранит реляционные таблицы с информацией, описывающей состав видеосистемы и результаты семантической обработки, а именно: таблица поворотных и обзорных видеокамер системы с настройками обработчиков для данных камер, таблица-архив событий вместе со свойствами (время возникновения, распознанный тип, файл с фотографией, место события в архиве видео, т.д.), таблица фильтров событий, иначе называемой таблицей «датчиков тревог», таблица кластеров системы с их параметрами (см. определение 2.6 для понятия «кластер»), таблица операторов системы с их правами и обязанностями, связанными с наблюдением за зоной.

В системе, состоящей из небольшого числа видеокамер (не более 100), обычно используется одна ЭВМ с реляционной базой данных. В более крупных системах используется распределённая база данных для балансировки нагрузки при операциях удаления и вставки данных методом распределения задач на несколько ЭВМ. Любой компонент системы получает доступ к данным в базе посредством SQL-запросов, возможно, выполняемых по сети.

Сервер семантического сжатия запускает модули обработчиков видеосигнала, выполняющих алгоритмы системы по семантическому анализу видео с целью распознавания событий, а также алгоритмы управления поворотными камерами и сопряжения различных камер между собой. Выделенные и классифицированные события, в том числе факты срабатывания одного из «датчиков тревог» в результате обнаружения события, прошедшего заданный фильтр, пересыпаются по сети в терминальный модуль, чтобы на терминале отображалась информация, полученная после семантического сжатия видеосигналов в системе.

Терминальный модуль, иначе называемый «клиентом системы», запускается на терминалах и отображает информацию с наблюданной территорией, полученную по сети от серверов системы. Внешний вид одного из режимов терминального модуля представлен на рис. 2.2. На этом рисунке мы видим, как выделяются рамочками движущиеся объекты в кадре обзорной видеокамеры, каждому объекту приписана категория (человек, машина, группа людей), причем поворотная камера (левый верхний угол) дает изображение одного из объектов крупным планом (функция наведения

доступна после процедуры сопряжения поворотной и неподвижной видеокамер).

Под изображением от поворотной камеры отображен план местности с мнемоническим отображением объектов со всех обзорных камер, причем мнемонические изображения объектов рисуются именно в том месте на плане, где они на самом деле находятся (автомобиль, проезжающий по дороге, будет двигаться по дороге и на плане). На карте условно изображены области видимости камер, при указании конкретной зоны с помощью устройства ввода-вывода можно выбрать камеру, которая этой зоне соответствует. Данные функции доступны после процедуры сопряжения неподвижных видеокамер с картой.

Помимо функций наблюдения за охраняемой территорией в терминальном модуле для тех операторов, которым даны соответствующие полномочия, доступны функции просмотра архивов и функции редактирования фильтров событий (датчиков тревог).

Программы настройки. Перед запуском систем семантического сжатия в эксплуатацию необходима предварительная настройка, большая часть которой связана с определением ориентации камер в пространстве и взаимной ориентацией, т.е. с процедурами сопряжения, определёнными в п. 1.1. Перечислим программы настройки аналитических видеосистем.

1. Программа калибровки поворотных видеокамер. Программа позволяет выполнить процедуру сопряжения неподвижных и поворотных видеокамер в полуавтоматическом режиме. Процедура сопряжения описана в п. 2.6.
2. Программа предварительной настройки сопряжения неподвижных видеокамер между собой и планом местности. Процедура сопряжения описана в п. 2.3 и п. 2.4.
3. Программа настройки параметров семантической обработки сигналов видеокамер.
4. Редактор списка операторов видеосистемы. Программа предназначается для ввода, редактирования и удаления операторов, обладающих полномочиями для работы с терминалами. Для каждого оператора определены пароли и список доступных функций терминального модуля. Например, некоторым операторам можно запретить ввод и изменение тревожных датчиков (фильтров событий).

Рассмотрим более подробно модули, в которых задействованы алгоритмы распознавания, сопряжения и синхронизации.

Схема синхронизации обработчиков видеосигнала. Прежде всего, опишем основные модули аналитических видеосистем — обработчики неподвижных и поворотных камер.

Семантический обработчик изображений мастер-камер выделяет движущиеся объекты в поле зрения видеокамеры, проверяет, нет ли событий соответствующих тревожным фильтрам, сохраняет видеосигнал в архив. Схема алгоритма обработчика

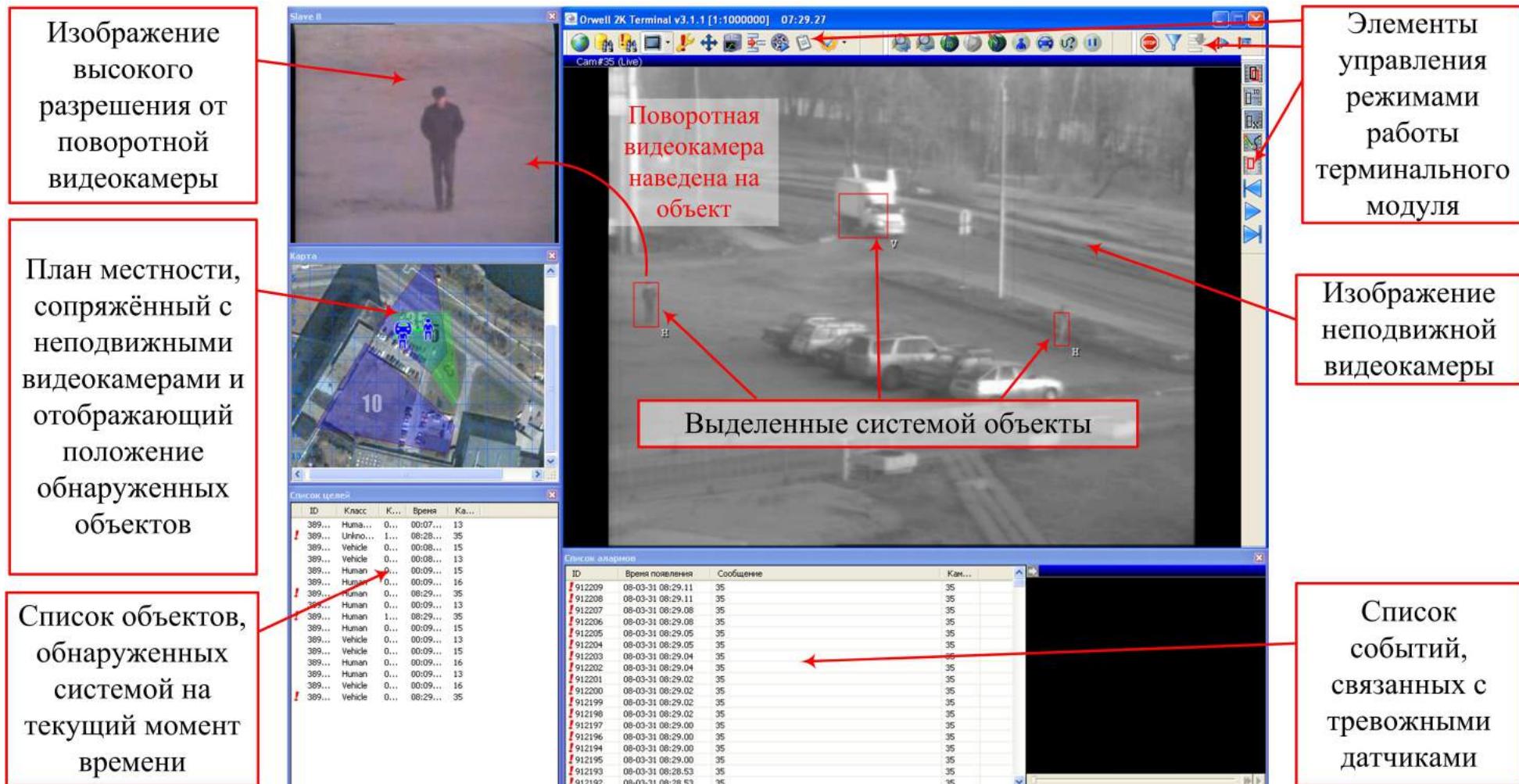


Рисунок 2.2. Внешний вид терминального модуля.

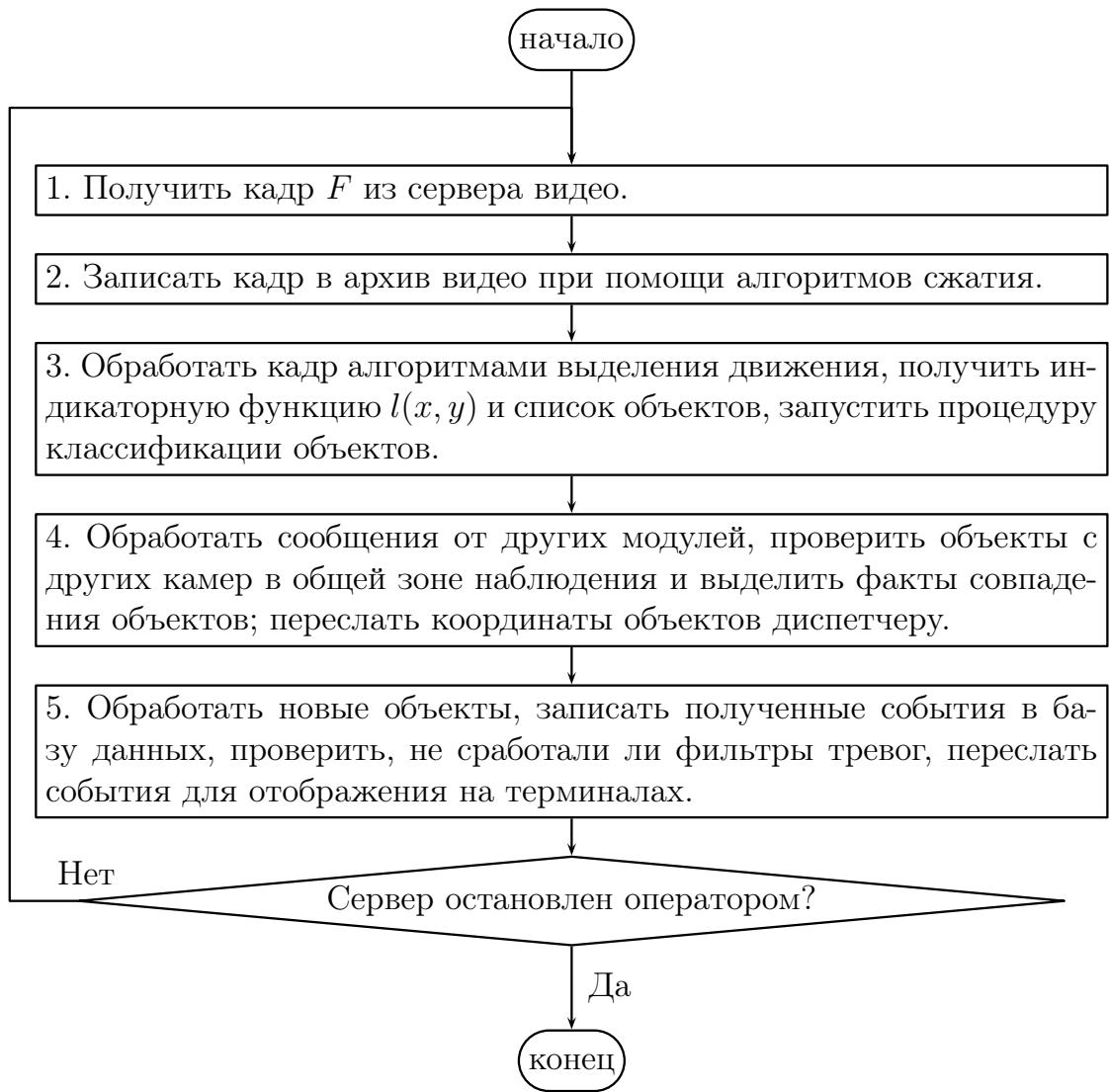


Рисунок 2.3. Блок-схема алгоритма обработчика видеосигнала мастеров.

мастер камеры изображена на рис. 2.3.

На шаге 1 и 2 происходит получение кадра от сервера видео в форме (1.1) и сохранение в сжатой форме в архиве видео, соответственно. На шаге 3 выполняется алгоритм выделения движения, см. определение 1.1. Информация о выделенных движущихся объектах поступает от алгоритма выделения движения в форме нескольких параметров:

- 1) порядковый номер выделенного объекта n_{id} ;
- 2) минимальный прямоугольник $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$, ограничивающий положение объекта в координатах изображения кадра F (задается четырьмя параметрами, как следует из записи);
- 3) индикаторная функция $l(x, y)$, определяющая положение каждой точки выделенного объекта с номером n_{id} следующим образом: точка с координатами (x, y) принадлежит объекту в том и только том случае, если $l(x, y) = n_{id}$ (если точка (x, y) не принадлежит ни одному выделенному объекту, то $l(x, y) = -1$);

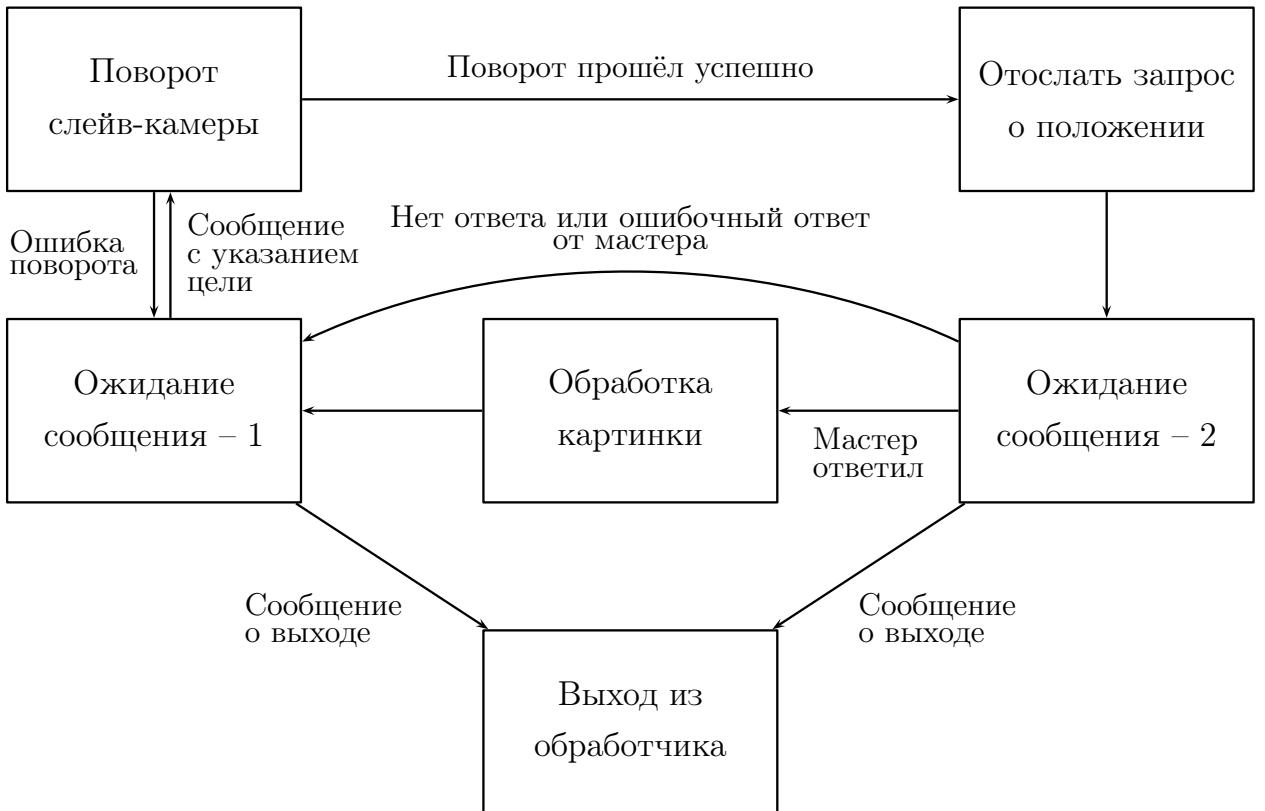


Рисунок 2.4. Граф перехода между состояниями для обработчика слейва.

4) параметры классификации и распознавания: номера категорий, к которым принадлежит принадлежит выделенный объект.

На шаге 4 выполняются основные процедуры синхронизации (см. определение 1.4): производится сопоставление объектов, полученных с помощью сообщений от обработчиков сопряжённых мастер-камер и информация о выделенных объектах отправляется модулю-диспетчеру неподвижных и поворотных камер (о назначении диспетчера см. ниже). Шаг 5 необходим для передачи данных о выделенных объектах и событиях терминальным модулям с целью визуализации и для сохранения этих событий в базе данных. Обработчик видеосигнала с мастер-камер циклически обрабатывает последовательность кадров до тех пор, пока сервер распознавания не завершит работу по указанию оператора. Частота обработки зависит от вычислительной мощности ЭВМ и количества модулей обработчиков, запущенных на ЭВМ одновременно.

Обработчики сигналов слейв-камер предназначены для выполнения функции наведения поворотных камер на движущиеся объекты. Посредством сервера видео модуль-обработчик поворотной камеры формирует команды управления, необходимые для поворота оптической оси камеры и изменения параметров увеличения. Работу обработчиков слейвов можно наглядно представить в форме детерминированного конечного автомата ([125]), функция переходов которого представлена в виде графа на рис. 2.4.

Множество состояний рассматриваемого конечного автомата является процедурой-

ми обработчика. Всего существует шесть состояний, они отображены в прямоугольниках на рис. 2.4. Алфавит конечного автомата представляет набор выходных данных процедур, соответствующих состояниям, в зависимости от результата на выходе происходит переход в то или иное состояние, то есть выполнение той или иной процедуры. Множество символов алфавита состоит из подписей на стрелках изображенного графа перехода, в нём всего семь элементов, включая «пустой» символ. Начальным является состояние «ожидание сообщения – 1», конечным — «выход из обработчика».

«Ожидание сообщения – 1» предварительно отсылает модулю-диспетчеру сообщение о том, что обработчик готов к получению задания и находится в режиме бесконечного ожидания. Затем процедура ждёт первого сообщения, которое имеет отношение к обработчику, все остальные сообщения игнорируются. Сообщения, принимаемые обработчиком, могут быть указанием объекта-цели от диспетчера, либо командой завершить работу.

Процедура «поворот камеры» передаёт через сервер видео сигналы управления на слейв-камеру, необходимые для наведения на объект-цель. Параметры для этой команды передаются от диспетчера, то есть диспетчер ответственен за расчёт упреждения наведения на объект. Время выполнения команды зависит от конкретного поворотного устройства, а также от величин углов поворота и смещения положений увеличения и фокусировки. Максимальное время исполнения поворота устанавливается равным 4с, поскольку в противном случае из-за большой погрешности упреждения (см. п. 3.4) наведение на движущийся объект невозможно.

В случае успешного выполнения поворота запускается процедура «отослать запрос о положении», которая запрашивает текущее положение объекта у обработчика мастера, с которого поступил сигнал-изображение объекта-цели. Затем в процедуре «ожидание сообщения – 2» некоторое время (несколько секунд) ожидается ответ. Время ожидания выбирается таким, чтобы оно значительно превосходило возможные задержки передачи данных по сети и задержки приёма сообщения мастером.

Наконец, процедура «обработки картинки» сохраняет полученный сразу после поворота снимок объекта в архиве, предварительно проанализировав, насколько правильно было дано упреждение, исходя из того, какое положение объекта было получено от мастера. Поскольку обработчик мастера каждый кадр обрабатывает сообщения, то снимок и информация о положении объекта не сильно рассинхронизированы. В процедуре может быть запущен алгоритм, уточняющий классификацию объекта на основе более детального, чем на мастере, изображения.

Исходя из того, как закреплены поворотные и обзорные видеокамеры, существует возможность определить, какие обзорные камеры (мастера) будут сопряжены с какой-либо поворотной камерой (слейвом) с помощью программы калибровки. Таким

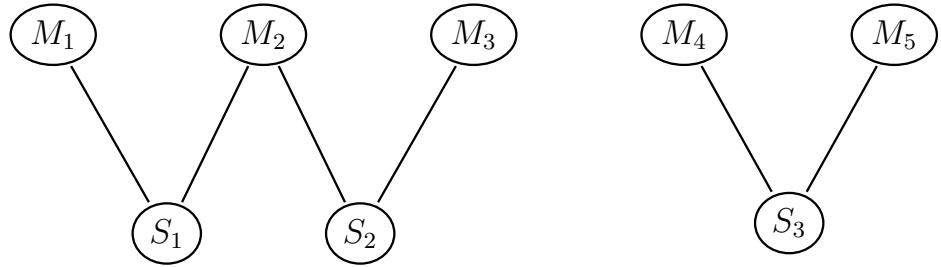


Рисунок 2.5. Пример графа обработчиков системы (M_i — обработчики мастеров, S_j — обработчики слейвов).

образом, все обработчики мастеров и слейвов системы являются вершинами графа, в котором ребра между вершинами присутствуют в том и только том случае, если пара вершин соответствует паре мастера и слайва, сопряжённых между собой. Между двумя слейвами и между двумя мастерами не может быть рёбер, поэтому граф обработчиков двудольный. Такой граф мы назовём *графом обработчиков* аналитической видеосистемы.

Определение 2.6. Кластером является изолированный подграф графа обработчиков, то есть такой подграф, для которого любое ребро из графа обработчиков, имеющее на конце вершину кластера, присутствует и в графе кластера.

Например, для графа, изображённого на рис. 2.5, подмножества вершин $\{M_1, M_2, M_3, S_1, S_2\}$, $\{M_4, M_5, S_3\}$ и их объединение могут образовать кластер, а вершины $\{M_1, M_2, M_3, S_1, S_3\}$ — не могут, так как в подграф из этих вершин не войдут рёбра M_2S_2 , M_3S_2 , M_4S_3 , M_5S_3 .

Одна из целей аналитической видеосистемы — по возможности получить детальные изображения для всех объектов, появившихся в зоне наблюдения. Для достижения максимума детальных изображений необходим модуль диспетчера, который организует общую последовательность наведения поворотных камер.

Модуль диспетчера запускается для кластера видеокамер, и принимает от мастер-камер сообщения о положении объектов. На основе этого массива данных диспетчер формирует задания для обработчиков слейв-камер. Описание алгоритмов синхронизации диспетчера см. в п. 3.1.

В заключение представим общую схему, отображающую описанные взаимодействия сопряжённых модулей и их синхронизацию в кластере аналитической видеосистемы (см. рис. 2.6).

2.2. Теоретический анализ ограничений, вытекающих из пропускной способности сети и быстродействия устройств обработки

Изображённая на рис. 2.6 модель взаимодействия модулей семантического сжатия подразумевает, что количество обработчиков слейвов и мастеров, которые входят в один кластер, ограничено, несмотря на то, что отдельные обработчики могут выпол-

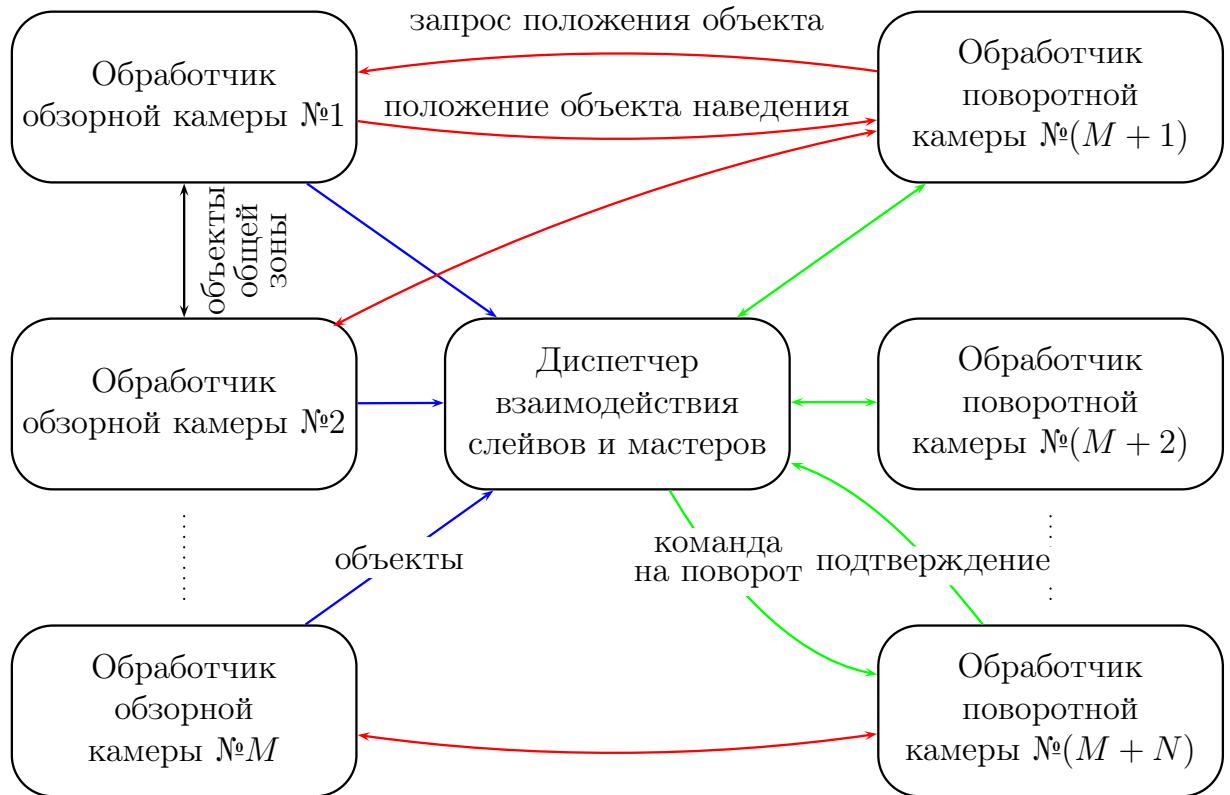


Рисунок 2.6. Схема взаимодействия обработчиков в кластере.

няться на отдельных ЭВМ, и вычислительных ресурсов достаточно для работы сколь угодно большого количества обработчиков в режиме без сопряжения и синхронизации с остальными модулями при условии достаточного количества вычислительных устройств. Ограничения возникают в силу двух причин: первая — это ограниченная пропускная способность сети, по которой данные поступают к диспетчеру и к обработчикам, вторая — это ограниченная эталонная вычислительная мощность (см. определение 2.5), не позволяющая диспетчеру выполнять необходимые действия для большого количества обработчиков.

Ограничения пропускной способности сети. Синхронизация данных между модулями кластера аналитической видеосистемы производится посредством передачи сообщений в ЛВС. Сообщение — это набор цифровых данных, состоящих из заголовка, определяющего тип передаваемой информации и, собственно, самой информации. Список сообщений, которые решают задачу синхронизации данных в рамках схемы взаимодействия модулей, изображённой на рис. 2.6, приведён в таблице 2.1.

Сообщения НОВАЯ_КАМЕРА, ГДЕ_ТЫ, Я_ЗДЕСЬ, НОВЫЙ_ДИСПЕТЧЕР, АДРЕС_ДИСПЕТЧЕРА применяются при установки связи между модулями кластера с целью выяснить адреса передачи сообщений. Эти сообщения посылаются только при начале работы модулей, поэтому они практически не занимают сетевой канал аналитической видеосистемы. Точно также сообщение ВЫХОД используется только при завершении работы модулей системы, а сообщение РУЧНОЕ_УПРАВЛЕНИЕ высылается только тогда, когда

Сообщение	Описание
ВЫХОД	Сообщение, при получении которого диспетчер и обработчик слейвов заканчивают работу.
НОВАЯ_КАМЕРА	Обработчики слейвов и мастеров посылают это сообщение на всем запущенным модулям системы, чтобы найти диспетчера своего кластера.
АДРЕС_ДИСПЕТЧЕРА	Ответ диспетчера на предыдущее сообщение.
ПЕРЕДАЧА_ОБЪЕКТОВ	Сообщение отсылается диспетчеру обработчиками мастеров со списком объектов в поле зрения мастера после обработки каждого кадра.
НАВЕСТИ_СЛЕЙВ	Сообщение от диспетчера обработчику слейва с указанием точки наведения и/или объекта для наведения.
СЛЕЙВ_НАВЕДЁН	Подтверждение от обработчика слейва для диспетчера о завершении наведения, ошибочном или успешном.
ЗАПРОС_ПОЛОЖЕНИЯ	Запрос текущего положения объекта от слейва мастеру.
ПОЛОЖЕНИЕ_ОБЪЕКТА	Передача положения объекта и его свойств от мастера слейву в ответ на предыдущий запрос.
РУЧНОЕ_УПРАВЛЕНИЕ	Отключить определенные слейвы от управления диспетчера, чтобы перевести их в ручное управление.
ГДЕ_ТЫ	Сообщение высыпается всем запущенным модулям системы, чтобы связаться с определённым обработчиком мастера или слейва.
Я_ЗДЕСЬ	Ответ на сообщение ГДЕ_ТЫ.
НОВЫЙ_ДИСПЕТЧЕР	Сообщение о запуске нового диспетчера. В сообщении пересыпается список камер, которые должны подключиться к новому диспетчеру.

Таблица 2.1. Сообщения компонент сервера распознавания.

оператор переходит в ручной режим управления камерой.

Существенный вклад в использование сетевого канала вносят сообщения ПЕРЕДАЧА_ОБЪЕКТОВ, ЗАПРОС_ПОЛОЖЕНИЯ, ПОЛОЖЕНИЕ_ОБЪЕКТА, НАВЕСТИ_СЛЕЙВ, СЛЕЙВ_НАВЕДЁН, поскольку они высыпаются на каждом шаге циклов обработчиков, отображённых на рис. 2.3 и 2.4.

Для того, чтобы оценить, насколько пропускная способность сети $w_0 = 1\text{Гбит}/\text{с}$ ограничивает количество камер в кластере, введем следующие обозначения:

- 1) M — количество мастер-камер;
- 2) N — количество слейв-камер;
- 3) n — среднее количество объектов на одной обзорной камере;
- 4) w_{obj} — объём данных в байтах, описывающих один объект при передачи от обработчика мастера диспетчеру или от мастера к мастеру в сообщении ПЕРЕДАЧА_ОБЪЕКТОВ;

- 5) k — среднее количество мастеров, с которыми пересекается каждый мастер, и, следовательно, должен передавать объекты своим k соседям;
- 6) w_t — размер сообщения-целеуказания НАВЕСТИ_СЛЕЙВ от диспетчера, указывающее слейву, куда нужно повернуться;
- 7) w_q — объём сообщения ЗАПРОС_ПОЛОЖЕНИЯ о положении объекта от поворотной камеры к обзорной;
- 8) w_a — объём ответа ПОЛОЖЕНИЕ_ОБЪЕКТА от мастера слейву;
- 9) w_f — объём сообщения СЛЕЙВ_НАВЕДЁН о готовности принять следующее задание, передаваемое диспетчеру слейвом;
- 10) f — частота обработки кадров с обзорных камер;
- 11) T_R — среднее время исполнения поворота слейвами.

Будем считать, что мы находимся в наихудших условиях, и все данные пересылаются по сети (при пересылке данных в пределах одной ЭВМ тратится значительно меньшее время, чем при пересылке в ЛВС). Объем данных, пересылаемых в единицу времени, не превосходит

$$w = (1 + k)Mn w_{obj} f + \frac{1}{T_R} N(w_t + w_q + w_a + w_f). \quad (2.1)$$

Первое слагаемое в формуле (2.1) учитывает как пересылку $Mn w_{obj} f$ байт от мастеров диспетчеру, так и пересылку $kMn w_{obj} f$ байт между обработчиками сопряжённых мастер-камер при сопоставлении объектов в общей зоне. Второе слагаемое учитывает пересылку сообщений во время работы обработчика слейва в предположении, что цикл посылки данных совпадает с циклом обработчика слейв-камеры и выполняется $1/T_R$ раз в секунду.

Обычно по экономическим соображениям $N \leq M/3$ (поворотные камеры довольно дороги, в несколько раз дороже обзорных), но мы сделаем оценку для наиболее сложных для аналитической системы условий, предположив, что $M = N$, то есть каждую мастер-камеру обслуживает в среднем один слейв. Тогда максимальное количество мастеров в кластере мы можем получить исходя из того, что при максимальной загрузке обменами сообщений

$$\frac{w}{w_0} = \eta, \quad (2.2)$$

где $\eta = 0,5$ — коэффициент «полезного действия» пересылки по сети: из-за издержек протоколов на физическом и транспортном уровне невозможно добиться максимальной пропускной способности w_0 , например, из-за наличия системной информации помимо основных данных в заголовках пакетов передачи, см. [126, 127]. Получаем

максимальное число обзорных камер в кластере при $M = N$

$$M_{max} = \frac{\eta w_0}{(1+k)nw_{obj}f + \frac{1}{T_R}(w_t + w_q + w_a + w_f)} \quad (2.3)$$

Чтобы передать минимум информации, необходимой для синхронизации распределённых модулей, необходимы следующие объёмы данных:

1) $w_{obj} = 20$ байт: по два байта на каждое из целых чисел $x_{min}, x_{max}, y_{min}, y_{max}$, задающих минимальный прямоугольник изображения объекта, по четыре байта на две вещественные компоненты оценки скорости движения изображения, два байта для номера объекта и два байта для описания свойств объекта, полученных алгоритмами классификации — итого, 20 байт;

2) $w_t = w_q = w_a = w_f = 24$ байта, поскольку у сообщений таблицы 2.1 есть заголовок, составляющий 16 байт, остальные данные описывают специфику сообщения.

Для оценки наибольшего объёма передаваемых данных возьмем максимально возможные значения для k, f, n и $1/T_R$: $f = 25\text{с}^{-1}$, $n = 50$, $T_R = 500\text{мс}$, $k = 3$. Значение $k = 3$ очень велико, поскольку большинство мастер-камер в аналитических видеосистемах не имеют общей зоны обзора. Подставив эти значения в формулу (2.3), получим $M_{max} \approx 650$ для максимального количества камер в *одном* кластере. В нескольких кластерах камер может быть больше, поскольку ограничения рассчитывались для одного компьютера.

Итак, даже в наихудшем случае сетевые ограничения позволяют подключить в один кластер достаточно большое число камер. Ограничения, накладываемые на диспетчер производительностью процессора, сильнее, чем ограничения, накладываемые пропускной способностью сети, как мы выясним далее.

Ограничение вычислительной мощности ЭВМ. Диспетчер примерно f раз в единицу времени получает данные об объектах от каждого из M мастеров, и $1/T_R$ раз в единицу времени сообщения об готовности выполнять задания от N слейвов. Диспетчер работает в режиме ожидания сообщений, после каждого сообщения он может выполнять действия для распределения заданий поворотным камерам.

Как показано в [128] и в п. 3.1, при определённой организации работы диспетчера доля процессорного времени, занимаемая алгоритмами синхронизации диспетчера, задаётся соотношением

$$\delta = fM(C_1n + C_3Nn) + \frac{1}{T_R}N(C_2Mn + C_5N!Mn), \quad (2.4)$$

где значения констант установлены экспериментально для эталонной вычислительной мощности:

$$\begin{aligned} C_1 &= 0,1\text{мкс}, \\ C_2 &= 2\text{мкс}, \\ C_3 &= C_5 = 1\text{мкс}. \end{aligned} \quad (2.5)$$

На практике для диспетчера может быть выделено лишь 10% времени процессора, то есть $\delta < 0, 1$. Подставляя в это соотношение формулу (2.4), получим

$$(f(C_1 + C_5 N) + \frac{N}{T_R} (C_2 \log_2(Mn) + C_5 N!)) Mn < 0, 1. \quad (2.6)$$

Подставив в это неравенство $f = 25\text{с}^{-1}$, $T_R = 500\text{мс}$, $n = 50$ и значения (2.5), а также воспользовавшись неравенством $N \leq M$, можно легко решить (2.6) численно, поскольку левая часть монотонно возрастает по M и по N . Перебор осуществляется следующим образом: сначала перебираем значения N в порядке возрастания, начиная с 1, затем проверяем неравенство (2.6) для всех $M : N \leq M$ при зафиксированном N . Перебор заканчивается на том значении N , при котором неравенство не выполняется уже при $M = N$. Решениями будут натуральные числа, удовлетворяющие соотношению

$$\begin{cases} \{N = 4, M = 4\}, \\ \{N = 3, 3 \leq M \leq 9\}, \\ \{N = 2, 2 \leq M \leq 14\}, \\ \{N = 1, M \leq 28\}. \end{cases} \quad (2.7)$$

Система (2.7) задает ограничение на число мастеров и слейвов, которое способен обработать один диспетчер в наихудших условиях работы видеосистемы с точки зрения оценки количества обработчиков в одном кластере. Если количество камер в кластере N и M не удовлетворяют приведённым ограничениям, следует разбить систему на большее число кластеров, возможно, пожертвовав при этом некоторыми связями сопряжения мастер-слейв.

2.3. Основы теории алгоритмов сопряжения видеокамер с общей зоной обзора

Введём нумерацию видеокамер и соответствующих им обработчиков. Для сопоставления объектов, выделенных обработчиками сопряжённых обзорных камер в общей зоне видимости, требуется достаточно точно задать функцию перевода координат

$$(x', y') = \vec{u}_{ds}(x, y) \quad (2.8)$$

из координат (x, y) изображения мастера s в координаты (x', y') мастер d . Далее в параграфе индексы s и d не указываются, поскольку рассматривается фиксированная пара камер. Введём также обозначения $\vec{u}' = (x', y')$, $\vec{v} = (x, y)$, и будем считать запись (2.8) эквивалентной записи $\vec{u}' = \vec{u}(\vec{v})$.

Исходные данные для преобразования. Поиск функции преобразования координат $\vec{u}(\vec{v})$ можно провести при помощи интерполяции и экстраполяции заданных в некоторых точках опорных значений для данной функции. Мы будем считать, что

нам известны наборы значений координат $\{\vec{u}_i\}_{i=1}^k$ мастера d , в которые переходят точки \vec{v}_i , мастера s при применении преобразования (2.8), то есть $\vec{u}_i = \vec{u}(\vec{v}_i)$, $i \in \overline{1, k}$. Договоримся обозначать координаты векторов \vec{u}_i и \vec{v}_i как (x_i, y_i) и (x'_i, y'_i) , соответственно.

Значения функций \vec{u} сетке (x_j, y_j) задаётся вручную при помощи интерфейса программы настройки сопряжения, изображенной на рис. 2.7. На рисунке изображены две камеры (справа), план местности (в центре) и список групп точек (слева). На плане и камерах указаны пронумерованные группы точек, для удобства отображаемые разными цветами, которые расставляются вручную и как раз определяют сетку $\{\vec{v}_i\}$ и значения $\{\vec{u}_i\}$. Например, точки под номерами 1, 2, 3, 4 соответственно зелёного, коричневого, синего и фиолетового цвета, присутствуют на рис. 2.7 на плане и на «зеленой» камере №1 и «фиолетовой» камере №4. Координаты данной четверки на камере №4 считаются сеткой, а на камере №1 — значениями для функции $\vec{u}_{14}(\vec{v})$. Для функции $\vec{u}_{41}(\vec{v})$ смысл значений координат взаимно меняется.

Таким образом мы можем получить необходимые исходные данные (сетки и значения) для вычисления преобразования (2.8). Рассмотрим конкретные методы экстраполяции и интерполяции.

Метод приближенного определения преобразования координат. Рассмотрим три случая расположения двух мастер-камер s и d .

1. Камеры закреплены на значительном расстоянии друг от друга, поверхность зоны наблюдения камер плоская.
2. Камеры закреплены на значительном расстоянии друг от друга, поверхность зоны наблюдения не является плоской, т.е. на экране присутствуют крутые холмы либо индустриальные объекты.
3. Камеры закреплены близко друг от друга настолько, что расстояние между ними пренебрежимо мало по сравнению с расстояниями до зон наблюдения, форма поверхности зоны наблюдения — произвольная.

Первому случаю соответствует рис. 2.8, на котором камера s закреплена в точке C_1 и обозревает зону $M_1M_2M_3M_4$ на плоскости наблюдения, а камера d закреплена в точке C_2 и обозревает зону $M_5M_6M_7M_8$. Введем на плоскости наблюдения $M_1M_2M_8$ систему координат (\tilde{x}, \tilde{y}) . В силу законов геометрической оптики и свойств центральной проекции преобразование координат (x, y) изображения мастера s в координаты (\tilde{x}, \tilde{y}) плоскости наблюдения является проективным. Аналогично, преобразование координат (\tilde{x}, \tilde{y}) в координаты (x', y') изображения с мастера d является проективным. Рассматривая композицию преобразований

$$(x, y) \rightarrow (\tilde{x}, \tilde{y}) \rightarrow (x', y'),$$

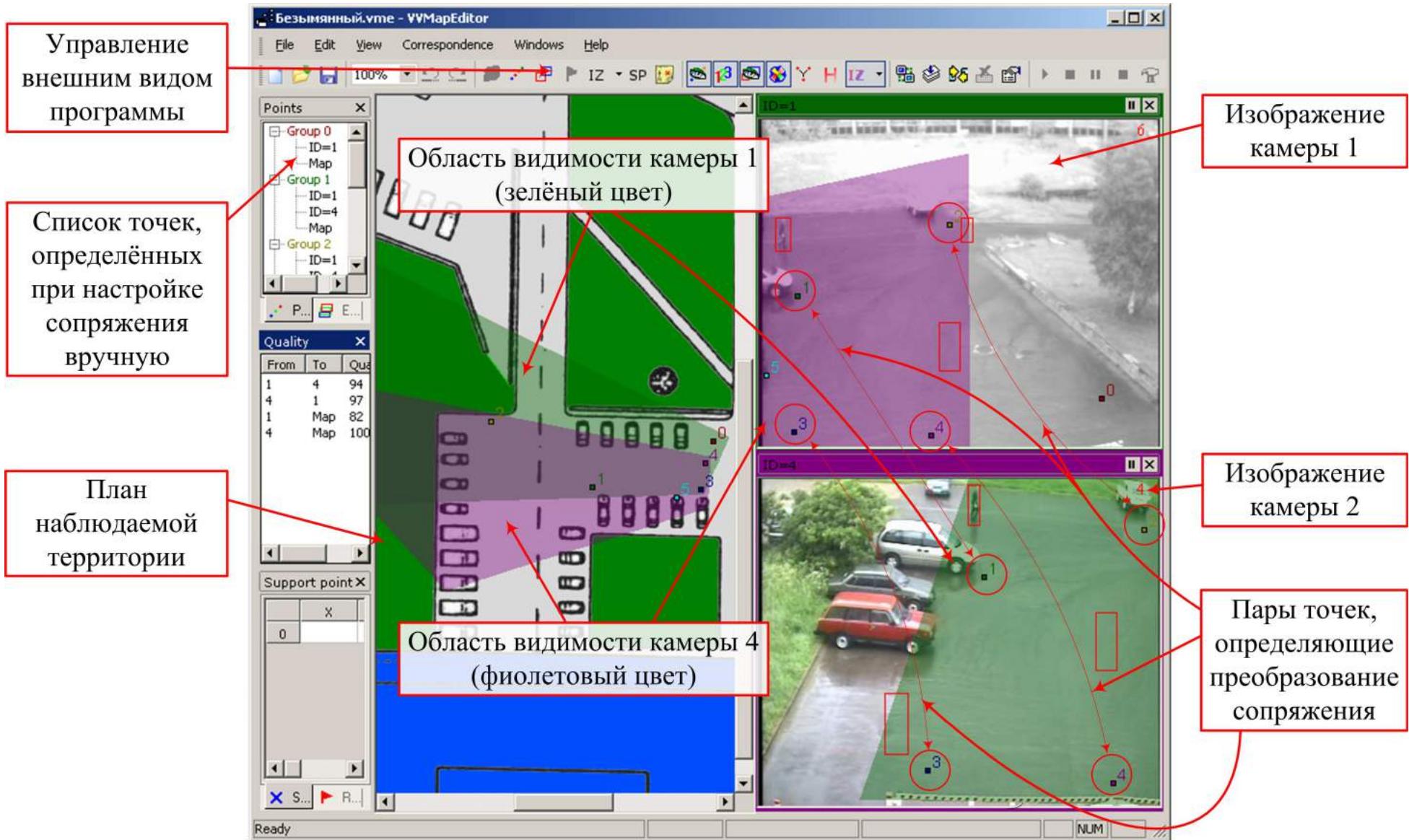


Рисунок 2.7. Интерфейс программы сопряжения изображений.

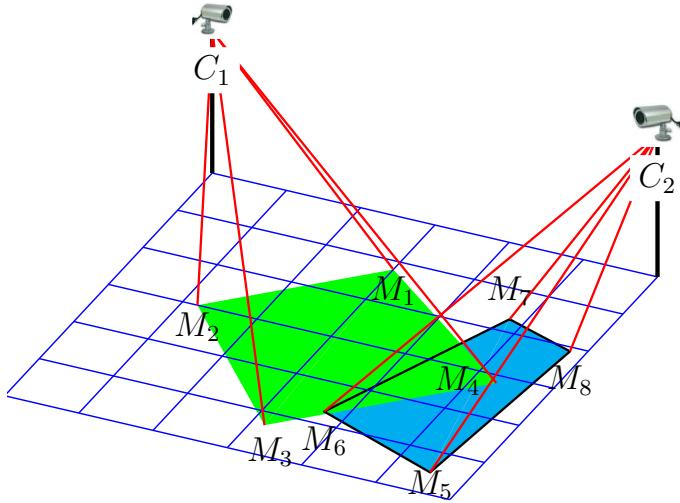


Рисунок 2.8. Два мастера, удалённые друг от друга на расстояние $|C_1C_2|$.

мы получаем, что преобразование $(x, y) \rightarrow (x', y')$ является проективным, и искомая функция $\vec{u}(\vec{v})$ задаётся формулами проективного преобразования:

$$\begin{aligned} x' &= \frac{Ax + By + C}{Gx + Hy + I} \\ y' &= \frac{Dx + Ey + F}{Gx + Hy + I} \end{aligned}, \text{ где } \Delta = \begin{vmatrix} A & B & C \\ D & E & F \\ G & H & I \end{vmatrix} \neq 0. \quad (2.9)$$

Задача поиска функции преобразования \vec{u} сводится к поиску девяти коэффициентов $A, B, C, D, E, F, G, H, I$ в формулах (2.9), если заданы значения \vec{u}_i на сетке $\vec{v}_i, i \in \overline{1, k}$. Поскольку при пропорциональном изменении всех девяти коэффициентов формулы (2.9) переходят в эквивалентные, то без ограничения общности можно считать, что либо $I = 0$, либо $I = 1$. Случай $I = 0$ соответствует переходу прямой $Gx + By + C = 0$, проходящей через точку $(0, 0)$ на мастере s , в бесконечность мастера d . Так может быть, если, например, линия горизонта на изображении мастера s проходит через точку $(0, 0)$, т.е. через верхний левый угол изображения. На практике такое происходит редко, и поэтому случай $I = 0$ отдельно не рассматривается, хотя теоретически к этому нет никаких препятствий: можно подобрать коэффициенты сначала при $I = 0$, затем при $I = 1$ и из двух приближений выбрать наилучшее.

Поскольку значения \vec{u}_i и \vec{v}_i заданы с некоторой погрешностью, то приходится применять метод наименьших квадратов для поиска коэффициентов в формулах (2.9): необходимо решить задачу

$$\min \sum_{i=1}^k \left[\left(x'_i - \frac{Ax_i + By_i + C}{Gx_i + Hy_i + 1} \right)^2 + \left(y'_i - \frac{Dx_i + Ey_i + F}{Gx_i + Hy_i + 1} \right)^2 \right]. \quad (2.10)$$

Формулы (2.9) нелинейны относительно искомых коэффициентов, поэтому задачу (2.10) в общем случае приходится решать одним из методов градиентного спуска,

например, методом Левенберга-Марквардта [110]. Однако в случае, когда $k = 4$, т.е. сетка состоит из четырёх точек, можно рассмотреть систему уравнений

$$\left\{ \begin{array}{l} x'_1 = \frac{Ax_1 + By_1 + C}{Gx_1 + Hy_1 + 1}, \quad y'_1 = \frac{Dx_1 + Ey_1 + F}{Gx_1 + Hy_1 + 1}, \\ x'_2 = \frac{Ax_2 + By_2 + C}{Gx_2 + Hy_2 + 1}, \quad y'_2 = \frac{Dx_2 + Ey_2 + F}{Gx_2 + Hy_2 + 1}, \\ x'_3 = \frac{Ax_3 + By_3 + C}{Gx_3 + Hy_3 + 1}, \quad y'_3 = \frac{Dx_3 + Ey_3 + F}{Gx_3 + Hy_3 + 1}, \\ x'_4 = \frac{Ax_4 + By_4 + C}{Gx_4 + Hy_4 + 1}, \quad y'_4 = \frac{Dx_4 + Ey_4 + F}{Gx_4 + Hy_4 + 1}, \end{array} \right. \quad (2.11)$$

из которой следует система линейных уравнений

$$\left\{ \begin{array}{l} Ax_1 + By_1 + C - x'_1(Gx_1 + Hy_1 + 1) = 0, \\ Dx_1 + Ey_1 + F - y'_1(Gx_1 + Hy_1 + 1) = 0, \\ Ax_2 + By_2 + C - x'_2(Gx_2 + Hy_2 + 1) = 0, \\ Dx_2 + Ey_2 + F - y'_2(Gx_2 + Hy_2 + 1) = 0, \\ Ax_3 + By_3 + C - x'_3(Gx_3 + Hy_3 + 1) = 0, \\ Dx_3 + Ey_3 + F - y'_3(Gx_3 + Hy_3 + 1) = 0, \\ Ax_4 + By_4 + C - x'_4(Gx_4 + Hy_4 + 1) = 0, \\ Dx_4 + Ey_4 + F - y'_4(Gx_4 + Hy_4 + 1) = 0. \end{array} \right. \quad (2.12)$$

Система (2.12) при корректных исходных данных имеет единственное решение. Подставив его в формулы (2.9), получим искомое преобразование.

Можно выписать систему линейных уравнений типа (2.12) и в случае, когда количество точек больше, чем 4, а затем найти решение с минимальной невязкой по методу наименьших квадратов (в данном случае метод наименьших квадратов сводится к решению системы линейных уравнений). Однако это решение будет соответствовать минимуму невязки для уравнений типа (2.12), а не типа (2.11), поэтому его нельзя использовать в качестве окончательного ответа. Тем не менее, полученное решение можно использовать в качестве начального приближения при поиске минимума (2.10) методом градиентного спуска, причём практика показывает, что в подавляющем большинстве случаев при этом достигается удовлетворительный результат, т.е. формулы (2.9) подбираются верно.

Во втором случае установки камер, когда наблюдаемая поверхность не плоская, не существует простого вида формул для преобразования $\vec{u}(\vec{v})$. В общем случае это преобразование не является даже непрерывным: то, что находится в поле зрения камеры s , на камере d может быть скрыто каким-нибудь препятствием. Чтобы получить приближение для преобразования $\vec{u}(\vec{v})$ в данном сложном случае, необходимо составить либо кусочное преобразование, отдельно выделив на камерах «приближенно-

плоские» участки, либо использовать менее точные методы, описанные в п. 2.4. Последний способ не всегда даёт удовлетворительный результат, а выделение участков плоской поверхности связано с объёмным и рутинным подбором исходных данных — на каждом плоском «куске» требуется сетка из не менее, чем четырёх точек, чтобы задать проективное преобразование. Поэтому на практике камеры, размещённые в соответствии со вторым случаем, не сопрягают друг с другом.

Третий случай расположения камер в одной точке замечателен тем, что преобразование координат изображения с одной камеры на другую всегда проективное, независимо от формы наблюдаемой поверхности. В доказательство, рассмотрим плоскость, расположенную перпендикулярно оптической оси камеры s . Любая точка изображения в общей зоне соответствует прямой, проходящей через центр проекции для камеры s и точку зоны, причем эта же прямая соответствует изображению данной точки и для камеры d , поскольку центры проекции у камер s и d находятся в одной точке. Рассматриваемая прямая пересекает введённую нами плоскость в некоторой точке, поэтому мы можем интерпретировать изображение камер s и d как изображение, получаемое с введённой нами плоскости. Таким образом, третий случай расстановки камер сводится к первому.

Таким образом, наиболее предпочтительно сопрягать камеры, расположенные в одной точке. В этом случае формулы преобразования координат являются проективными, и можно применить описанный метод поиска преобразования с помощью введённых вручную опорных точек.

2.4. Исследование особенностей преобразования алгоритмов переноса координат изображения с видеокамер на топографический план местности

Аналогия с сопряжением неподвижных камер. Карту или план местности можно рассматривать как изображение камеры, полученное сверху, «со спутника». Поэтому сопряжение карты и камер проходит по схеме, при которой камеры закреплены на некотором расстоянии друг от друга, см. [129]. Назначим изображению карты индекс m , отличный от всех номеров обработчиков камер. Как было установлено в предыдущем параграфе, если поверхность плоская, то преобразование координат изображения мастер-камеры s в координаты изображения карты m $\vec{u}_{sm}(\vec{v})$, также как и преобразование с карты на камеру $\vec{u}_{ms}(\vec{v})$, задаётся формулами (2.9), коэффициенты которых находятся при решении задачи (2.10). Исходные данные в виде сетки и соответствующих её узлам опорным значениям вводятся с помощью интерфейса программы сопряжения (см. рис. 2.7), также как при сопряжении мастер-камер.

Учёт погрешностей проективного преобразования. Можно выделить три основных проблемы, которые могут встретиться при подобном подходе. Во-первых,

местность, которую обозревает камера, часто бывает неровной, т.е. неплоской. Вторых, план местности может быть нарисован схематично, без соблюдения масштабов и пропорций. В обоих случаях проективное преобразование не годится в качестве приближения. В-третьих, в отличие от изображений камер, иногда сложно сопоставить изображение камеры и карты, например, если камера направлена на какое-нибудь поле, которое отображено на плане монотонным пятном. В этом случае необходимо дополнительно решить, каким образом мы зададим исходную сетку значений для подбора преобразований с карты на камеру и наоборот.

Для решения первых двух проблем в системе можно использовать два подхода. Первый подход заключается в том, что неровность местности моделируется дополнительными добавками (w и h — высота и ширина изображения карты/камеры, измеряемая в точках изображения):

$$\begin{aligned} x' &= \frac{Ax + By + C}{Gx + Hy + I} + \sum_{i=1}^4 \left(C_i \cos\left(\frac{x\pi}{w}i\right) + S_i \sin\left(\frac{x\pi}{w}i\right) \right), \\ y' &= \frac{Dx + Ey + F}{Gx + Hy + I} + \sum_{i=1}^4 \left(C'_i \cos\left(\frac{y\pi}{h}i\right) + S'_i \sin\left(\frac{y\pi}{h}i\right) \right), \end{aligned} \quad (2.13)$$

т.е. неровность земной поверхности приближается первыми четырьмя членами ряда Фурье (см. [130]). Сначала все дополнительные шестнадцать коэффициентов C_i , C'_i , S_i , S'_i , где $i \in \overline{1, 4}$, считаются равными нулю, и решается задача (2.10) способом, описанным в предыдущем пункте. Затем полученное решение принимается в качестве нулевого приближения для нового градиентного спуска, уже с участием дополнительных слагаемых.

Второй способ обойти нелинейность поверхности и неточность графического представления карты заключается в кусочно-линейном приближении преобразования $\vec{u}(\vec{v})$. Для этого мы на сетке значений $\{\vec{v}_i\}$ проводим триангуляцию Делоне, а затем в каждом треугольнике полученной триангуляцией с вершинами в точках \vec{v}_a , \vec{v}_b , \vec{v}_c определяем преобразование $\vec{u}(\vec{v})$ как линейную функцию, подчиняющуюся условиям

$$\begin{cases} \vec{u}(\vec{v}_a) = \vec{v}_a \\ \vec{u}(\vec{v}_b) = \vec{v}_b \\ \vec{u}(\vec{v}_c) = \vec{v}_c \end{cases} \quad (2.14)$$

Таким образом мы определим преобразование $\vec{u}(\vec{v})$ в каждом треугольнике триангуляции, т.е. внутри выпуклой оболочки сетки $\{\vec{v}_i\}$. Чтобы вычислить преобразование в точке \vec{v} , нужно найти треугольник, в котором эта точка находится, а затем вычислить линейную функцию, найденную с учётом условий (2.14). Для точек вне выпуклой оболочки можно воспользоваться формулами (2.13), найденными методом

первого подхода. При этом на границе выпуклой оболочки возможны большие скачки найденного преобразования. Чтобы избежать их, рекомендуется добавлять точки сетки в углы экрана. Тогда выпуклая оболочка закроет всё изображение камеры.

Оба метода обладают значительной погрешностью и плохо учитывают возможность разрыва внутри области определения преобразования. Разрыв в данном случае может, например, возникнуть, когда камера смотрит на край обрыва: на изображении камеры соседние точки края обрыва стоят близко, а на плане должен быть разрыв, который может измеряться сотнями метров. Чтобы учесть такие разрывы, на карте вводятся «тёмные зоны» для каждой камеры в форме многоугольников, т.е. зоны, которые камера не видит из-за препятствий. Преобразование с карты на мастер $\vec{u}_{ms}(\vec{v})$ в таких зонах неопределено, а если точка с мастера попадает в тёмную зону, то срабатывает дополнительная функция: преобразование $\vec{u}_{sm}(\vec{v})$ выдаёт ближайшую точку границы зоны. Приведенные эвристики позволяют приблизительно отображать траектории выделенных движущихся объектов на карте. При сопряжении двух мастеров такой подход не применим из-за сложности ручного ввода тёмных зон для двух камер.

Методы ввода исходных данных. Для того, чтобы составить сетку преобразований между камерой и картой в условиях, когда на карте нет видимых на изображении ориентиров, необходима техническая система измерения мировых координат. Удобнее всего воспользоваться системой позиционирования с помощью спутника, например, «Гланас» или GPS, однако точность определения координат в несколько метров на данный момент предлагают лишь военные системы, для гражданского же варианта точность составляет несколько десятков метров, что сравнимо с геометрическими размерами зоны наблюдения. Поэтому задать сетку преобразования с помощью гражданской системы спутниковой навигации затруднительно.

Если нам известны физические координаты не менее четырёх точек-ориентиров в некоторой мировой трёхмерной декартовой системе координат, причем, во-первых, положение этих точек на карте известно, во-вторых, эти ориентиры просматриваются из наблюдаемой зоны, и, в-третьих, ориентиры не лежат в одной плоскости, то при составлении сетки преобразований можно обойтись лазерным дальномером и участием двух человек[131]: оператора компьютера и наблюдателя. Обозначим мировые координаты ориентиров как (x_i, y_i, z_i) , $i \in \overline{1, 4}$. Ориентирами могут служить участки стен зданий, вершины холмов, т.д. Пусть наблюдатель находится в точке с мировыми координатами (x, y, z) . С помощью дальномера он может измерить расстояние l_i , $i \in \overline{1, 4}$, до каждого из ориентиров соответственно. Координаты (x, y, z) задаются системой

уравнений

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 = l_1^2, \\ (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 = l_2^2, \\ (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 = l_3^2, \\ (x - x_4)^2 + (y - y_4)^2 + (z - z_4)^2 = l_4^2, \end{cases}$$

которая эквивалентна системе

$$\begin{cases} x_1x + y_1y + z_1z - R^2/2 = (R_1^2 - l_1^2)/2, \\ x_2x + y_2y + z_2z - R^2/2 = (R_2^2 - l_2^2)/2, \\ x_3x + y_3y + z_3z - R^2/2 = (R_3^2 - l_3^2)/2, \\ x_4x + y_4y + z_4z - R^2/2 = (R_4^2 - l_4^2)/2, \end{cases} \quad (2.15)$$

где $R^2 = x^2 + y^2 + z^2$, $R_i^2 = x_i^2 + y_i^2 + z_i^2$, $i \in \overline{1, 4}$. Последняя система является линейной относительно четвёрки (x, y, z, R^2) , причем её определитель

$$\begin{aligned} \begin{vmatrix} x_1 & y_1 & z_1 & -\frac{1}{2} \\ x_2 & y_2 & z_2 & -\frac{1}{2} \\ x_3 & y_3 & z_3 & -\frac{1}{2} \\ x_4 & y_4 & z_4 & -\frac{1}{2} \end{vmatrix} &= \begin{vmatrix} x_1 & y_1 & z_1 & -\frac{1}{2} \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 & 0 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 & 0 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 & 0 \end{vmatrix} = \\ &= \frac{1}{2} \cdot \begin{vmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \end{vmatrix} \neq 0, \end{aligned}$$

если точки-ориентиры не лежат на одной плоскости. Таким образом, решая систему (2.15) относительно (x, y, z, R^2) , мы получим единственное решение. Если ориентиры выбраны удачно, то данная система будет устойчива ко входным данным, и мы вправе ожидать небольшого превышения окончательной погрешности координат (x, y, z) над погрешностью измерения величин l_i , x_i , y_i , z_i . Поскольку погрешность современных дальномеров примерно 0,5–1 метр, то и результат (x, y, z) должен отклоняться от действительного не более, чем на несколько метров.

Итак, наблюдатель измерил расстояния до ориентиров l_i . Теперь он может сообщить их, например, по радио, оператору, который находится у терминала и видит наблюдателя на изображении камеры. Оператор отмечает на камере точку положения наблюдателя и вносит измеренные до ориентиров расстояния в программу. Программа решает систему (2.15) и находит мировые координаты (x, y, z) точки наблюдателя, а также вычисляет положение этой точки на карте, поскольку по известным координатам ориентиров и известному их положению на карте не сложно составить линейные формулы параллельной проекции мировых координат в координаты карты. Таким образом получается пара узла сетки и значения в этом узле (\vec{v}, \vec{u}) для искомого преобразования.

Для повышения точности можно использовать измерение расстояний до большего, чем четыре, количества ориентиров. Тогда система (2.15) будет состоять из большего числа уравнений, и решать её нужно будет методом наименьших квадратов.

Вывод уравнения линии горизонта[132]. Вычислив преобразование с изображения мастер-камеры на план местности по формулам (2.9), мы найдём коэффициенты A, B, C, D, E, F, G, H . В случае, если наблюдаемая поверхность — плоская, то уравнение линии горизонта на изображении мастер-камеры задаётся соотношением

$$Gx + Hy + 1 = 0, \quad (2.16)$$

поскольку точки с координатами, удовлетворяющими данному уравнению, переходят в бесконечность при расчётах по формулам (2.9). Уравнение (2.16) может быть использовано для дополнительной фильтрации движущихся объектов, поскольку в обычных условиях наблюдение за объектами над горизонтом не ведётся.

2.5. Оценка точности преобразований координат объектов семантического анализа при сопряжении изображений

Поскольку исходные точки для преобразования сопряжения (2.9) задаются с некоторой погрешностью, то коэффициенты преобразования, полученные при решении задачи минимизации (2.10) или при решении системы (2.12) также содержат в себе неточность, и, следовательно, преобразование тоже содержит погрешность.

Для оценки влияния погрешности входных данных на решение задачи (2.10) рассмотрим случай, когда опорных точек всего 4, и поиск коэффициентов сводится к решению линейной системы (2.12).

Введём обозначения

$$M = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1 x_1 & -x'_1 y_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x'_2 x_2 & -x'_2 y_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x'_3 x_3 & -x'_3 y_3 \\ x_4 & y_4 & 1 & 0 & 0 & 0 & -x'_4 x_4 & -x'_4 y_4 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1 x_1 & -y'_1 y_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -y'_2 x_2 & -y'_2 y_2 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -y'_3 x_3 & -y'_3 y_3 \\ 0 & 0 & 0 & x_4 & y_4 & 1 & -y'_4 x_4 & -y'_4 y_4 \end{bmatrix}, \quad r = \begin{bmatrix} A \\ B \\ C \\ D \\ E \\ F \\ G \\ H \end{bmatrix}, \quad b = \begin{bmatrix} x'_1 \\ x'_1 \\ x'_1 \\ x'_1 \\ y'_1 \\ y'_1 \\ y'_1 \\ y'_1 \end{bmatrix}, \quad (2.17)$$

тогда система (2.12) записывается в виде $Mr = b$. Известно (см. [133]), что вариация решения при неустойчивых данных для линейных систем задаётся формулой

$$\Delta r = M^{-1}(\Delta b - \Delta Mr), \quad (2.18)$$

где r — точное решение системы, а $\Delta b, \Delta M$ — дифференциалы при отклонении исходных точек от действительного положения.

С другой стороны, обозначив $L = Gx + Hy + 1$ и дифференцируя формулы (2.9), получим

$$\begin{aligned}\Delta x' &= \frac{\partial x'}{\partial A} \Delta A + \frac{\partial x'}{\partial B} \Delta B + \frac{\partial x'}{\partial C} \Delta C + \frac{\partial x'}{\partial G} \Delta G + \frac{\partial x'}{\partial H} \Delta H = \\ &= \frac{x}{L} \Delta A + \frac{y}{L} \Delta B + \frac{1}{L} \Delta C - \frac{x'x}{L} \Delta G - \frac{x'y}{L} \Delta H = \\ &= \frac{1}{L} (\tilde{x}, \Delta r), \text{ где } \tilde{x} = (x, y, 1, 0, 0, 0, -x'x, -x'y).\end{aligned}\quad (2.19)$$

Аналогично, $\Delta y' = \frac{1}{L} (\tilde{y}, \Delta r)$, где $\tilde{y} = (0, 0, 0, x, y, 1, -y'x, -y'y)$. Подставляя (2.18) в (2.19), получим

$$\begin{aligned}\Delta x &= \frac{(\tilde{x}, M^{-1}(\Delta b - \Delta Mr))}{L}, \\ \Delta y &= \frac{(\tilde{y}, M^{-1}(\Delta b - \Delta Mr))}{L}.\end{aligned}\quad (2.20)$$

Полученные формулы линейны относительно дифференциалов-погрешностей исходных данных, поэтому, вычислив точную обратную матрицу M^{-1} и решение r , легко найти максимальные отклонения Δx и Δy . Согласно теории линейного программирования, максимум погрешности соответствует одному из наборов максимальных по модулю отклонений от истинных значений введённых координат $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x'_1, y'_1), (x'_2, y'_2), (x'_3, y'_3), (x'_4, y'_4)$, поскольку именно они являются вершинами выпуклого множества, на котором ведётся поиск максимума.

Следствием данного факта является то, что если все координаты, кроме одной, для определённости x_1 , заданы точно, то можно проверить значения погрешностей по формулам (2.20) только для минимального и максимального значения отклонения x_1 от точного, а затем выбрать максимальные значения из двух полученных отдельно для Δx и Δy , чтобы получить ожидаемый максимум ошибки в фиксированной точке (x, y) .

Отметим, что максимальная погрешность для преобразования сопряжения сильно зависит от конфигурации точек в исходных данных, так как обусловленность матрицы M системы линейных уравнений (2.12), то есть величина отношения собственных чисел матрицы $M^T \cdot M$, определяет устойчивость численного решения данной системы. Например, если выбрать точки изображения близкими к какой-либо прямой, то обусловленность матрицы будет велика, и, следовательно, велика будет погрешность численного решения систем линейных уравнений, поскольку погрешность решения системы линейных уравнений в первом приближении пропорциональна обусловленности [134]. Таким образом, при оценке погрешности численного решения с помощью формул (2.20) необходимо также оценивать погрешность численного обращения матрицы M^{-1} , которая определяется как раз обусловленностью данной матрицы.

2.6. Разработка алгоритмов автоматизации настройки сопряжения неподвижных и поворотных видеокамер

Команды управления поворотными камерами. Каждый тип поворотных камер поддерживает свой протокол команд для управления перемещением, но, несмотря на различный формат управляющих сигналов, существует более-менее общий набор команд, поддерживаемых всеми протоколами. Далее приведен список наиболее часто встречающихся команд с описанием.

1. **ПОВЕРНУТЬ**(*куда*, v_α , v_β): двигаться в одном из восьми направлений, задаваемых параметром *куда*, с параметрами скорости v_α и v_β . Все камеры на поворотных устройствах, как правило, могут вращаться вокруг оси подвеса, по азимуту, и в плоскости, проходящей через линию подвеса, по склонению. Параметр *куда* принадлежит множеству значений *влево* (уменьшение азимута), *вправо* (увеличение азимута), *вверх* (уменьшение склонения), *вниз* (увеличение склонения) и их комбинациям *вверх_влево*, *вверх_вправо*, *вниз_влево*, *вниз_вправо*, означающих одновременное изменение азимута и склонения. Параметры v_α и v_β являются целыми положительными номерами скоростей, которые исполнительное устройство поворотной видеокамеры переводит в скорость изменения, соответственно, азимута и склонения, измеряемую в градусах в секунду. Номер скорости переводится в величину физической скорости согласно спецификации устройства.

2. **УВЕЛИЧЕНИЕ**(*куда*, v_z): если *куда*=*приблизить*, то увеличить угол зрения поворотной камеры, то есть увеличить фокусное расстояние объектива. Если *куда*=*удалить*, то уменьшить угол зрения. Скорость изменения задается целым положительным параметром v_z , обычно физический смысл данного параметра в спецификациях устройств не указывается.

3. **ФОКУС**(*куда*, v_f): команда настройки фокуса. Изменяет положение фокальной плоскости по отношению к чувствительному элементу камеры (ПЗС-матрице), не изменяя при этом (или изменяя незначительно) фокусное расстояние, то есть увеличение. Если *куда*=*фокус+*, то фокальная плоскость движется в одном из направлений по отношению к ПЗС-матрице, если *куда*=*фокус-*, то фокальная плоскость движется в обратном направлении. Скорость движения задается номером скорости v_f .

4. **ОСТАНОВИТЬ()**: остановить все движения — изменение углов, фокуса, увеличения. Таким образом, чтобы подвинуться на один градус влево, нужно сначала посмотреть в таблицу скоростей и выбрать в ней определенную скорость, например, $v_u = 2$. Пусть она соответствует физической скорости 10°с^{-1} . Тогда необходимо выполнить команду **ПОВЕРНУТЬ**(*влево*, $2,0$), затем сделать паузу в 100мс, затем послать камере команду **ОСТАНОВИТЬ()**.

5. ЗАПОМНИТЬ_ПОЗИЦИЮ(n): запомнить позицию с номером n . Во внутренней памяти некоторых поворотных устройств могут быть сохранены текущие значения азимута и склонения, положение увеличения и фокусировки (фокальной плоскости), а также некоторые настройки, например, степень раскрытия диафрагмы. Обычно количество позиций в памяти поворотного устройства ограничено. Обозначим через N_p количество доступных позиций.

6. ВЫЗВАТЬ_ПОЗИЦИЮ(n): восстановить позицию с номером n . При выполнении команды поворотная видеокамера повернется в ту позицию и установит на объективе такие увеличение и фокусировку, которые были сохранены командой ЗАПОМНИТЬ_ПОЗИЦИЮ(n).

7. ВЫЗОВ_КООРДИНАТ(α, β, γ, f): запись координат в камеру. Некоторые камеры делят диапазон углов азимута и склонения, а также диапазон изменения увеличения и фокусировки на дискретные деления, приписывая каждому некоторую координату. Команда ВЫЗОВ_КООРДИНАТ() позволяет установить камеру в положение с координатами α, β, γ, f для азимута, склонения, увеличения и фокусировки, соответственно.

8. ЧТЕНИЕ_КООРДИНАТ(α, β, γ, f): чтение текущих координат. В переменных α, β, γ, f сохраняется текущее положение поворотной камеры. Смысл параметров аналогичен смыслу параметров предыдущей команды.

Чтобы поворотная видеокамера была приспособлена к работе в аналитической видеосистеме, необходимо, чтобы она поддерживала по крайней мере первые 6 перечисленных команд. Для большинства поворотных камер точность наведения с помощью команд ВЫЗВАТЬ_ПОЗИЦИЮ() или ВЫЗОВ_КООРДИНАТ() меньше 0, 1° по азимуту и склонению, в то время как попытки использовать управление без обратной связи при помощи только первых четырёх команд приводят к тому, что погрешность наведения постоянно возрастает, и уже после нескольких команд составляет больше градуса. Большая погрешность наведения при помощи команд ПОВЕРНУТЬ() / ОСТАНОВИТЬ() обусловлена не только отсутствием возможности проверить текущее положение, но и случайным характером задержек передачи сигналов управления между исполнительным устройством камеры и алгоритмами наведения.

Пара команд ЗАПОМНИТЬ_ПОЗИЦИЮ() / ВЫЗВАТЬ_ПОЗИЦИЮ() работает с положением камеры, которое запоминается поворотным устройством во внутреннем формате данных, и к ним нет доступа извне. Таким образом, эта пара команд даёт возможность точного наведения только в N_p позиций, которые надо предварительно установить вручную. У пары команд ВЫЗОВ_КООРДИНАТ() / ЧТЕНИЕ_КООРДИНАТ() отсутствует такое ограничение, но переход в заданное положение с помощью указания координат обычно происходит медленнее, чем с помощью команд вызова позиций из внутренней памяти.

В аналитической системе семантического сжатия перечисленные команды выполняет модуль сервера видео, поэтому все поворотные камеры, независимо от особенностей протокола, управляются другими программами с помощью приведённой общей формы команд.

Постановка задачи автоматического сопряжения поворотной и неподвижной камер[135]. Модуль обработчика слайва должен выполнять более сложный тип перемещений поворотного устройства, чем перечисленные выше команды управления низкого уровня. Определим этот тип перемещений как результат команды ПЕРЕЙТИ(s, m, x, y, z), где s — номер слайв-камеры, которая выполняет команду, m — номер мастер-камеры, которая сопряжена со слайвом s , (x, y) — координаты изображения мастер-камеры, указывающие точку, которая должна оказаться в центре изображения слайв-камеры после выполнения команды, и z — это коэффициент увеличения изображения слайва по сравнению с изображением мастера после завершения перемещения.

Реализация команды ПЕРЕЙТИ(s, m, x, y, z) полностью решает задачу сопряжения поворотной и неподвижной видеокамер, согласно определению 1.2. Чтобы реализовать данную команду, необходимо сначала построить преобразование, которое переводит величины x, y, z в вектор координат углов поворота $\vec{\tau} = (\alpha, \beta)$ (азимут и склонение) и величины увеличения γ и фокусировки f для заданной пары мастера m и слайва s . Обозначим данное преобразование как \mathbf{a}_m^s : $(\vec{\tau}, \gamma, f) = \mathbf{a}_m^s(x, y, z)$. Затем, после получения координат $(\vec{\tau}, \gamma, f)$, необходимо вызвать подходящие команды перемещения поворотной видеокамеры: либо ВЫЗВАТЬ_ПОЗИЦИЮ(), либо ВЫЗОВ_КООРДИНАТ() с соответствующими параметрами. Как показано далее, эти два варианта существенно отличаются друг от друга.

Основная задача «автоматизированной» настройки заключается в том, чтобы при настройке преобразований сопряжения действия оператора сводились к минимуму. Обычно это означает, что оператор вводит небольшое количество опорных точек преобразований, а далее система экстраполирует заданные значения для всех остальных координат.

Преобразования \mathbf{a}_m^s для наведения слайв-камер задаются в программе калибровки поворотных камер. Найденные преобразования сохраняются в специальном формате в базу данных, поэтому компонент системы может воспользоваться командой ПЕРЕЙТИ().

Методика настройки сопряжения с использованием команды вызова позиции. Камеры, не поддерживающие координатные команды перемещения, ограничены в точности наведения в координаты x, y, z , поскольку существует не более N_p^s наборов координат $\vec{\tau}, \gamma, f$, в которые слайв-камера может переместиться с достаточ-

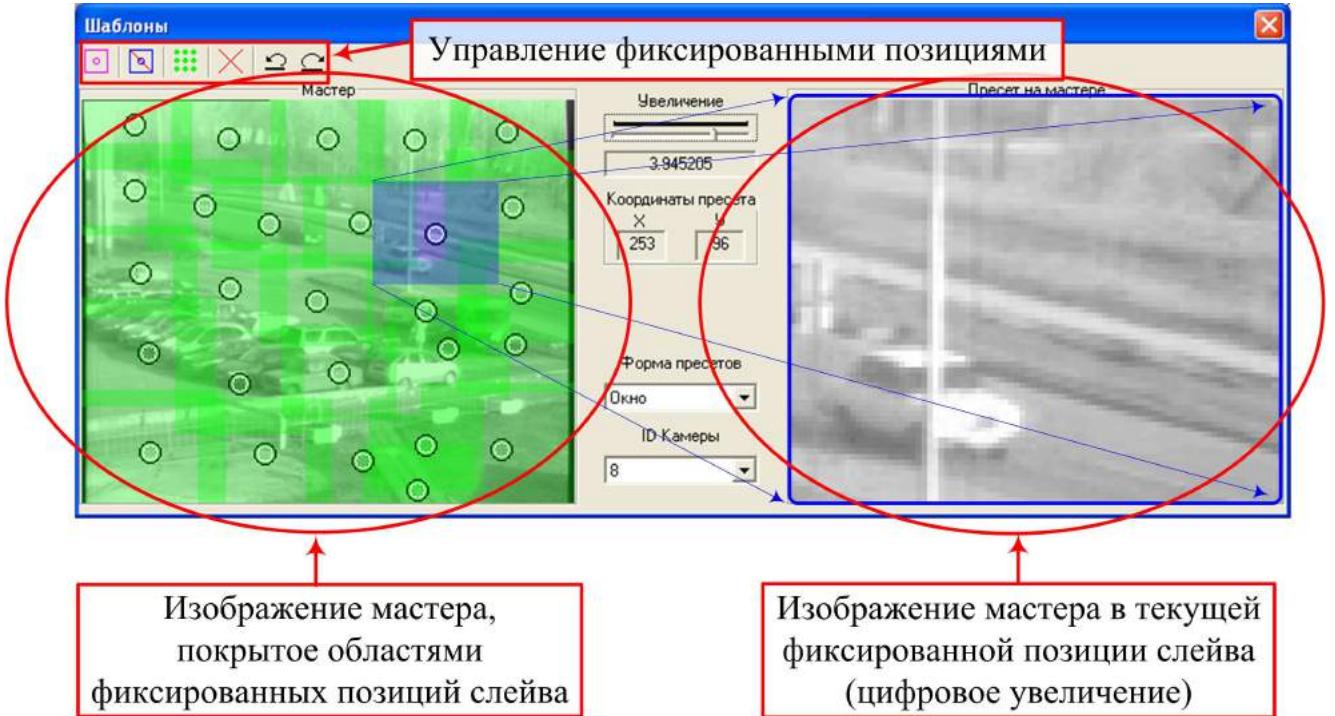


Рисунок 2.9. Диалог для введения координат позиционирования x, y, z .

ной точностью (N_p^s — количество зафиксированных в памяти позиций, поддерживаемое камерой s). Из-за того, что слейв-камера обычно обслуживает сразу несколько мастеров, N_p^s позиций приходится распределять между разными мастерами, что ещё больше ограничивает возможности наведения.

Определим позицию \mathcal{P} из памяти поворотного устройства как тройку $(\vec{\tau}, \gamma, f)$. Соответственно, обозначим множество фиксированных позиций камеры s как $\{\mathcal{P}_i\}_{i=1}^{N_p^s}$. Напомним, что непосредственного доступа к координатам $(\vec{\tau}_i, \gamma_i, f_i)$ позиции \mathcal{P}_i для рассматриваемых камер не существует, у нас есть лишь возможность сохранить текущее положение камеры в i -ую позицию камеры, выполнив команду **ЗАПОМНИТЬ_ПОЗИЦИЮ(i)**, и только затем возвращаться к запомненным в позиции \mathcal{P}_i координатам при помощи команды **ВЫЗВАТЬ_ПОЗИЦИЮ(i)**.

Пусть для определённости первые M_p^s из N_p^s позиций отведены для мастер-камеры m . Для каждой позиции \mathcal{P}_i , $i \in \overline{1, M_p^s}$, мы определим тройку координат x_i, y_i, z_i относительно мастера m . Фактически мы для позиции \mathcal{P}_i выделим на мастере m зону, которую просматривает слейв в данной позиции. Для этой процедуры в программе калибровки существует специальный диалог, внешний вид которого показан на рис. 2.9. В этом диалоге можно либо воспользоваться предопределенной схемой расстановки x_i, y_i, z_i для всех M_p^s позиций, либо расставить зоны вручную. Справа на рисунке показана картинка с мастер-камерой, на которой зелеными кружками обозначены все точки различных позиций. Синий кружок означает текущую точку, синий прямоугольник — зону текущей позиции. При креплении камер следует позаботиться о том, чтобы горизонтальные линии кадра были параллельны горизонту для естественного

восприятия видеоизображения. Если это условие выполнено и для слейва, и для мастера, то зона позиции слейва будет прямоугольником со сторонами, параллельными горизонтали и вертикали изображения, а размер прямоугольника будет ровно в z_i раз меньше размеров картинки мастера по каждому из измерений (z_i соответствует позиции \mathcal{P}_i). Это следует из того, что камеры установлены в одной точке, а увеличение слейва является ничем иным, как сужение угла зрения по сравнению с углом зрения мастера в z_i раз. Слева на диалоге, изображённом на рис. 2.9, цифровым способом сделано увеличение мастера, чтобы показать, каким примерно должно быть изображение со слейва, если его установить в данную позицию.

После установки соответствия между позициями \mathcal{P}_i и координатами на мастере x_i, y_i, z_i , нужно сохранить в этих позициях соответствующие координаты $(\vec{\tau}_i, \gamma_i, f_i)$ с помощью команды **ЗАПОМНИТЬ_ПОЗИЦИЮ(*i*)**. Это можно сделать вручную при помощи интерфейса, изображенного на рис. 2.10. Слева на рисунке — изображение со слейв камер (на нижнее изображение может накладываться изображение с мастером для наглядности совмещения), справа вверху — изображение мастер-камеры с позициями слейва в виде зелёных кружков, справа внизу — цифровое увеличение мастера в районе текущей позиции из памяти слейва. С помощью кнопок «», «Увеличение +/−», «Фокус дальше/ближе» мы посылаем соответственно команды **ПОВЕРНУТЬ()**, **УВЕЛИЧЕНИЕ()**, **ФОКУС()** при нажатии, и команду **ОСТАНОВИТЬ()** при отжатии, управляя тем самым поворотной камерой. С помощью этих кнопок необходимо так повернуть слейв, чтобы нижние картинки совпадали. Тогда можно будет запомнить положение камеры в выделенной позиции \mathcal{P}_i , нажав кнопку «Установить»; тем самым будет выполнено сопоставление координат x_i, y_i, z_i с позицией $\mathcal{P}_i = (\vec{\tau}_i, \gamma_i, f_i)$. Изменяя скорость для команд перемещения, можно выполнять точную и «грубую» подстройку.

Операцию сопоставления x_i, y_i, z_i и $\mathcal{P}_i = (\vec{\tau}_i, \gamma_i, f_i)$ необходимо проделать для всех позиций, задействованных в системе, поэтому при ручном сопряжении по запоминаемым позициям оператор производит довольно много действий. При некоторых допущениях процесс сопоставления можно автоматизировать.

Мы будем считать, что нам известна функция $\vec{r} = \vec{d}(F_1, F_2, w_1, w_2)$, которая определяет для двух квадратных окон w_1 и w_2 одинакового размера на изображениях $F_1(x, y), F_2(x, y)$ вектор параллельного переноса \vec{r} , с помощью которого из изображения окна w_1 получается изображение окна w_2 , если первоначально окна уже совмещены (то есть для общего смещения картинок нужно к полученному смещению добавить первоначальное смещение окон). Окном w является множество $\{(x, y) | x^w \leq x \leq x^w + N - 1, y^w \leq y \leq y^w + N - 1\}$ точек изображения, его можно задать тройкой чисел (x^w, y^w, N) . Идея быстрого вычисления функции \vec{d} описана в п. 1.3.

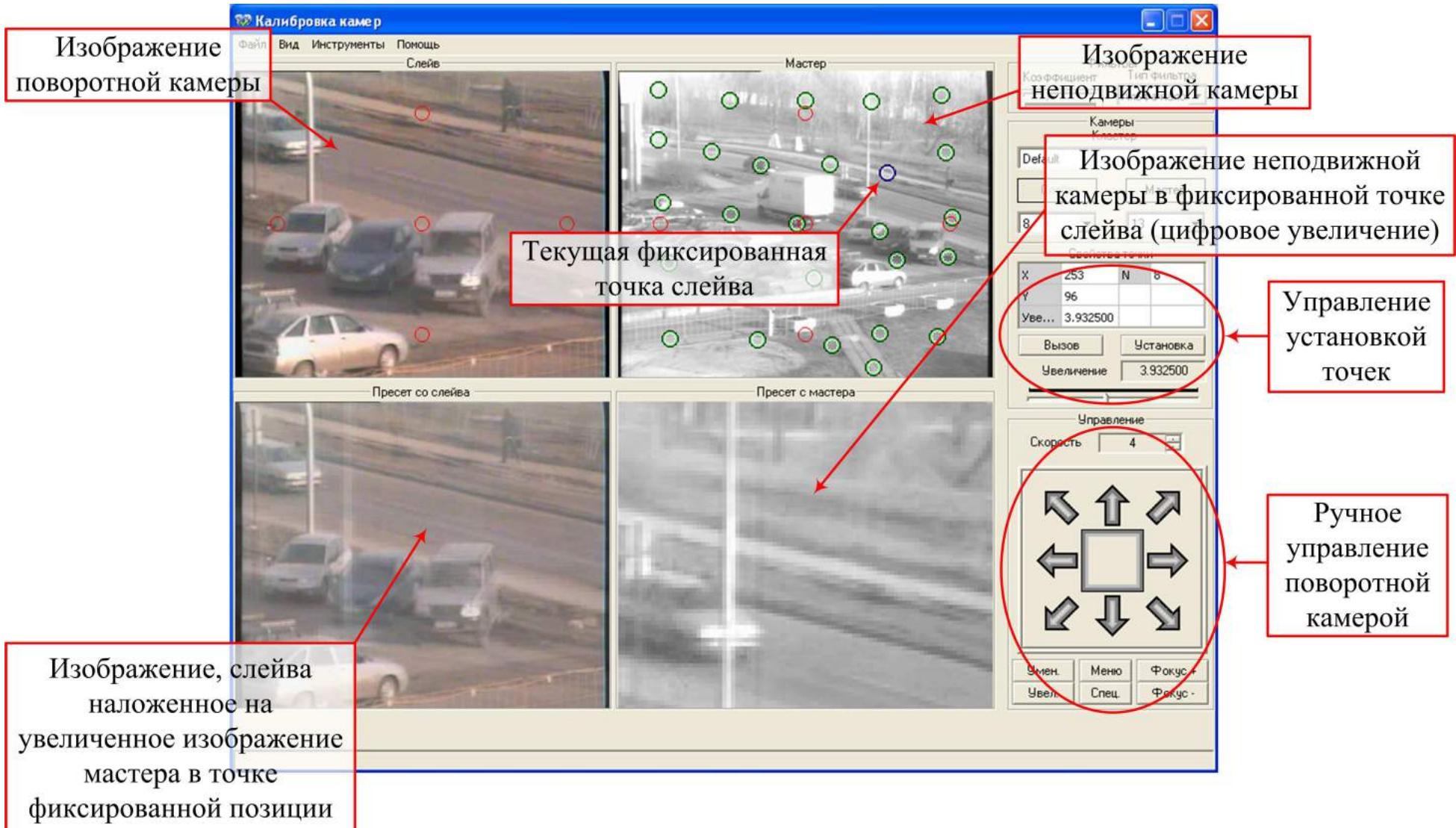


Рисунок 2.10. Интерфейс программы калибровки.

Процесс автоматического сопоставления состоит из нескольких шагов, участие оператора необходимо только на первом шаге.

1. После расстановки зон позиций из памяти поворотной видеокамеры на мастере m оператор поворачивает слейв таким образом, чтобы картинки на слейве s и мастере m совпадали, то есть слейв вручную устанавливается в положение $x = w/2$, $y = h/2$, $z = 1,0$ относительно мастера ($w \times h$ — разрешение изображения мастера в точках).

2. В автоматическом режиме производится измерение скоростей команды **ПОВЕРНУТЬ()**. Скорости измеряются в координатах изображения мастер-камеры в секунду, смещения вычисляются с помощью функции $\vec{d}(F_1, F_2, w_1, w_2)$.

3. После измерения скоростей для всех позиций P_i производится наведение в соответствующие координаты (x_i, y_i) мастер-камеры с единичным относительным увеличением. После каждого наведения положение камеры запоминается при помощи **ЗАПОМНИТЬ_ПОЗИЦИЮ(i)**. Процедура перехода от позиции к позиции использует движение по команде **ПОВЕРНУТЬ()** с измеренными скоростями, время движения рассчитывается исходя из данных, полученных на предыдущем шаге, а с помощью функции $\vec{d}(F_1, F_2, w_1, w_2)$ производится проверка действительного смещения. Более подробно алгоритм смещения на заданный вектор описан в [136], а также далее по тексту.

4. С помощью команды **УВЕЛИЧЕНИЕ()** устанавливается некоторое увеличение в каждой позиции P_i , одинаковое для всех на данном мастере. Если поворотная камера поддерживает включение/выключение автофокусировки, то устанавливается фокусировка средствами камеры. В противном случае с помощью команды **Focus()** подбирается положение фокусировки с максимальной мерой контраста (см. стр. 111). На данный момент не разработано способов установки нужного увеличения для позиции в памяти слейва из-за потери информации при цифровом увеличении мастера. Поэтому в случае, когда необходимо устанавливать разные увеличения в различных позициях, необходимо проводить ручную калибровку.

Рассмотрим более подробно некоторые шаги.

Будем называть скоростью камеры тройку чисел $\boldsymbol{\varkappa} = (d, v_\alpha, v_\beta)$, где (d, v_α, v_β) — параметры команды **ПОВЕРНУТЬ()**. Измерение определенной скорости $\boldsymbol{\varkappa}$ — это создание таблицы для функций $\vec{r}_{\boldsymbol{\varkappa}}(t_i)$ и $\sigma_{\boldsymbol{\varkappa}}(t_i)$, где t_i — сетка отсечек времени. На практике достаточно взять равномерную сетку с шагом 50мс на отрезке от 0 до 4с. Вначале измерений мы запоминаем командой **ЗАПОМНИТЬ_ПОЗИЦИЮ()** в «служебную» позицию начальное, совмещённое с мастером, положение поворотной камеры. Затем для каждого t_i производим 5 измерений смещения центра поворотной камеры, смешая её командой **ПОВЕРНУТЬ($\boldsymbol{\varkappa}$)** в течение времени t , возвращая перед каждым смещением камеру в служебную позицию с помощью команды **ВЫЗОВ_ПОЗИЦИИ()**. Среднее пяти

полученных при помощи функции $\vec{d}(F_1, F_2, w_1, w_2)$ смещений мы заносим в таблицу как $\vec{r}_\kappa(t_i)$, а средне-квадратичное отклонение — как $\sigma_\kappa(t_i)$ (F_1 — изображение мастера, F_2 — смешённое изображения слейва, w_1 и w_2 — окна в центре экрана). Измерения начинаются с отсечки времени $t_0 = 50\text{мс}$, затем время последовательно увеличивается с шагом 50мс, пока либо не достигает четырёх секунд, либо смещение превышает $N/4$ точек изображения.

Справедливы следующие утверждения:

- 1° Скорости вида $(\text{влево}, v_\alpha, v_\beta)$ и $(\text{вправо}, v_\alpha, v_\beta)$ являются одной физической скоростью для любых v_β , и, аналогично, скорости $(\text{вверх}, v_\alpha, v_\beta)$ и $(\text{вниз}, v_\alpha, v_\beta)$ являются одинаковой скоростью для любых v_α .
- 2° Тройка $(\text{вверх_влево}, v_\alpha, 0)$ эквивалентна тройке $(\text{влево}, v_\alpha, 0)$, тройка $(\text{вниз_вправо}, 0, v_\beta)$ эквивалентна тройке $(\text{вправо}, 0, v_\beta)$, так далее.
- 3° Пусть $\kappa_1 = (d, v_\alpha, 0)$, $\kappa_2 = (d, 0, v_\beta)$, $\kappa = (d, v_\alpha, v_\beta)$. Тогда

$$\vec{r}_\kappa(t_i) = \vec{r}_{\kappa_1}(t_i) + \vec{r}_{\kappa_2}(t_i), \quad (2.21)$$

$$\sigma_\kappa(t_i) = \sqrt{\frac{\sigma_{\kappa_1}^2(t_i) + \sigma_{\kappa_2}^2(t_i)}{2}}. \quad (2.22)$$

- 4° Если $\kappa_1 = (\text{вверх}, v_\alpha, v_\beta)$, $\kappa_2 = (\text{вниз}, v_\alpha, v_\beta)$, то

$$\vec{r}_{\kappa_1}(t_i) = -\vec{r}_{\kappa_2}(t_i), \quad (2.23)$$

аналогичное утверждение справедливо для $\kappa_1 = (\text{влево}, v_\alpha, v_\beta)$ и $\kappa_2 = (\text{вправо}, v_\alpha, v_\beta)$.

Утверждения 1° и 2° справедливы всегда, 3° и 4° — в предположении, что углы поворота камеры небольшие. Воспользовавшись этими свойствами, мы можем провести измерения только для скоростей $(\text{влево}, v_\alpha, 0)$ и $(\text{вверх}, 0, v_\beta)$, а таблицу $\vec{r}_\kappa(t_i)$ и $\sigma_\kappa(t_i)$ для остальных скоростей κ заполнить, используя (2.21), (2.22) и (2.23).

На рис. 2.11 показана блок-схема алгоритма составного перемещения на вектор \vec{R} . При поиске оптимальных (κ, i) используется полный перебор в таблице скоростей, коэффициент 3 при σ_κ обозначает 99-процентную вероятность оказаться в пределах δ после перемещения. При вычислении $\vec{R}_{\text{ост}} = \vec{d}(F_1, F_2, w_1, w_2)$ F_1 — кадр слейва, F_2 — кадр мастера, w_1 — окно в центре слейва, w_2 — окно в окрестности той точки мастера, в которой мы должны оказаться после перемещения на \vec{R} . Если не вычислять остаточное смещение, то в результате после перемещений будет накапливаться погрешность, см. рис. 2.12. Пользуясь процедурой перемещения, мы можем расставить позиции в памяти слейва с единичным увеличением, по очереди передвигаясь из уже установленных к ближайшей неустановленной.

Времена всех этапов автоматического сопоставления складываются так, что время, за которое выполняется процесс автоматически, сравнимо с временем ручной настройки. Кроме того, процесс не всегда работает устойчиво, и основной причиной

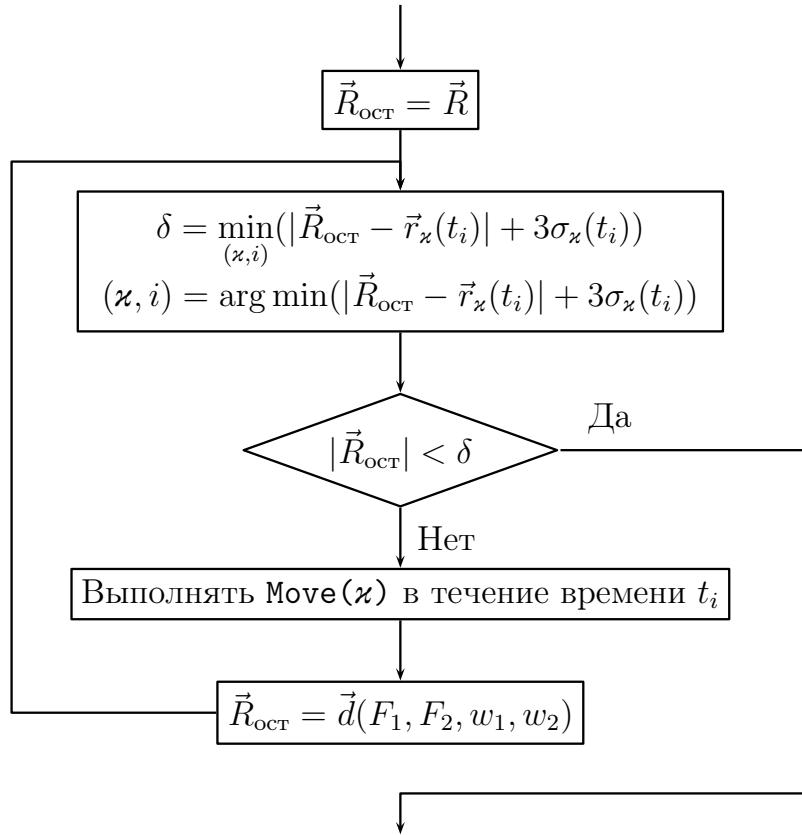


Рисунок 2.11. Блок-схема алгоритма составного перемещения на вектор \vec{R} .

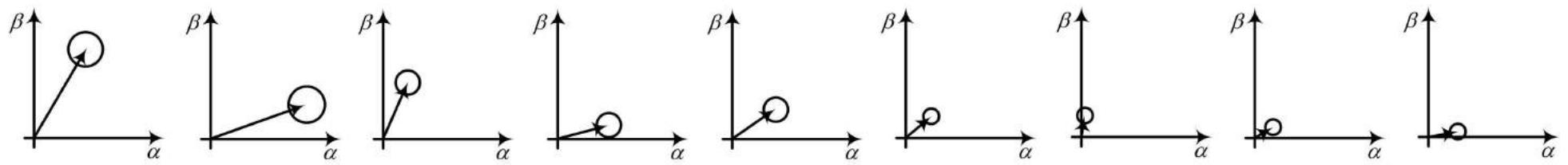
является то, что вычисление функции смещения \vec{d} невозможно просто потому, что структура изображений F_1, F_2 однородна (например, зона наблюдения — асфальтированная площадка).

В итоге, после ручной или автоматической настройки преобразование для команды `ПЕРЕЙТИ()`, которое мы настраивали, будет

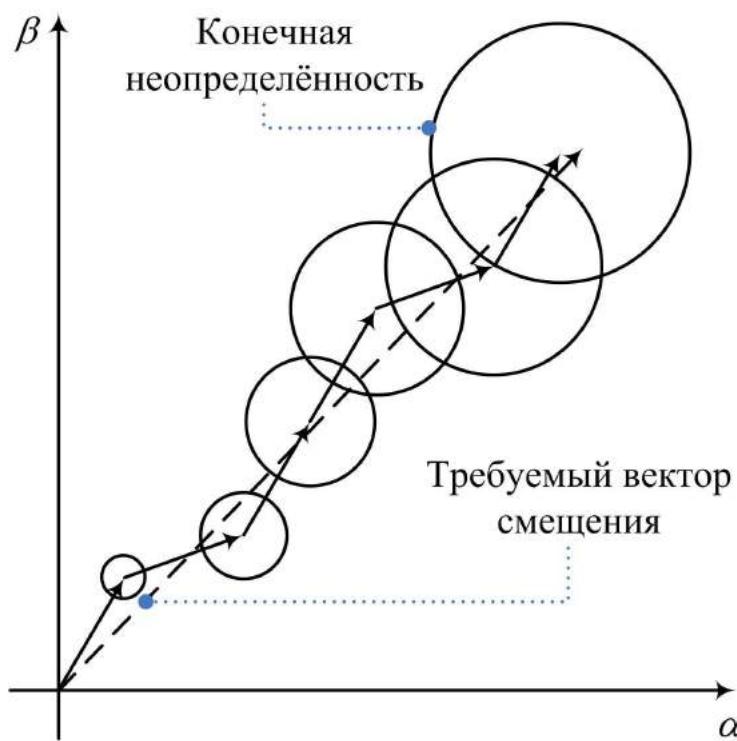
$$\mathbf{a}_m^s(x, y, z) = \arg \min_{\substack{|x - x_i| \leq w/2z_i \\ |y - y_i| \leq h/2z_i}} (W_1 \sqrt{(x - x_i)^2 + (y - y_i)^2} + W_2 |z - z_i|),$$

где $W_1 = 10$ и $W_2 = 1$ — некоторые веса. Команда `ПЕРЕЙТИ()`, таким образом, выбирает наиболее удачную позицию из памяти поворотного устройства, которая показывает окрестность точки (x, y) с наиболее близким к z увеличением.

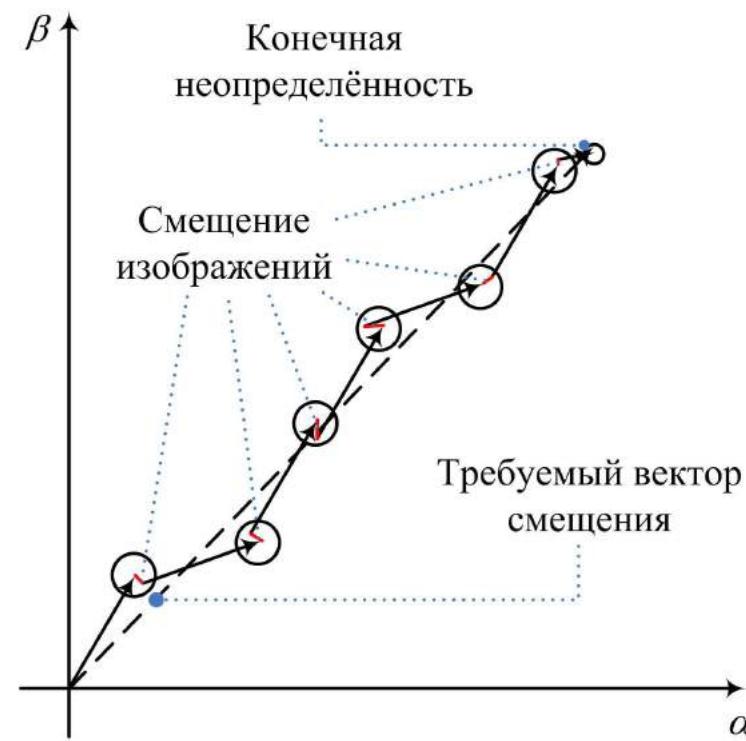
Методика настройки с использованием команд наведения по координатам. Для камер, поддерживающих команды `ВЫЗОВ_КООРДИНАТ()` и `ЧТЕНИЕ_КООРДИНАТ()`, подбор преобразования \mathbf{a}_m^s сразу решает задачу реализации команды `ПЕРЕЙТИ()`: достаточно вызвать полученные после преобразования координаты на поворотном устройстве. Задача поиска преобразования \mathbf{a}_m^s разделяется на три разные функции: $\vec{r} = A_m^s(x, y)$ (углы поворота не зависят от увеличения), $\gamma = \Gamma_m^s(z)$ (увеличение пропорционально углу обзора, поэтому нет зависимости от x и y), $f = \Phi_m^s(x, y, z)$.



а) измеренные перемещения поворотных камер (окружность обозначает неопределённость конечного положения из-за погрешности)



б) составное перемещение поворотной камеры без оценки смещения с помощью сравнения изображений



в) составное перемещение поворотной камеры с оценкой смещения с помощью сравнения изображений

Рисунок 2.12. Сравнение работы алгоритмов составного перемещения поворотной видеокамеры на заданный вектор.

Введём несколько точек $\mathcal{R}_i = (\tau_i, \gamma_i, f_i)$ ($i \in \overline{1, K}$), которые назовём *опорными* точками. Каждой опорной точке сопоставим тройку x_i, y_i, z_i . Эта операция полностью аналогична расстановке позиций из памяти поворотного устройства и описана в предыдущем пункте: сначала расставляются точки на изображении мастер-камеры, затем вручную совмещаются изображения слейва с изображением мастера. Соответствие устанавливается нажатием на кнопку «Установить» в программе калибровки. Разница заключается в том, что значения координат (τ_i, γ_i, f_i) в опорных точках доступны для чтения.

Проведём триангуляцию среди опорных точек в координатах мастера (x_i, y_i) . Это можно сделать за $O(K \log_2 K)$ действий (см. [138]), однако, поскольку точек немногого, в действительности используется простейший алгоритм поиска ближайшей подходящей точки с асимптотикой $O(K^2)$, тем более для него разработана целочисленная реализация, свободная от ошибок округления. Для каждого треугольника данной триангуляции с вершинами $(x^1, y^1), (x^2, y^2), (x^3, y^3)$ на мастере и соответствующими им координатами поворотной камеры $\vec{\tau}^1 = (\alpha^1, \beta^1), \vec{\tau}^2 = (\alpha^2, \beta^2), \vec{\tau}^3 = (\alpha^3, \beta^3)$ существует единственное аффинное преобразование A_m^s , для которого $\vec{\tau}^1 = A_m^s(x^1, y^1), \vec{\tau}^2 = A_m^s(x^2, y^2), \vec{\tau}^3 = A_m^s(x^3, y^3)$. Выбор аффинного преобразования обусловлен тем, что мы работаем в приближении небольших углов. Если формулы преобразования A_m^s записать в виде

$$\begin{aligned}\alpha &= ax + by + c, \\ \beta &= dx + ey + f,\end{aligned}\tag{2.24}$$

то коэффициенты преобразования получаются при решении системы линейных уравнений

$$\underbrace{\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}}_{\text{матрица}} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}}_{\text{вектор}}, \quad \underbrace{\begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}}_{\text{матрица}} \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \underbrace{\begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}}_{\text{вектор}}.\tag{2.25}$$

В результате получаем преобразование A_m^s для выпуклой оболочки точек (x_i, y_i) . Чтобы определить это преобразование на всей области изображения мастера, достаточно ввести четыре опорные точки по углам изображения. Если при наведении в точку (x, y) наблюдается значительное отклонение, то мы можем ввести новую опорную точку с координатами (x, y) . Точность наведения повысится за счёт увеличения мелкости разбиения треугольниками (определение понятия мелкости разбиения многомерного множества см., например, в [137], с. 130). Именно этим обусловлено применение триангуляции и линейной интерполяции при подборе преобразования A_m^s .

Функции увеличения Γ_m^s и фокусировки Φ_m^s зависят от устройства камеры [139]. Например, существует модель камеры, у которой координаты увеличения пропорци-

ональны кратности увеличения, то есть

$$\Gamma_m^s(z) = k_m^s z, \quad (2.26)$$

где k_m^s — постоянная для данной пары мастер-слейв. Фокусировка же для этой камеры подчиняется модели

$$\Phi_m^s(z) = \begin{cases} C, & \gamma > B \\ A\gamma, & \gamma \leq B \end{cases}, \quad (2.27)$$

где A, B, C — некоторые коэффициенты, все лежат в пределах от 0 до 300, $\gamma = k_m^s z$ — значение координаты увеличения для z . Оказалось, что для всех практических применений у данной камеры хватает глубины резкости для того, чтобы не учитывать зависимость фокусировки от координат (x, y) , то есть от расстояния до точки.

Постоянная k_m^s вычисляется при помощи метода наименьших квадратов из увеличений z_i и соответствующих координат γ_i в введённых опорных точках:

$$k_m^s = \frac{\bar{\gamma}z}{\bar{z}^2}. \quad (2.28)$$

Для определения A, B, C требуется сначала проделать серию измерений фокусировки хотя бы в одной точке (x, y) мастера. Для этого нам понадобится мера фокусировки изображения $\phi(F)$ на кадре $F(x, y) = (F^R(x, y), F^G(x, y), F^B(x, y))$ (см. формулу (1.1)) с разрешением $w \times h$. Известно, что мерой фокусировки служит мера контрастности, один из способов оценить контрастность — это посчитать величину

$$\phi(F) = \frac{4}{w} \cdot \frac{4}{h} \sum_{x=0, y=0}^{w/4, h/4} \frac{i_{max}(x, y) - i_{min}(x, y)}{i_{max}(x, y) + i_{min}(x, y) + 1}, \quad (2.29)$$

$$i_{min}(x, y) = \min \frac{1}{3} \sum_C F^C(x', y') \quad , \quad i_{max}(x, y) = \max \frac{1}{3} \sum_C F^C(x', y') \quad ,$$

$$\begin{array}{ll} 4x \leq x' < 4(x+1) & 4x \leq x' < 4(x+1) \\ 4y \leq y' < 4(y+1) & 4y \leq y' < 4(y+1) \end{array}$$

то есть $\phi(F)$ — среднее арифметическое контрастностей в окошках изображения размером 4×4 . Усреднение введено, чтобы исключить влияние шума, из-за которого на изображении практически обязательно присутствуют черные и белые точки.

Диапазон координат γ от нуля до 255 мы разобьём на 6 отсечек γ_j (число отсечек подобрано экспериментально), и в каждой отсечке увеличения переберем координаты фокуса в диапазоне от 0 до 300 с шагом 20 (таково минимальное изменение при управлении командой **ФОКУС()** на минимальной скорости), выбирая оптимальную координату фокусировки f_j по мере (2.29). Далее мы перебором значений A, B, C с шагом 20 в кубе $[0, 300]^3$ решим задачу

$$\min_{A, B, C} \sum_{j=1}^6 (f_j - \Phi_m^s(\gamma_j/k_m^s))^2,$$

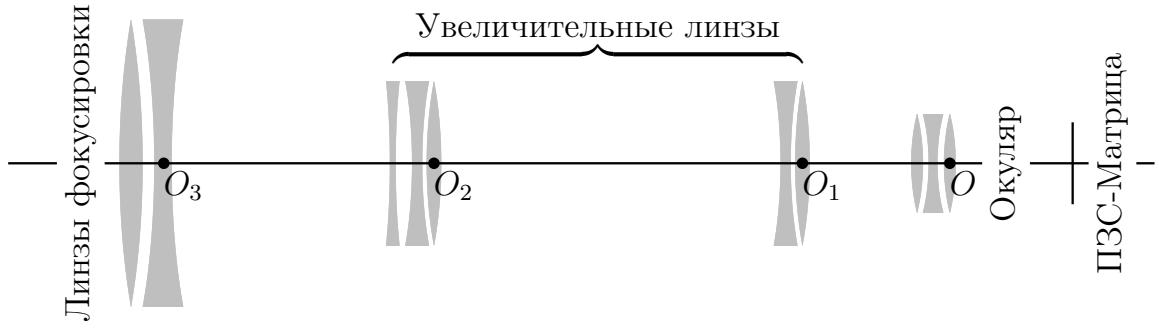


Рисунок 2.13. Оптическая схема объектива.

где Φ_m^s рассчитывается по формуле (2.27). Вся процедура занимает около 6 минут для каждой пары камер мастер-слейв, и не требует участия оператора после расстановки нескольких опорных точек и точки измерения фокуса.

В общем случае для определения функций $\gamma = \Gamma_m^s(z)$ и $f = \Phi_m^s(x, y, z)$ требуется проделать более детальные измерения. Большинство производителей объективов не указывает в спецификации зависимости между координатой увеличения и кратностью увеличения; такую зависимость необходимо устанавливать экспериментально. *Кратность увеличения* — отношение максимального угла зрения для данного объектива к текущему углу зрения. Устройство современных объективов достаточно сложное и включает в себя около десятка линз. Примерная оптическая схема объектива представлена на рис. 2.13.

Для большинства объективов окуляр и ПЗС-матрица расположены неподвижно на оптической оси, передняя группа линз регулирует фокусировку, а две системы линз внутри объектива, эквивалентные двум тонким линзам, регулируют увеличение. Изменение координаты γ влечет изменение положения увеличительных линз относительно окуляра; при изменении координаты f меняется расстояние OO_3 . Для объективов, использовавшихся в ходе диссертационной работы, экспериментально выявлены следующие соотношения:

$$\begin{aligned} OO_1 &= \chi(\gamma), \\ OO_2 &\sim \gamma + \text{const}, \\ OO_3 &\sim f + \text{const}, \end{aligned} \tag{2.30}$$

где $\chi(\gamma)$ — некоторая нелинейная функция. При этом установлено, что функция кратности увеличения объектива $\tilde{\Gamma}(\gamma)$ и функция фокусировки $\Phi_m^s(x, y, z)$ являются дробно-линейными относительно γ :

$$\tilde{\Gamma}(\gamma) = \frac{A_\gamma \gamma + B_\gamma}{C_\gamma \gamma + D_\gamma}, \tag{2.31}$$

$$\Phi_m^s = \frac{A_f \gamma + B_f}{C_f \gamma + D_f}, \tag{2.32}$$

где коэффициенты $A_\gamma, B_\gamma, C_\gamma, D_\gamma$ постоянны для отдельно взятого объектива, а ко-

эффициенты A_f, B_f, C_f, D_f зависят от расстояния до объекта, то есть от координат (x, y) . Это означает, что функцию (2.31) достаточно измерить один раз для данного объектива, а затем использовать её при каждой настройке, в отличие от функции (2.32), которую необходимо измерять на некоторой сетке (x, y) координат мастер-камеры, с которой проводится сопряжение.

Алгоритм измерения функции $\tilde{\Gamma}$ устроен следующим образом. Пусть $\gamma \in [\gamma_{min}, \gamma_{max}]$. Введём на этом промежутке равномерную сетку $\{\gamma_i\}_{i=0}^n$, $n \leq 6$, $\gamma_i = \gamma_{min} + i \cdot (\gamma_{max} - \gamma_{min})/n$. Наведём объектив на такой объект, что при любом увеличении мы можем получить чёткое изображение. Сначала установим на объективе координату γ_0 , вручную или автоматически подберём фокусировку и запомним изображение с камеры как эталонное с увеличением кратности 1,0. Для остальных точек сетки γ_i после установки фокусировки мы определяем кратность увеличения, используя совмещение картинки камеры с цифровым увеличением эталонного изображения. Коэффициент цифрового увеличения подбирается вручную. Получив таким образом таблицу значений $\tilde{\Gamma}_i$ в точках γ_i , мы можем найти коэффициенты $A_\gamma, B_\gamma, C_\gamma, D_\gamma$ методом наименьших квадратов:

$$A_\gamma, B_\gamma, C_\gamma, D_\gamma = \arg \min \sum_{i=0}^n \left(\tilde{\Gamma}_i - \frac{A_\gamma \gamma_i + B_\gamma}{C_\gamma \gamma_i + D_\gamma} \right)^2, \quad (2.33)$$

эта задача решается, например, методом градиентного спуска.

Измерив функцию $\tilde{\Gamma}(\gamma)$, легко определить функцию $\Gamma_m^s(z)$, поскольку из соотношений углов зрения следует, что

$$\tilde{\Gamma}(\Gamma_m^s(z)) = k_m^s z \Rightarrow \Gamma_m^s(z) = \tilde{\Gamma}^{-1}(k_m^s z), \quad (2.34)$$

то есть определение функции $\Gamma_m^s(z)$ сводится к нахождению коэффициента k_m^s . Расчёт коэффициента можно провести по формуле (2.28), подставив в неё данные введённых опорных точек, предварительно произведя «замену координат» $\gamma \rightarrow \tilde{\Gamma}(\gamma)$.

Метод измерения функции фокусировки для модели (2.32) практически полностью повторяет метод измерения функции (2.27). Отличия заключаются в том, что в силу малой глубины резкости большинства объективов необходимо провести измерения в сетке точек $\{(x_k, y_k)\}_{k=0}^m$, находящихся на разных расстояниях от объектива, и в том, что задача минимизации для каждой точки сетки

$$\min_{A_f^k, B_f^k, C_f^k, D_f^k} \sum_{j=0}^m \left(f_j^k - \frac{A_f^k \gamma + B_f^k}{C_f^k \gamma + D_f^k} \right)^2$$

решается методом градиентного спуска, а не полным перебором. В итоге подсчёт функции фокусировки осуществляется по формуле

$$\Phi_m^s(x, y, z) = \frac{A_f^k \Gamma_m^s(z) + B_f^k}{C_f^k \Gamma_m^s(z) + D_f^k}, \quad (2.35)$$

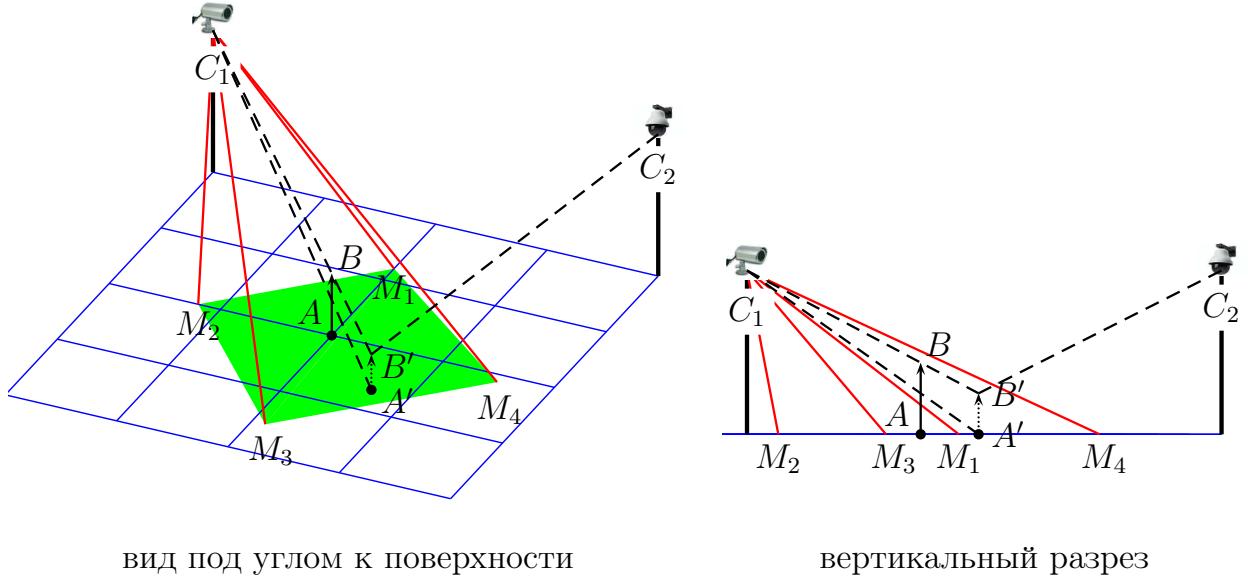


Рисунок 2.14. Мастер и слейв, удалённые друг от друга.

где $k = \arg \min ((x - x_k)^2 + (y - y_k)^2)$ — индекс ближайшей к (x, y) точки из сетки измерения фокусировки.

После ручного ввода данных для определения функции $\tilde{\Gamma}(\gamma)$ и ввода опорных точек \mathcal{R}_i дальнейшая настройка объектива происходит автоматически. Из-за ограничений, накладываемых глубиной резкости объектива, процедура определения функции фокусировки, задаваемой моделью (2.32), работает значительно дольше, чем для модели (2.27). Время, необходимое для сопряжения, может достигать 40 минут для одной пары камер мастер-слейв.

Особенности методики сопряжения для камер, закреплённых в удалённых точках. Пусть наблюдаемый объект находится в точке A , слейв-камера установлена в точке C_2 и расположена на значительном расстоянии от мастера, установленного в точке C_1 , т.е. расстояние $|C_1C_2|$ сравнимо с расстоянием $|C_1A|$ от точки мастера до объекта (см. рис. 2.14). Тогда методы настройки функции $\mathbf{a}_m^s(x, y, z)$, в целом, не изменяются по сравнению с методами, описанными в двух предыдущих пунктах. Однако необходимо иметь в виду следующие особенности:

1. Установка позиций и опорных точек возможна только вручную, поскольку метод сравнения изображений в данном случае неприменим. Даже при ручной установке необходимы дополнительные ориентиры на изображении сопрягаемых видеокамер. Иногда оператору требуется помощник, который перемещается по зоне наблюдения и выполняет функции «ориентира».
2. В случае, если C_2 лежит на большом расстоянии от оптической оси мастера (см. рис. 2.14) или расстояние от мастера до объекта больше расстояния $|C_1C_2|$ или сравнимо с ним, линейность преобразования $\vec{r} = A_m^s(x, y)$ соблюдается лишь в некоторой небольшой окрестности точки (x, y) . Поэтому в данном случае требуется ввести

большое количество опорных точек.

3. Модели для увеличения (2.26), (2.34) и для фокусировки (2.27), (2.32) не работают, причем в данном случае функция $\gamma = \Gamma_m^s(x, y, z)$ зависит от точки (x, y) , в которой объект находится на мастере. Чтобы учесть это, приходится вводить сетку на координатах (x, y) мастере, и в каждом узле вручную проводить измерения функции увеличения по приведённому выше алгоритму. Сетка требуется достаточно плотная, обычно используется около 16 точек равномерной двумерной сетки.

4. Вероятность ошибочного наведения значительно возрастает при неправильном определении рамки объекта на мастере.

Поясним последний пункт. На рис. 2.14 изображена ситуация, при которой только верхняя половина объекта AB определена обработчиком мастер-камеры, расположенной в точке C_1 . Чтобы навести слейв-камеру, установленную в точке C_2 , на объект, система сначала продолжит луч C_1A' до пересечения с поверхностью $M_1M_2M_3M_4$ зоны наблюдения мастера и получит положение объекта в физическом пространстве $A'B'$, значительно отличающееся от действительного (иногда погрешность $|AA'|$ бывает порядка 10 метров). После наведения слейва по лучу C_2B' объект, скорее всего, окажется вне кадра. Погрешность выделения рамки не повлияла бы на точность наведения в случае, если бы слейв находился бы вблизи оптической оси мастера, и выполнялось бы ограничение $|C_1C_2| \ll |C_2A|$. В этом случае слейв-камера может быть наведена по лучу C_1B , который начиная с некоторого момента попадет в «пирамидный» конус лучей проекции слейв-камеры, поэтому объект обязательно окажется в кадре, так как точка $B' \in C_1B$.

Итак, использование удалённых пар камер мастер-слейв приводит к трудоёмкой настройке и повышению погрешности наведения. Поэтому чаще всего поворотные камеры ставят либо в одной точке с мастером, либо вблизи оптической оси сопряжённого мастера (при этом необходимо, чтобы соблюдалось условие значительного превышения расстояния от мастера до объекта над расстоянием между слейвом и мастером).

Выводы

1. Определены основные понятия и термины, связанные с анализом систем технического зрения с применением семантического сжатия. Разработана модель взаимодействия распределённых модулей семантической обработки видеосигналов для сопряжения видеокамер и для синхронизации данных на основе предложенного автором нового сетевого протокола информационного обмена.

2. Определены ограничения на количество видеоприборов, которые могут быть сопряжены при помощи синхронизации данных. На основании теории сложности алгоритмов показано, что предельное количество видеокамер в кластере, обусловленное

производительностью ЭВМ, оказалось значительно меньше, чем аналогичное ограничение, обусловленное пропускной способностью ЛВС. Допустимое количество мастеров M и слейвов N , участвующих в сопряжении, определяется ограничениями $M = 4$ при $N = 4$, $3 \leq M \leq 9$ при $N = 3$, $2 \leq M \leq 14$ при $N = 2$ и $M \leq 28$ при $N = 1$, при $N > 4$ диспетчер не сможет обрабатывать видеокамеры на ЭВМ с эталонной производительностью.

3. На основе полученных соотношений для сопряжения изображений двух неподвижных видеокамер разработан способ подбора преобразования с помощью методов оптимизации. Проанализированы ограничения метода: метод применяется при условии установки сопрягаемых камер в точках, находящихся близко друг от друга.
4. Исследованы особенности преобразования координат с изображения видеокамеры на изображение карты. Использован метод приближения частичными суммами ряда Фурье, определены простые способы подбора кусочных преобразований.
5. Разработан способ привязки изображения к плану местности при помощи инструментов измерения расстояний, получено уравнение линии горизонта на изображении камеры.
6. Для случая сопряжения приборов технического зрения с помощью четырёх опорных точек получена оценка погрешности сопряжения в виде линейных функционалов погрешностей входных данных (см. формулу (2.20)).
7. Разработаны алгоритмы сопряжения поворотных видеокамер с приборами технического зрения для различных типов управления поворотным устройством. На основе математических моделей устройств управления разработаны алгоритмы, частично автоматизирующие настройку сопряжения.
8. Предложен алгоритм составного перемещения поворотной видеокамеры, не увеличивающий погрешность перемещения за счёт сравнения изображений. Полностью автоматизирован выбор параметров управления фокусировкой и увеличением объективов поворотных устройств на основе методов оптимизации и интерполяции.

3. Исследование и разработка алгоритмов синхронизации, управления и семантической обработки информации в системах технического зрения

3.1. Разработка алгоритмов синхронизации программных модулей обработки цифровых информационных массивов

В современных аналитических видеосистемах, как правило, применяются комплексы видеокамер, как стационарных, так и установленных на поворотных платформах. Используя методы семантического сжатия видеосигналов, программные модули могут выделить значимые события в поле зрения видеокамер и существенно сократить объём информации, представляемой оператору видеосистемы через терминальные модули.

Сопряжение видеокамер и синхронная обработка позволяет существенно повысить детализацию информационных сообщений о событиях в поле зрения видеокамер в тех случаях, когда за одними и теми же событиями производится наблюдение с помощью нескольких различных видеокамер. Например, с помощью команд поворота и управления увеличением видеокамер на поворотных платформах можно вместе с видеозаписью событий представить детальное оптически увеличенное изображение участников инцидентов. Однако для этого необходимо разработать алгоритмы синхронизации модулей обработки видеосигналов в составе одной видеосистемы семантического сжатия.

Как следует из рис. 2.6 (с. 85), основную роль в синхронизации данных выполняет диспетчер кластера видеокамер (см. также [128]). Именно в диспетчере должны быть реализованы основные алгоритмы синхронизации программных модулей.

Введём обозначения, аналогичные обозначениям в п. 2.2:

- 1) M — количество мастер-камер в кластере диспетчера;
- 2) N — количество слейв-камер в кластере диспетчера;
- 3) n — среднее количество объектов на одной обзорной камере;
- 4) f — частота обработки кадров с обзорных камер;
- 5) T_R — среднее время исполнения поворота слейвами.

Диспетчер примерно f раз в единицу времени получает данные об объектах от каждого из M мастеров, и $1/T_R$ раз в единицу времени сообщения о готовности выполнять задания от N слейвов. Диспетчер работает в режиме ожидания сообщений, после каждого сообщения он может выполнять действия для распределения заданий поворотным камерам, чтобы они наводились на движущиеся объекты. Тем самым выполняется синхронизация положения объектов между слейв-камерой и мастер-камерой,

номер слейв-камеры и номер мастер-камеры, а также номер объекта выбирает диспетчер.

Существует несколько возможных алгоритмических схем работы диспетчера, в каждой из схем тем или иным образом группируются пять элементарных блоков:

- 1) копирование данных об n объектах из сообщения от мастера во внутренние переменные-массивы (разные для разных мастеров);
- 2) сортировка по приоритету объединения списка всех объектов, поступивших диспетчеру (максимум Mn объектов, по n от каждой из M мастер камер) с целью выделить наиболее вероятные объекты-кандидаты на фотографирование и поместить их в начало общего списка (на данный момент приоритет определяется временем последнего снимка объекта, чем оно больше, тем приоритетнее объект для съёмки, только что поступивший объект считается сфотографированным «бесконечно давно»);
- 3) проверка для n переданных от мастер-камеры и скопированных в блоке 1 объектов, может ли быть наведена на них какая-либо из N слейв-камер, т.е. можно ли выполнить команду ПЕРЕЙТИ() для текущего положения объекта;
- 4) добавление n объектов из сообщения от мастера или из соответствующего массива-копии в отсортированный список кандидатов, содержащий Mn элементов;
- 5) распределение Mn объектов из отсортированного списка по N поворотным камерам.

Введём для блоков 1 – 5 времена выполнения t_i , $i \in \overline{1, 5}$. Для этих времён справедливы следующие соотношения (в наихудшем случае):

$$t_1 = C_1 n \quad (3.1)$$

$$t_2 = C_2 Mn \log_2(Mn) \quad (3.2)$$

$$t_3 = C_3 Nn \quad (3.3)$$

$$t_4 = C_4 n \log_2(Mn) \quad (3.4)$$

$$t_5 = C_5 N! Mn \quad (3.5)$$

Блоки 1 – 4 стандартны, подробное описание алгоритмов для них можно найти, например, в источнике [140]. В этом источнике также приведены оценки времени исполнения (3.1) – (3.4) в форме $O(\dots)$. Пояснений требует лишь формула (3.5). Она следует из устройства алгоритма пятого блока: для каждой перестановки из свободных слейв-камер происходит просмотр отсортированного списка объектов в порядке убывания приоритета. Переход к следующему объекту в списке происходит в любом случае, а переход к следующему слейву в рассматриваемой перестановке — только в случае, если слейв может сфотографировать текущий объект (попутно для слейва помечается, что ему может быть назначено задание на фотографирование данного объ-

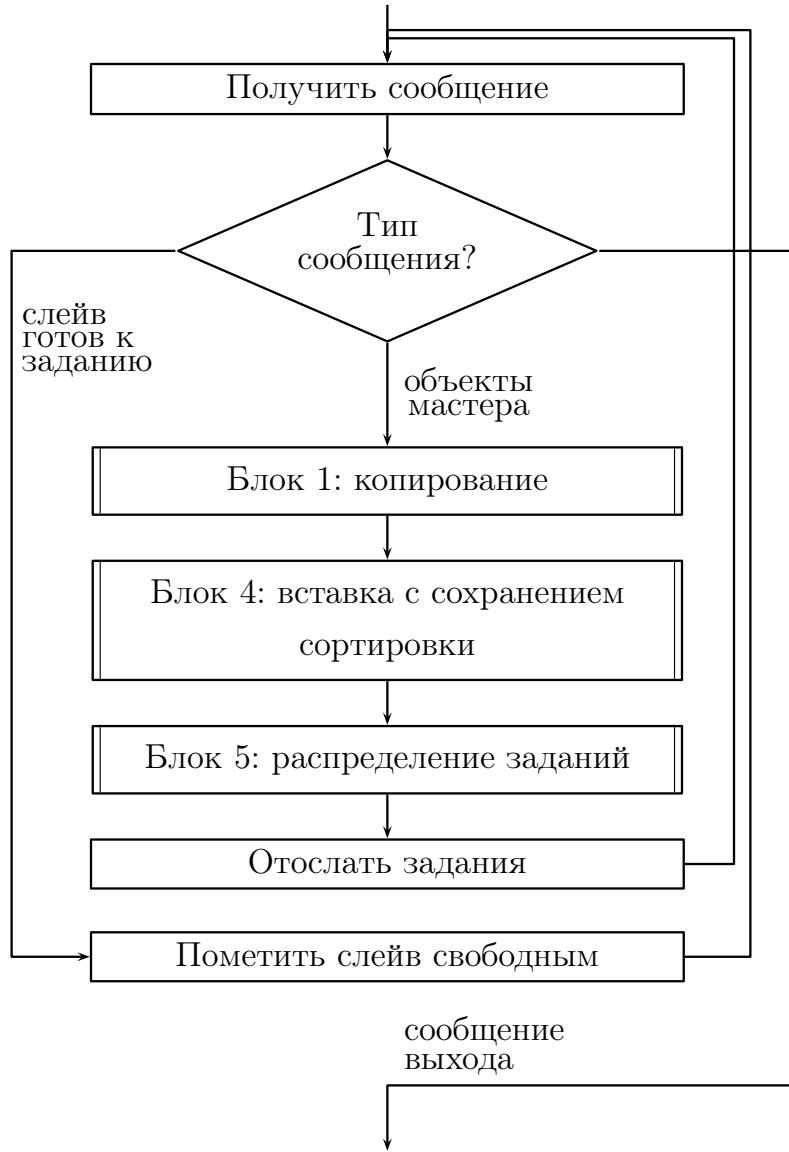


Рисунок 3.1. Блок-схема алгоритма диспетчера, функционирующего в цикле приёма сообщений от обработчиков неподвижных камер.

екта). Из всех перестановок слейлов выбирается наилучшая, то есть та, при которой сфотографировано наибольшее число объектов, которые находятся в верхней части списка кандидатов. Поскольку в наихудшем случае свободно N слейлов, то необходимо просмотреть $N!$ перестановок, для каждой перестановки происходит в наихудшем случае один проход по списку из Mn объектов. Отсюда получаем формулу (3.5).

Рассмотрим три схемы работы диспетчера, изображённые на рисунках 3.1–3.5.

1. При каждом получении списка объектов от мастера диспетчер сохраняет объекты (блок 1), добавляет их в отсортированный список (блок 4), и распределяет объекты из отсортированного списка по свободным слейвам (блок 5). Все три операции будут происходить с частотой fM . При получении сообщения от слейва диспетчер лишь исправляет статус занятости слейва, что практически не занимает процессорного времени. Таким образом, данный алгоритм большую часть времени работает в

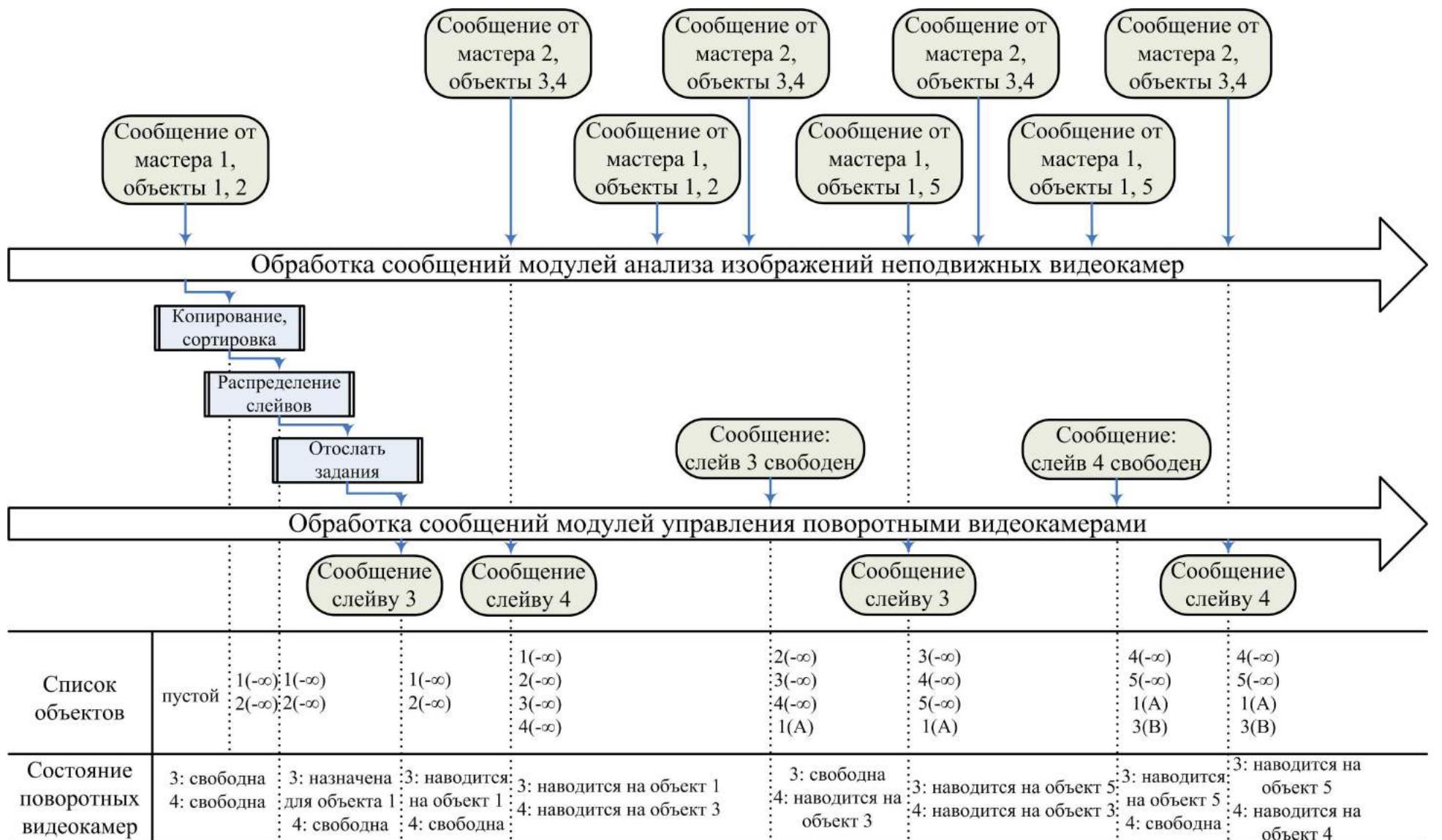


Рисунок 3.2. Иллюстрация работы диспетчера в цикле приёма сообщений от обработчиков неподвижных камер.

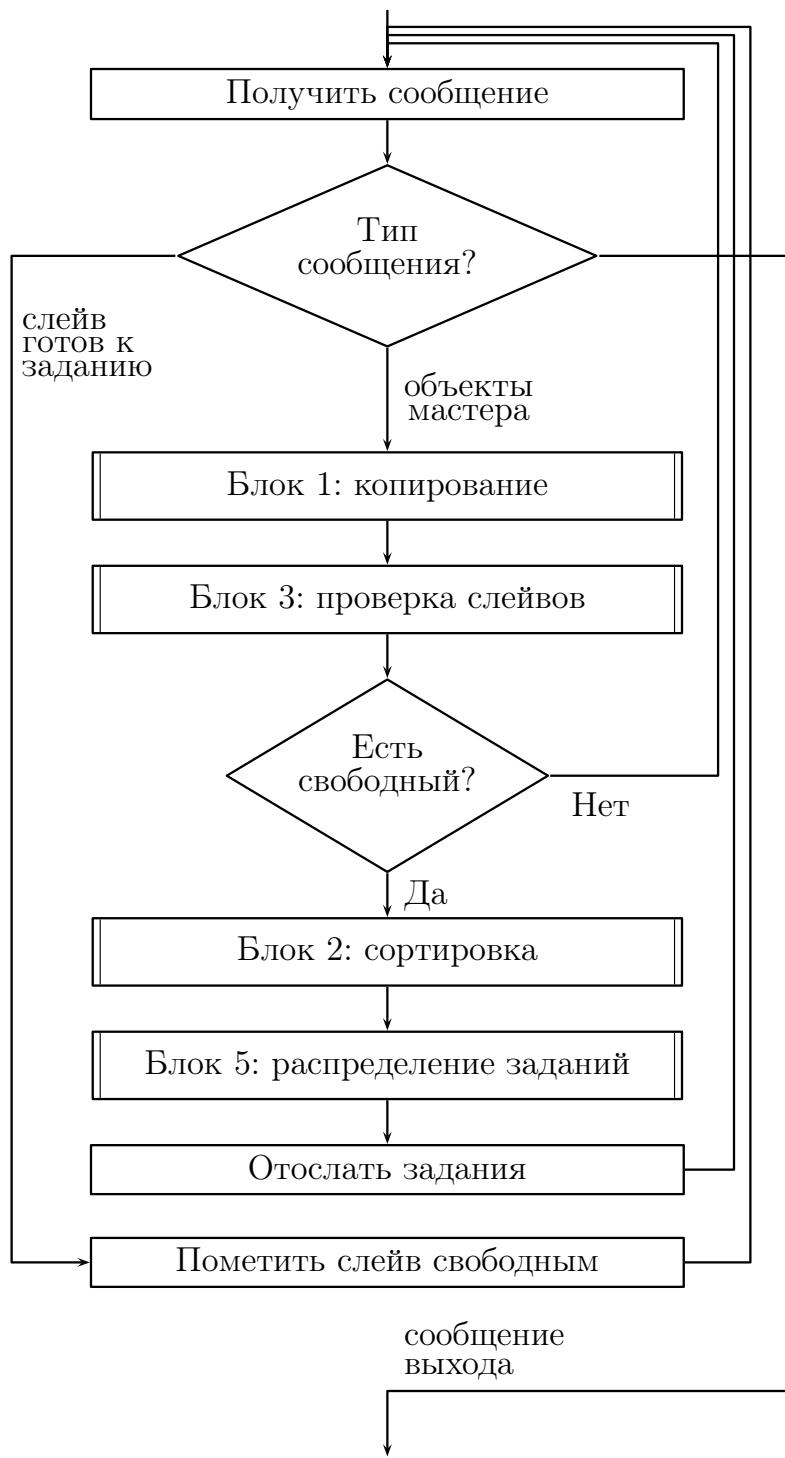


Рисунок 3.3. Блок-схема алгоритма диспетчера, функционирующего в циклах приёма сообщений от поворотных и неподвижных камер.

цикле приёма сообщений от мастеров. На рис. 3.1 изображена блок-схема алгоритма, а рис. 3.2 иллюстрирует пример обработки сообщений.

2. При каждом получении списка объектов от мастера (частота fM) диспетчер сохраняет объекты (блок 1), а затем проверяет, есть ли слейвы, которые могут данные объекты сфотографировать (блок 3). Только в случае, если последняя проверка даёт положительный результат, запускаются процедуры сортировки списка объектов

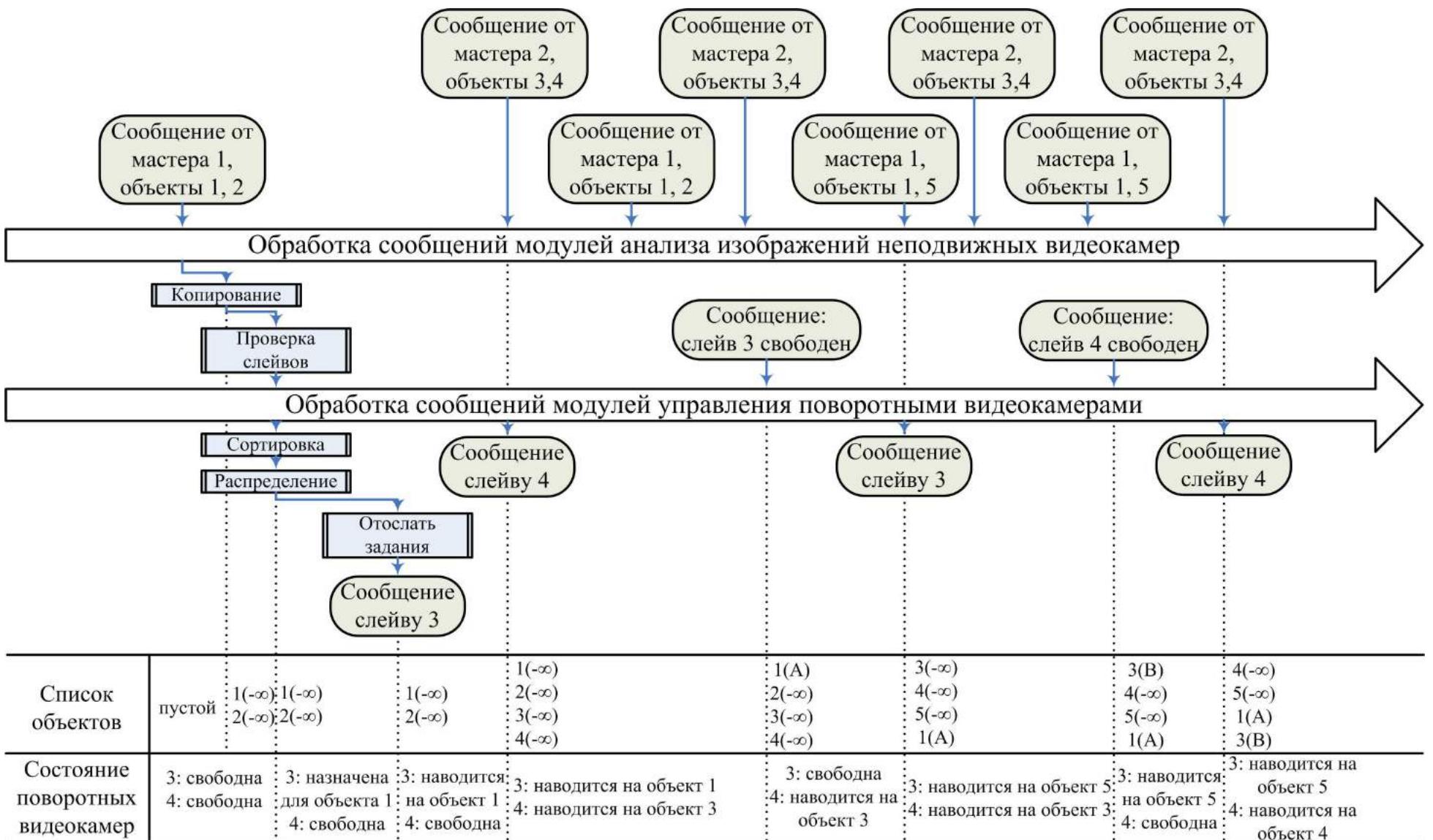


Рисунок 3.4. Иллюстрация работы диспетчера в цикле приёма сообщений от обработчиков неподвижных и поворотных камер.

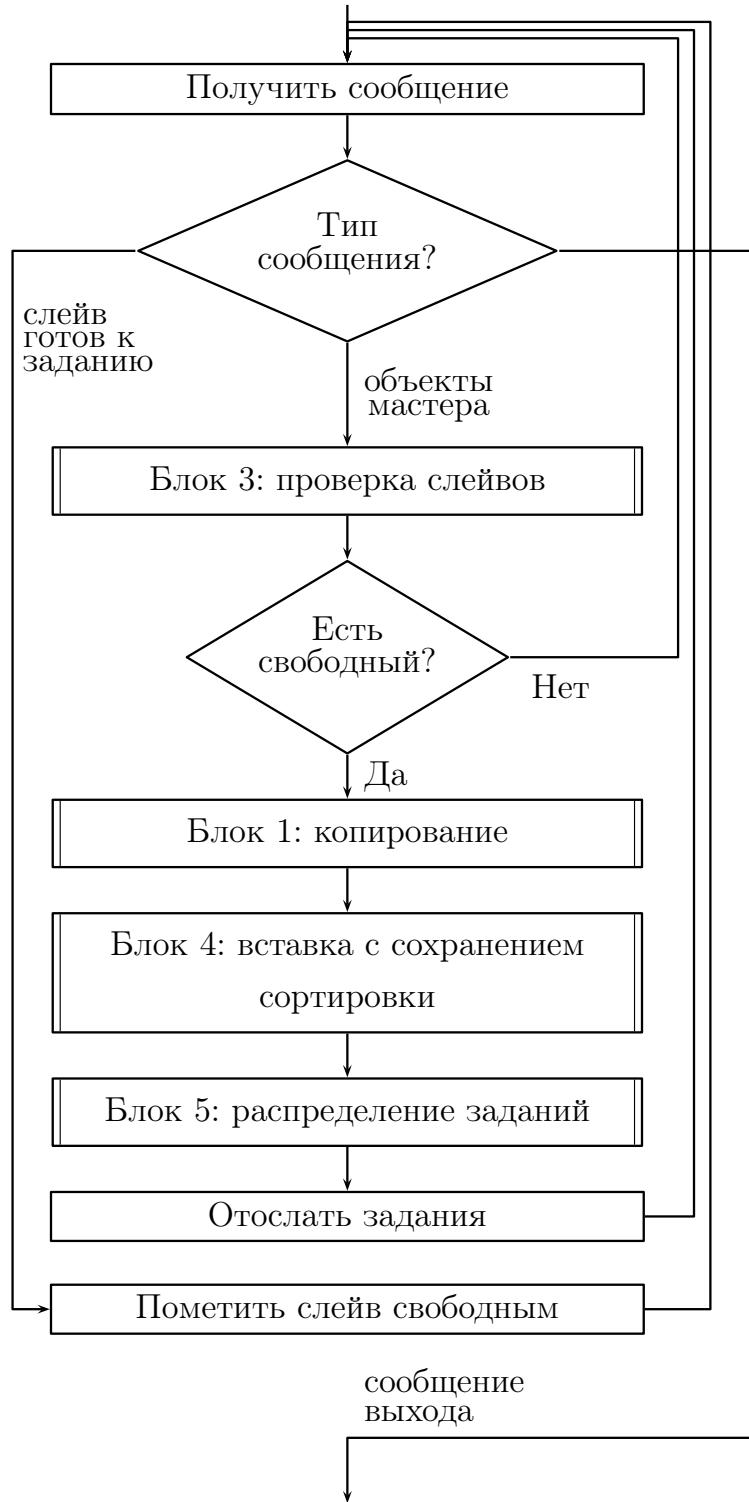


Рисунок 3.5. Блок-схема алгоритма диспетчера, функционирующего в цикле приёма сообщений от поворотных камер.

(блок 2) и распределения заданий (блок 5). Последнее будет происходить со средней частотой N/T_R . Сообщение от слейва исправляет статус занятости слейва в диспетчере. Работа алгоритма проводится как в цикле приёма сообщений от мастеров, так и в цикле приёма сообщений от слейвов, см. блок-схему на рис. 3.3 и пример на рис. 3.4.

3. При получении объектов от мастера (частота fM) диспетчер производит про-

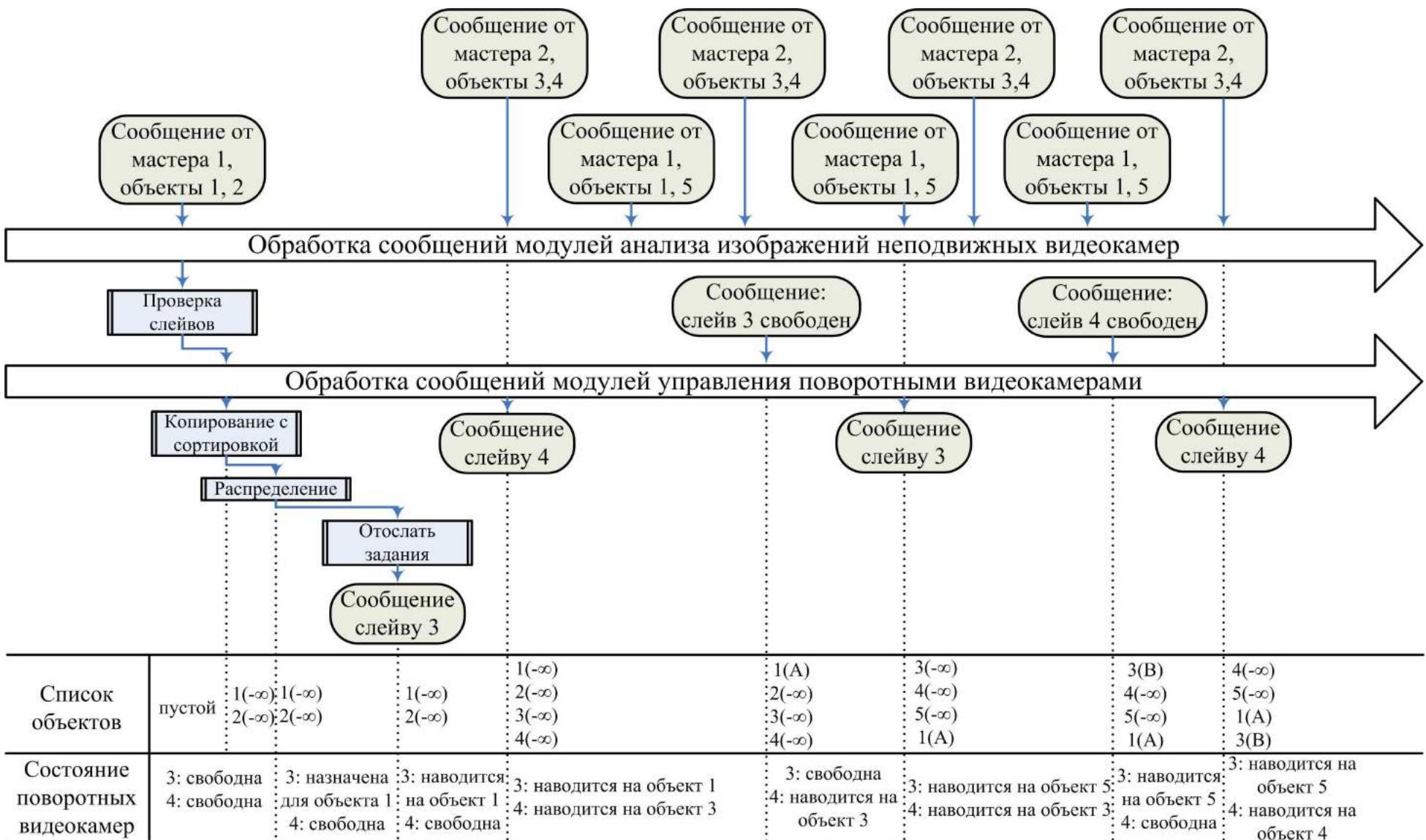


Рисунок 3.6. Иллюстрация работы диспетчера в цикле приёма сообщений от обработчиков поворотных камер.

верку на наличие свободных слейвов, которые могут сфотографировать данные объекты, и при положительном результате (частота N/T_R) сохраняет объекты (блок 1), добавляет их в отсортированный список (блок 4) и раздаёт задания слейвам (блок 5). Сообщение от слейва исправляет статус занятости. См. также рис. 3.5.

Каждая из схем обладает своими достоинствами и недостатками. Например, третья схема обладает задержкой обновления информации об объектах в диспетчере, поскольку период обновления данных для этого алгоритма равен в среднем T_R/N , что обычно значительно меньше, чем $1/fM$ для двух других вариантов работы. Это может привести к потери точности наведения поворотных камер из-за задержки синхронизации данных о положении объектов при сбоях передачи данных по ЛВС.

Из всех характеристик главным критерием качества алгоритма синхронизации является скорость его работы. Чем быстрее выполняются циклы синхронизации между ожиданием сообщений, тем для большего количества обработчиков видеокамер в кластере диспетчер может синхронизировать данные о положении объектов. Рассмотрим вычислительную сложность каждой схемы.

3.2. Теоретическая оценка быстродействия алгоритмов синхронизации

Для того, чтобы оценить вычислительную сложность для каждой из трёх алгоритмических схем синхронизации, мы будем пользоваться формулами (3.1) – (3.5) для времён t_1, t_2, t_3, t_4, t_5 , обозначающих время выполнения элементарных блоков, введённых в предыдущем пункте.

Рассмотрим свойства констант C_1, C_2, C_3, C_4, C_5 , задействованных в данных формулах. Заметим, что $C_3 = C_5$, поскольку речь идёт об одной и той же элементарной операции проверки возможности выполнения команды ПЕРЕЙТИ(). По той же причине $C_2 \approx C_4$, в данном случае элементарной операцией является операция сравнения двух времён при сортировке или поиске. Кроме того, при помощи счетчика тактов эталонного процессора были экспериментально измерены константы формул (3.1) – (3.5):

$$\begin{aligned} C_1 &= 0,1\text{мкс}, \\ C_2 &= C_4 = 2\text{мкс}, \\ C_3 &= C_5 = 1\text{мкс}. \end{aligned} \tag{3.6}$$

Относительная погрешность измерения констант составляет 20%.

Для каждой схемы можно определить доли $\delta_1, \delta_2, \delta_3$ процессорного времени, в течение которого будет работать диспетчер на ЭВМ с эталонной вычислительной

```

double n=50,f=25,TR=0.5;
double C1=0.1e-6,C2=2e-6,C5=1e-6;
double t1,t2,t3,t4,t5;
double T1,T2;
double Nfact;
for (long M=1;M<100;M++){
    Nfact=1;
    for (long N=1;N<=M;N++) {
        Nfact=Nfact*N;
        t1=C1*n;
        t2=C2*M*n*log(M*n)/log(2.0);
        t3=C5*N*n;
        t4=t2/M;//=C2*n*log(M*n)/log(2.0);
        t5=C5*Nfact*M*n;
        T1=f*M*(t1+t4+t5);
        T2=f*M*(t1+t3)+(t2+t5)*N/TR;
        if (T1<T2)
            printf("Не выполняется для: M=%i, N=%i\n",M,N);
    }
}

```

Рисунок 3.7. Программа, проверяющая неравенство $\delta_1 > \delta_2$.

МОЩНОСТЬЮ:

$$\delta_1 = fM(t_1 + t_4 + t_5), \quad (3.7)$$

$$\delta_2 = fM(t_1 + t_3) + \frac{1}{T_R}N(t_2 + t_5), \quad (3.8)$$

$$\delta_3 = fMt_3 + \frac{1}{T_R}N(t_1 + t_4 + t_5). \quad (3.9)$$

В приведённых формулах учтена частота работы блоков, выполняемых при поступлении сообщений о положении объектов от обработчиков мастер-камер, что происходит fM раз в секунду, а также частота работы блоков, которые работают при появлении свободного слайва, что происходит в среднем N/T_R раз в секунду.

Пользуясь тем, что $N \leq M$, а также тем, что максимальная частота анализа кадров обработчиками мастер камер является $f = 25\text{с}^{-1}$, а минимальное время выполнения команды ПЕРЕЙТИ() является $T_R = 500\text{мс}$, мы можем получить неравенства

$$\delta_1 > \delta_2 > \delta_3. \quad (3.10)$$

Неравенство $\delta_1 > \delta_2$ нелегко обосновать теоретически, однако тот факт, что оно выполняется для $M \in [0, 100]$, можно подтвердить при помощи небольшой программы на языке C, см. рис. 3.7.

Смысл данной программы в полном переборе всех допустимых пар (N, M) , для которых нужно проверить неравенство $\delta_1 > \delta_2$: если при каких то значениях неравенство не выполняется, программа выводит сообщение: «Не выполняется для: $M = \dots, N = \dots$ ». Ни для одной пары (N, M) при выполнении данной программы такого сообщения не было выведено, что подтверждает справедливость неравенства.

Подобный подход можно использовать для проверки большего диапазона значений, однако при этом необходимо дополнительно реализовать арифметику и логику чисел с «бесконечной» точностью. То есть необходимо ввести числа, точность которых ограничена лишь объемом ОЗУ для ЭВМ, а также реализовать операции сложения, вычитания, умножения, взятия логарифма и сравнения над этими числами.

Покажем, что $\delta_2 > \delta_3$:

$$\delta_2 - \delta_3 = (fM - \frac{N}{T_R})t_1 + \frac{N}{T_R}(t_2 - t_4),$$

оба слагаемые положительны, потому что из неравенств

$$M \geq N$$

$$f > 1/T_R$$

после умножения левых и правых положительных частей следует неравенство

$$fM > N/T_R,$$

а из равенства $t_2 = Mt_4$ и неравенства $M \geq 1$ следует $t_2 \geq t_4$.

Итак, для вычислительной сложности рассмотренных трёх алгоритмов синхронизации выполняется неравенство (3.10). Однако несмотря на то, что третья схема оптимальна по затрате процессорного времени, она обладает качественным недостатком: информация об объектах обновляется в диспетчере с задержкой, что может привести к нежелательным последствиям. Поэтому третья схема используется только для кластеров с одной слейв-камерой ($N = 1$), во всех остальных случаях следует использовать вторую алгоритмическую схему.

3.3. Вычисление угла упреждения и разработка алгоритмов управления наведением поворотных видеокамер на движущийся объект

Основное назначение модулей диспетчера и обработчиков слайва заключается в том, чтобы поворотные устройства автоматически наводились на движущиеся объекты (см. [141]). Диспетчер производит синхронизацию необходимых исходных данных для наведения (положение и скорость) между семантическим обработчиком мастер-камеры, который выделил движущийся объект, и обработчиком слейв-камеры, который должен провести наведение на объект на основании данных, переданных ему от

диспетчера. Введём для данных обработчиков видеокамер индексы: пусть t — индекс обработчика неподвижной камеры, и s — индекс обработчика поворотной камеры.

Пусть $\{x_{min}, y_{min}, x_{max}, y_{max}\}$ — координаты, ограничивающие положение объекта на изображении мастер-камеры, т.е. изображение объекта-цели является подмножеством минимального ограничивающего прямоугольника вида $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$. Введём обозначения $\vec{r}_1 = (x_1, y_1)$ для текущего положения той точки изображения объекта в координатах мастера t , в которую требуется произвести наведение, и $\vec{\tau}_1 = (\alpha_1, \beta_1)$ для соответствующих этому положению углов азимута и склонения поворотной камеры s . Из всего множества точек движения объекта координаты наведения \vec{r}_1 выбираются исходя из постановки задачи. В простом случае можно в качестве \vec{r}_1 выбрать «центр масс» множества точек объекта. Если обработчик мастер-камеры выполняет анализ движущихся объектов, то точка наведения может выбираться так, чтобы она попадала в область изображения лица для людей и в область изображения номерных знаков для автомобилей. Например, если применяются алгоритмы классификации по категориям «человек», «автомобиль», «группа людей» (см. п. 1.4), то после того, как анализ выявил принадлежность объекта к классу «человек», область лица можно установить чуть ниже координат $((x_{min} + x_{max})/2, y_{max})$. Для объектов класса «автомобиль» требуется применить какой-либо алгоритм локализации номерного знака на изображении, например, подобный [142]. Алгоритмы локализации данного типа применимы только в случае, если разрешение изображения объекта достаточно велико. Поэтому простой выбор средней точки объекта — самый надёжный, хоть и не самый оптимальный с точки зрения назначения поворотных устройств в аналитических видеосистемах.

Скорость объекта, определяемую перемещением в координатах точек изображения за секунду, будем обозначать $\vec{v} = (v_x, v_y)$, текущие угловые координаты положения камеры, из которого начинается процесс наведения, обозначим $\vec{\tau}_0 = (\alpha_0, \beta_0)$ (текущие азимут и склонение), γ_0 (текущая координата увеличения), f_0 (текущая координата фокусировки). Пусть $T(\vec{r}, \vec{\tau}_0, \gamma, \gamma_0, f, f_0)$ — функция времени, необходимого для перемещения поворотной камеры из координат $\vec{\tau}_0, \gamma_0, f_0$ в координаты \vec{r}, γ, f .

Обозначим $\vec{\tau} = A(\vec{r})$ — линейное преобразование сопряжения мастер-камеры t и слейв-камеры s (см. п. 2.6), которое действует в окрестности точки \vec{r} и переводит координаты мастер-камеры t в соответствующие координаты углов $\vec{\tau}$ слейв-камеры s , тогда, проанализировав обе части данного соотношения, получим $\vec{\omega} = (\omega_\alpha, \omega_\beta) = A(\vec{v})$ — угловая скорость объекта относительно точки закрепления камеры.

Для наведения на объект обработчик слейв-камеры должен выбрать координаты на мастере \vec{r} , в которые нужно перейти, а также нужное увеличение относительно мастера z . Увеличение выбирается так, чтобы, по возможности, объект занимал фик-

сированную, например, две трети, часть от изображения на поворотной камере по высоте или ширине. При известных $x_{min}, y_{min}, x_{max}, y_{max}$ увеличение z вычисляется по формуле(см. [143]):

$$z = \frac{2}{3} \min \left(\frac{w}{x_{max} - x_{min} + 1}, \frac{h}{y_{max} - y_{min} + 1} \right). \quad (3.11)$$

Здесь $w \times h$ — разрешение изображения видеокамер, а единица в знаменателях появляется из-за дискретности значений $x_{min}, y_{min}, x_{max}, y_{max}$.

Поворот в позицию с увеличением z и координатами $\vec{r} = (x, y)$ выполняется с помощью команды ПЕРЕЙТИ(s, m, x, y, z), см. п. 2.6. Если в качестве \vec{r} передать обработчик слайва использует текущие координаты положения объекта \vec{r}_1 , то за время выполнения поворота $T(\vec{\tau}_0, \vec{\tau}_1, \gamma_0, \gamma(z), f(z))$ объект успеет изменить своё положение и, скорее всего, окажется не в кадре поворотной видеокамеры. Поэтому необходимо сделать некоторое упреждение. В первом приближение скорость \vec{v} постоянна, поэтому координаты \vec{r} , в которые нужно произвести наведение, удовлетворяют уравнениям

$$\begin{cases} \vec{r} - \vec{r}_1 = \vec{v}T(\vec{\tau}, \vec{\tau}_0, \gamma, \gamma_0, f, f_0) \\ \vec{\tau} = A(\vec{r}), \vec{r}_1 = A(\vec{r}_1) \end{cases} \quad (3.12)$$

Применяя преобразование A к первому уравнению системы и выполнив подстановки, получим уравнение

$$\vec{\tau} - \vec{\tau}_1 = \vec{\omega}T(\vec{\tau}, \vec{\tau}_0, \gamma, \gamma_0, f, f_0) \quad (3.13)$$

В общем случае функция T имеет сложную зависимость от своих переменных, и уравнение (3.13) необходимо решать численно. Поскольку численное решение уравнения — это минимизация невязки, то задача поиска решения $\vec{\tau}$ для (3.13) сводится к задаче

$$\min_{\vec{\tau}} \|\vec{\tau} - \vec{\tau}_1 - \vec{\omega}T(\vec{\tau}, \vec{\tau}_0, \gamma, \gamma_0, f, f_0)\|, \quad (3.14)$$

где $\|\cdot\|$ — какая-либо норма в R^2 , например, евклидова. Заметим, что из уравнения (3.13) следует, что

$$\vec{\tau} = \vec{\tau}_1 + \vec{\omega}t, \quad (3.15)$$

где t — некоторая величина, измеряемая в секундах. Время поворота камеры T_R менее 4 секунд, поэтому $t \leq 4$ с. Следовательно, подставив (3.15) в (3.14), получим одномерную задачу

$$\min_{t \leq 4\text{c}} |t - T(\vec{\tau}_1 + \vec{\omega}t, \vec{\tau}_0, \gamma, \gamma_0, f, f_0)|. \quad (3.16)$$

Воспользовавшись, например, методом золотого сечения (см. [144]), эту задачу можно решить за шесть итераций. Погрешность решения t будет менее 100 миллисекунд, что следует из соотношения $\frac{4\text{c}}{2^6} \approx 63\text{мс}$. Обычно функция T нам не известна с большей

точностью, и, кроме того, время синхронизации положения между слейвом и мастером сопоставимо с данной величиной из-за того, что частоте обработке кадров $f = 25\text{с}^{-1}$ соответствует междукадровый интервал 40мс. Поэтому большее количество итераций не повысит действительной точности решения, даже если возможно уменьшить невязку (3.16).

Получив решение задачи t для (3.16), мы найдем координату упреждения $\vec{r} = A(\vec{r}_1 + \vec{\omega}t)$ для команды **ПЕРЕЙТИ()**.

В некоторых случаях можно получить аналитическое решение системы (3.13). Например, если поворотное устройство последовательно изменяет координаты при переходе из позиции в позицию, то функция T можно определить линейной моделью $T(\vec{r}, \vec{r}_0, \gamma, \gamma_0, f, f_0) = T_\alpha|\alpha - \alpha_0| + T_\beta|\beta - \beta_0| + T_\gamma|\gamma - \gamma_0| + T_f|f - f_0| + C_0 = T_\alpha|\alpha - \alpha_0| + T_\beta|\beta - \beta_0| + C$, где $T_\alpha, T_\beta, T_\gamma, T_f$ — характеристики скорости изменения координат, C_0 — постоянная, связанная с задержками передачи сообщений между диспетчером и обработчиками, а также с задержкой передачи команды в поворотное устройство по линии управления, и C — объединение трёх слагаемых, не связанных с изменением угловых координат, и поэтому не влияющих на решение исходной системы. Считая для определённости

$$\begin{cases} \alpha > \alpha_0, \\ \beta > \beta_0, \end{cases} \quad (3.17)$$

мы можем решить систему (3.13) относительно α и β :

$$\begin{aligned} & \begin{cases} \alpha - \alpha_1 = \omega_\alpha(T_\alpha(\alpha - \alpha_0) + T_\beta(\beta - \beta_0) + C), \\ \beta - \beta_1 = \omega_\beta(T_\alpha(\alpha - \alpha_0) + T_\beta(\beta - \beta_0) + C), \end{cases} \\ & \qquad \Downarrow \\ & \begin{cases} \alpha = \alpha_1 + \omega_\alpha \cdot \frac{T_\alpha(\alpha_1 - \alpha_0) + T_\beta(\beta_1 - \beta_0) + C}{1 - \omega_\alpha T_\alpha - \omega_\beta T_\beta}, \\ \beta = \beta_1 + \omega_\beta \cdot \frac{T_\alpha(\alpha_1 - \alpha_0) + T_\beta(\beta_1 - \beta_0) + C}{1 - \omega_\alpha T_\alpha - \omega_\beta T_\beta}. \end{cases} \end{aligned} \quad (3.18)$$

Если решение (3.18) удовлетворяет условиям (3.17), то оно является решением нашей исходной задачи $\vec{r} = A(\vec{r})$, где $\vec{r} = (\alpha, \beta)$. В противном случае необходимо рассмотреть остальные три варианта раскрытия модулей.

Система (3.13) может не иметь решений для всех четырёх случаев раскрытия модулей функции T . Такой результат следует интерпретировать как случай недостаточной скорости поворота камеры для того, чтобы успеть повернуться в сторону объекта-цели (объект «убегает» от камеры).

Итак, прямым следствием системы (3.12) является то, что только в частном случае при $\vec{v} = 0$ (объект неподвижен) при наведении можно использовать текущее положение объекта \vec{r}_1 в качестве параметра команды **ПЕРЕЙТИ()**, так как при $\vec{v} = 0$ получаем

$\vec{r} - \vec{r}_1 = 0$. Во всех остальных случаях требуется упреждение для уменьшения погрешности наведения поворотных видеокамер на движущийся объект. Покажем, что именно вычисление упреждения вносит наибольший значительный вклад в погрешность наведения.

3.4. Теоретическая оценка погрешностей наведения, обусловленных временными параметрами управления поворотными камерами

В ряде случаев при наведении слейв может «промахиваться» мимо объекта-цели из-за неточностей различного характера. Если погрешность наведения достаточно велика по отношению к выбранному углу зрения поворотной камеры (т.е. по отношению к увеличению на поворотной камере), то на изображении слейв-камеры не окажется объекта, на который производилось наведение.

Погрешность наведения обусловлена следующими факторами:

- 1) из-за неточных входных данных и из-за применения приближения малых углов преобразование координат $\vec{r} = A(\vec{r})$, переводящее координаты объекта $\vec{r} = (x, y)$ на мастер-камеры в азимут и склонение $\vec{\tau} = (\alpha, \beta)$ оптической оси слейва, вычислено неточно;
- 2) погрешность метода решения уравнения (3.13) оказывается достаточно велика, чтобы влиять на точность упреждения;
- 3) алгоритм выделения движения на изображении мастер-камеры может выдать ошибочные данные о положении объекта на кадре, поэтому скорость объекта \vec{v} , являющаяся параметром уравнения упреждения (3.13), тоже вычисляется с погрешностью;
- 4) уравнение вычисления упреждения (3.13) рассчитано на движение объекта с постоянной скоростью и не учитывает того, что объект может ускоряться и замедляться;
- 5) расчётное время поворота слейв-камеры T отличается от действительного в силу случайной природы некоторых процессов управления поворотным устройством.

Оценим влияние этих факторов на точность наведения при условии, что мастер-камера и слейв-камера закреплены в одной точке, а углы упреждения определяются формулами (3.18), т.е. функция времени наведения определяется формулой

$$T(\vec{\tau}, \vec{\tau}_0, \gamma, \gamma_0, f, f_0) = T_\alpha |\alpha - \alpha_0| + T_\beta |\beta - \beta_0| + C. \quad (3.19)$$

Оценка погрешности преобразования координат изображения мастер-камеры в углы наведения. Будем считать, что разрешение изображения мастера равно $w \times h$, $w = 352$, $h = 288$. Введём систему координат с началом в центре изображения мастера и предположим, что для преобразования $\vec{r} = A(\vec{r})$ задана сетка

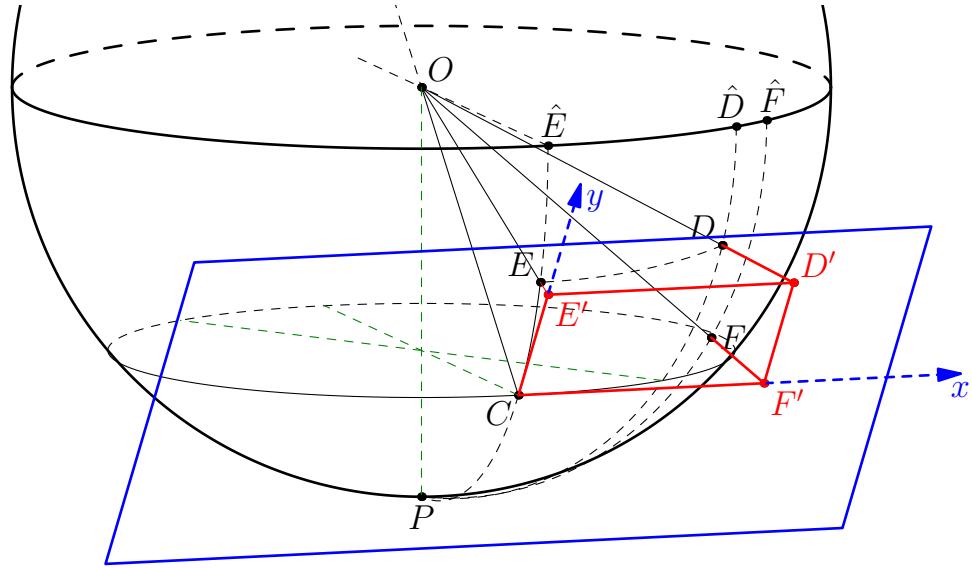


Рисунок 3.8. Точные углы азимута и склонения для точки (x, y) .

из пяти опорных точек, четыре точки располагаются в углах изображения мастера $C_1(-w/2, h/2)$, $C_2(w/2, h/2)$, $C_3(w/2, -h/2)$, $C_4(-w/2, -h/2)$ и одна точка в центре изображения C_0 с координатами $(0, 0)$.

Расчёт коэффициентов преобразования $A(\vec{r})$ производится на основании уравнения (2.25) отдельно для каждого треугольника из триангуляции, произведённой на опорных точках, см. п. 2.6. Поэтому, если погрешность исходных данных в точках C_i , $i = 0, 1, 2, 3, 4$, равна, соответственно, $(\Delta\alpha_i, \Delta\beta_i)$, то из свойств аффинного преобразования следует, что для вектора $\vec{r} = (x, y)$ погрешность преобразования $\vec{r}' = A(\vec{r})$ определяется формулами

$$\begin{aligned}\Delta\alpha &\leq \begin{cases} \max(\Delta\alpha_0, \Delta\alpha_1, \Delta\alpha_2), & (x, y) \in \Delta C_0 C_1 C_2, \\ \max(\Delta\alpha_0, \Delta\alpha_2, \Delta\alpha_3), & (x, y) \in \Delta C_0 C_2 C_3, \\ \max(\Delta\alpha_0, \Delta\alpha_3, \Delta\alpha_4), & (x, y) \in \Delta C_0 C_3 C_4, \\ \max(\Delta\alpha_0, \Delta\alpha_4, \Delta\alpha_1), & (x, y) \in \Delta C_0 C_4 C_1, \end{cases} \\ \Delta\beta &\leq \begin{cases} \max(\Delta\beta_0, \Delta\beta_1, \Delta\beta_2), & (x, y) \in \Delta C_0 C_1 C_2, \\ \max(\Delta\beta_0, \Delta\beta_2, \Delta\beta_3), & (x, y) \in \Delta C_0 C_2 C_3, \\ \max(\Delta\beta_0, \Delta\beta_3, \Delta\beta_4), & (x, y) \in \Delta C_0 C_3 C_4, \\ \max(\Delta\beta_0, \Delta\beta_4, \Delta\beta_1), & (x, y) \in \Delta C_0 C_4 C_1. \end{cases} \quad (3.20)\end{aligned}$$

Таким образом, точность подбора коэффициентов преобразования (2.24) напрямую определяется погрешностью входных данных, поэтому можно оценивать погрешность $\Delta\vec{r}'$ преобразования $A(\vec{r})$ внутри выпуклой оболочки опорных точек как

$$\|\Delta\vec{r}'\| \leq \max_i \|(\Delta\alpha_i, \Delta\beta_i)\|. \quad (3.21)$$

Для того, чтобы оценить погрешность линейной модели преобразования (2.24), приведём точные формулы для $\vec{r}' = A(\vec{r})$, воспользовавшись тем, что мастер и

слей в закреплены в одной точке. Допустим, нам известны углы азимута и склонения $\vec{\tau}_0 = (\alpha_0, \beta_0)$ для центра изображения мастер-камеры. Найдём углы $\vec{\tau} = (\alpha, \beta)$, соответствующие смещению-точке $\vec{r} = (x, y)$ на изображении мастера.

Для решения этой задачи рассмотрим рис. 3.8. На нём синим цветом схематично отображена плоскость изображения мастер-камеры с системой координат Cxy , где C — центр изображения, а точка $D'(x, y)$ — точка, соответствующая вектору смещения $\vec{r} = (x, y)$. Точки F' и E' являются проекциями точки D' на оси Cx и Cy , соответственно, т.е. координаты точек $F'(x, 0)$ и $E'(0, y)$. Пусть точка O является местом крепления поворотной видеокамеры и мастер-камер. Введём сферу в центре с точкой O , которая касается плоскости $CE'D'$ изображения мастера в точке C , т.е. радиус сферы $R = |OC|$. Введём также плоскость, относительно которой отсчитывается угол склонения поворотной камеры, т.е. такую плоскость, которая проходит через точку O , и углы между прямыми OC , OD' и данной плоскостью равны β_0 , β , соответственно. Пусть прямая, проходящая через точку O и перпендикулярная плоскости отсчёта склонения, пересекает сферу в точке P , которую мы будем называть полюсом сферы. Угол между плоскостями OPC и OPD' является разностью углов азимута α_0 и α , т.е. $(\widehat{OPC}, \widehat{OPD'}) = \alpha - \alpha_0$ при выборе направления отсчёта против часовой стрелки относительно вектора \overrightarrow{OP} . Обозначим пересечения прямых OD' , OE' , OF' с рассматриваемой сферой как, соответственно, точки D , E , F , а также пересечение плоскостей OPD , OPE , OPF с окружностью, образованной пересечением плоскости отсчёта со сферой, как, соответственно, точки \hat{E} , \hat{D} , \hat{E} . Не сложно показать, что плоскости OPC и OPE совпадают, и что $\angle \hat{E}OC = \beta_0$, $\angle \hat{D}OD = \beta$, $\angle \hat{E}O\hat{D} = \alpha - \alpha_0$. Дополнительно введём обозначения для углов $\angle COF = \varphi$, $\angle COE = \psi$, $\angle EOD = \varphi'$, а также для углов, составляющих половины горизонтального и вертикального углов зрения мастер-камеры, соответственно, φ_h и ψ_v .

Справедлива следующая последовательность выражений:

$$\tg \varphi = \frac{|CF'|}{|OC|} = \frac{x}{R} = \frac{x}{w/2} \cdot \frac{w/2}{R} = \frac{x}{w/2} \tg \varphi_h, \text{ аналогично,} \quad (3.22)$$

$$\tg \psi = \frac{y}{h/2} \tg \psi_v, \quad (3.23)$$

$$|OD'| = \sqrt{|OC|^2 + |CE'|^2 + |CF'|^2}, |OC| = R,$$

$$|CE'| = |OC| \tg \angle COE' = R \tg \psi, |CF'| = |OC| \tg \angle COF' = R \tg \varphi,$$

$$|OE'| = \frac{|OC|}{\cos \angle COF'} = \frac{R}{\cos \psi}, \text{ следовательно,}$$

$$\sin \varphi' = \frac{\tg \varphi}{\sqrt{1 + \tg^2 \varphi + \tg^2 \psi}} = \frac{|D'E'|}{|OD'|}, \text{ и, аналогично,} \quad (3.24)$$

$$\cos \varphi' = \frac{1}{\cos \psi \sqrt{1 + \tg^2 \varphi + \tg^2 \psi}} = \frac{|OE'|}{|OD'|}. \quad (3.25)$$

Рассмотрим сферический треугольник PDE . В нём известны все три стороны-дуги $\widehat{PE} = 90^\circ - (\beta_0 - \psi)$, $\widehat{DE} = \varphi'$, $\widehat{PD} = 90^\circ - \beta$, а также двугранные углы $\angle EPD = (\widehat{OPE}, \widehat{OPD}) = \alpha - \alpha_0$ и $\angle PED = (\widehat{OD'E'}, \widehat{OE'P}) = 90^\circ$, поскольку $E'D' \parallel CF'$, $CF' \perp OE'P$, следовательно, $E'D' \perp OE'P$, и из того, что $E'D' \subset OE'D'$, следует $OE'D' \perp OE'P$.

По теореме косинусов для сферического $\triangle PDE$

$$\cos \widehat{PD} = \cos \widehat{PE} \cdot \cos \widehat{ED} + \sin \widehat{PE} \cdot \sin \widehat{ED} \cdot \cos \angle PED, \text{ т.е.}$$

$$\sin \beta = \sin(\beta_0 - \psi) \cos \varphi',$$

и по теореме синусов для сферического $\triangle PDE$

$$\frac{\sin \widehat{DE}}{\sin \angle DPE} = \frac{\sin \widehat{PD}}{\sin \angle PED}, \text{ т.е. } \sin(\alpha - \alpha_0) = \frac{\sin \varphi'}{\cos \beta}.$$

Таким образом, из соотношений для сферического треугольника следуют формулы

$$\alpha = \alpha_0 + \arcsin \frac{\sin \varphi'}{\cos \beta},$$

$$\beta = \arcsin (\cos \varphi' \cdot \sin(\beta_0 - \psi)), \quad (3.26)$$

подставив в которые соотношения (3.22)–(3.25), получим окончательное решение, зависящее от w , h , φ_h , ψ_v , $\vec{r} = (x, y)$.

Пользуясь точной формулой (3.26) при известном азимуте α_0 и склонении β_0 , возможно вычислить углы, соответствующие углам изображения мастера C_1, C_2, C_3, C_4 , а затем для каждого из $\triangle CC_1C_2, \triangle CC_2C_3, \triangle CC_3C_4, \triangle CC_1C_4$ вычислить коэффициенты аффинного преобразования (2.24). Максимум нормы разности преобразований (3.26) и (2.24) на множестве точек изображения мастер-камеры является оценкой погрешности, обусловленной применением линейного приближения. При применении аналитического подхода к поиску максимума отклонения возникает задача поиска нулей производной разности функций вида (3.26) и (2.24), т.е. возникает необходимость решения трансцендентных уравнений. При помощи современной вычислительной техники задачу поиска максимума отклонения можно решить с помощью полного перебора значений отклонения во всех точках изображения с разрешением $w \times h = 352 \times 288$.

Рассмотрим, например, частный случай установки мастер-камеры с горизонтальным углом зрения, равным 30° , т.е. $\varphi_h = 15^\circ$ и $\psi_v = \operatorname{arctg}\left(\frac{288}{352} \operatorname{tg} \varphi_h\right) \approx 12,4^\circ$. При этом эффективная дальность наблюдения составляет 70м, и при высоте подвеса камеры 5м в предположении, что поверхность наблюдения является плоскостью и перпендикулярна линии подвеса, угол склонения β_0 оптической оси мастер-камеры примерно равен 25° . На рис. 3.9 изображён трёхмерный график модуль разности векторов \vec{r} для значений, полученных с помощью формул (3.26) и (2.24), вычисленных для всех

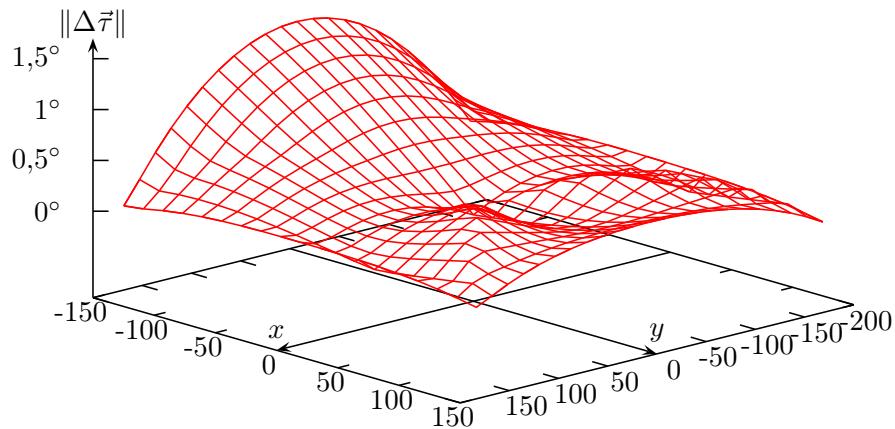


Рисунок 3.9. График зависимости нормы разности $\|\vec{\Delta\tau}\|$ точного и линейного преобразований от координат изображения (x, y) .

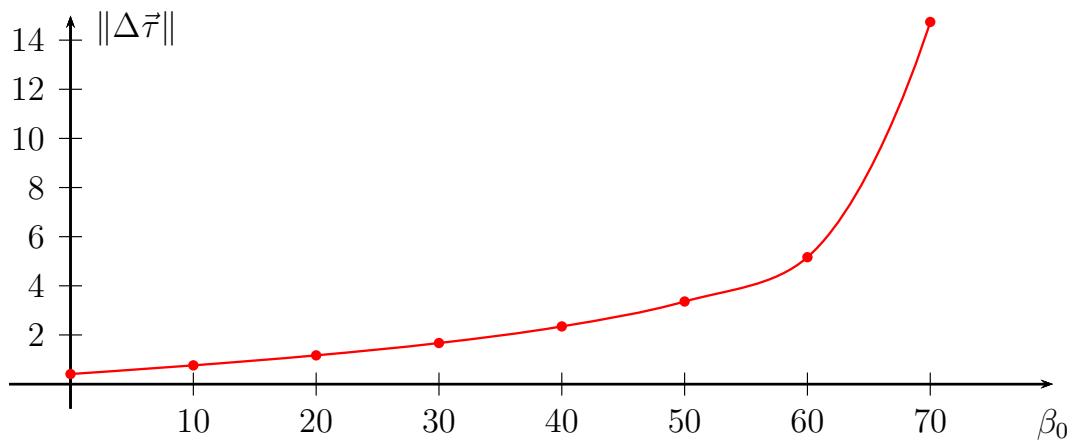


Рисунок 3.10. Зависимость $\|\vec{\Delta\tau}\|$ от наклона оптической оси β_0 в градусах.

точек изображения мастера. С помощью анализа этого графика можно оценить максимальную погрешность для данных условий: $\|\vec{\Delta\tau}\| < 0,5^\circ$.

На графике 3.9 прослеживается закономерность: чем больше угол β склонения оптической оси слайва, тем больше погрешность линейной модели. С помощью автоматического анализа данных для крупного массива числовых значений нормы разности $\|\vec{\Delta\tau}\|$ во всех точках изображения для различных углов β_0 наклона оптической оси мастера, можно эмпирически установить зависимость $\|\vec{\Delta\tau}\|$ от β_0 при горизонтальном угле зрения мастер-камеры, равным 30° , см. график на рис. 3.10.

Из анализа полученного графика следует, что погрешность линейного приближения превышает 5° при углах $\beta_0 > 50^\circ$. Угол 5° соответствует примерным линейным размерам 4м на расстоянии 50м от точки закрепления камер. Таким образом, для получения изображения, например, человека, с помощью наведения слайва, при данных параметрах необходимо установить увеличение, при котором угол зрения слайв-камеры не меньше 5° , что сильно ограничивает разрешение изображения.

С другой стороны, погрешность линейного приближения не превышает 2° при $\beta_0 < 30^\circ$, поэтому в данном случае оно применимо.

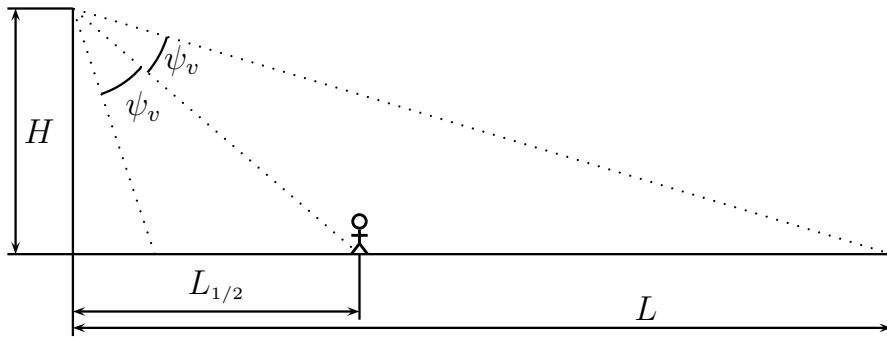


Рисунок 3.11. Положение объекта в центре изображения мастер-камеры.

При высоких значениях погрешности линейных формул, приближающих сопряжение поворотной и неподвижной видеокамер в определённой точке изображения мастера, необходимо ввести для данной точки изображения дополнительную опорную точку \mathcal{R} (см. п. 2.6), либо, если сопрягаемые камеры расположены в одной точке, воспользоваться точными формулами (3.26). Точный расчёт сопряжения для камер, установленных на расстоянии, на практике не применяется из-за сложности точной оценки параметров взаимного расположения камер.

Влияние временных параметров на погрешность наведения. Согласно уравнению (3.13), временные параметры, определяющие управление поворотной камерой при наведении на объект — это скорость перемещения объекта по изображению \vec{v} , измеряемая в пикселях за секунду, и время исполнения команды наведения T . Оценим влияние этих параметров на погрешность вычисления азимута и склонения оси поворотной камеры при наведении на движущийся объект.

Пусть горизонтальный угол зрения мастер-камеры равен 30° , т.е. $\varphi_h = 15^\circ$, $\psi_v = 12,5^\circ$, наблюдение ведётся на дистанции до $L = 100\text{м}$, поверхность наблюдения плоская, высота установки камеры $H = 5\text{м}$. Пусть объект, на который производится наведение — это человек, находящийся в центре изображения мастер-камеры. С помощью чертежа на рис. 3.11 можно рассчитать расстояние $L_{1/2}$ от объекта до основания подвеса мастер-камеры:

$$L_{1/2} = H \cdot \operatorname{tg}(\operatorname{arctg} \frac{L}{H} - \psi_v) \approx 20\text{м}.$$

Таким образом, угловой размер человека в центре изображения примерно равен $0,4\text{м}/20\text{м} = 0,02\text{рад}$. Поэтому ширина изображения человека равна $352 \cdot \frac{0,02}{2\varphi_h} \approx 13$ пикселям. Определение положения движущегося объекта с помощью алгоритма выделения движения может быть не точным из-за свойств изображения: например, алгоритм может выделять только левую сторону человека или только правую. В случае скачка в определении человека с левой стороны на правую и наоборот максимальную погрешность вычисления горизонтальной составляющей скорости \vec{v} можно оценить по порядку величины как $\Delta v_x = 10\text{с}^{-1} \approx 13$ пикселов/с. Отсюда возникает погрешность

наведения по азимуту, если принять, что время позиционирования камеры равно одной секунде

$$\Delta\alpha \approx 0,02 \text{рад} \approx 1^\circ,$$

поскольку данный угол соответствует скорости перемещения объекта по изображению Δv_x в течение одной секунды. В общем случае погрешность наведения пропорциональна произведению времени исполнения команды T и погрешности измерения скорости Δv_x .

При изменении скорости в момент выполнения команды наведения погрешность определения скорости можно оценить как $\Delta \vec{v} = \vec{a}T$, где \vec{a} — ускорение изображения, измеряемое в пикселях в секунду за секунду. Оценивая физическое ускорение объекта как 10м/c^2 (примерное ускорение бегуна в первую секунду на дистанции 100м) и предполагая, что это ускорение полностью перейдет в компоненту ускорения изображения a_x (это так в случае движения перпендикулярно оси камеры), получим

$$a_x \approx \frac{352}{2L_{1/2} \operatorname{tg} \varphi_h} \cdot 10 \text{м/c}^2 \approx 330 \text{c}^{-2},$$

т.е. за секунду $\Delta v_x = 330 \text{c}^{-1}$, что приведёт к погрешности примерно $\Delta\alpha \approx 30^\circ$ для азимута. Но на практике невозможно постоянно выдерживать такие ускорения, поэтому погрешности данной величины обычно не достигаются, или достигаются в единственной из нескольких попыток наведения на один объект.

Расчётное время выполнения T команды наведения может отличаться от действительного в силу различных причин: задержки передачи сигналов в ЛВС, задержки передачи сигналов в линии управления, задержки, обусловленные работой процедур передачи сигналов низкого уровня в операционной системе, которая управляет программными модулями аналитической видеосистемы. Погрешность времени выполнения может составлять до 0,2с для некоторых поворотных видеокамер. При физической скорости рассматриваемого объекта 10м/с (быстрый бег), движущегося перпендикулярно оптической оси, скорость изображения будет горизонтальна,

$$v_x = \frac{352}{2L_{1/2} \operatorname{tg} \varphi_h} \cdot 5 \text{м/c} \approx 330 \text{ пикселя/c},$$

поэтому погрешность $\Delta T = 0,2\text{с}$ обозначает погрешность определения положения объекта $330 \cdot 0,2 \approx 66$ пикселов, что примерно соответствует погрешности азимута $\Delta\alpha = 5^\circ$.

Вывод соотношения для общей погрешности наведения. Чтобы получить общую формулу для погрешности, введем обозначение $\vec{r} = A(\vec{r})$ для преобразования по формулам (2.24) и $\hat{\vec{r}} = \hat{A}(\vec{r})$ для преобразования по формулам (3.26). Пусть T , \vec{v} — точные значения времени наведения и средней скорости изображения объекта за

время наведения, соответственно, а ΔT , $\Delta \vec{v}$ — отклонения от точных значений из-за ошибки оценки. Воспользовавшись уравнением (3.13) и линейностью преобразования A , получим выражение для оценки отклонения $\Delta \vec{\tau} = (\Delta\alpha, \Delta\beta)$ действительных значений углов азимута и склонения от расчётных:

$$\begin{aligned}\Delta \vec{\tau} &= A(\vec{r}_0 + (\vec{v} + \Delta \vec{v})(T + \Delta T)) - \hat{A}(r_0 + \vec{v}T) \approx \\ &\approx \Delta A(\vec{r}) + A(T\Delta \vec{v}) + A(\vec{v}\Delta T),\end{aligned}\quad (3.27)$$

где $\vec{r} = \vec{r}_0 + \vec{v}T$ — точное конечное положение объекта, \vec{r}_0 — начальное положение изображения объекта на кадре, и $\Delta A = A(\vec{r}) - \hat{A}(\vec{r})$.

Таким образом, погрешность наведения $\|\Delta \vec{\tau}\|$ складывается из трёх величин: погрешности, обусловленной неточной моделью преобразования координат мастера в углы азимута и склонения оптической оси слейв-камеры и неточными входными данными процедуры сопряжения (первое слагаемое формулы (3.27)), погрешности, обусловленной неточной оценкой скорости с погрешностью $\Delta \vec{v}$ (второе слагаемое формулы (3.27)), и погрешности, возникающей из-за погрешности ΔT расчётного времени поворота (третье слагаемое формулы (3.27)).

Погрешность ввода данных при сопряжении можно не учитывать, поскольку она соответствует небольшим значениям углов. Погрешность линейной модели преобразования A можно устраниТЬ, либо применяя точные формулы расчёта углов, либо используя опорные точки в точках изображения с максимальной погрешностью. Погрешность, возникающая из-за погрешностей $\Delta \vec{v}$ и ΔT , обусловлена современным состоянием теории алгоритмов, анализирующих движущиеся объекты, а также техническими характеристиками поворотных видеокамер и вычислительных машин, поэтому в рамках данной работы она не может быть устранена. Из оценок, приведённых выше, следует, что для человека, находящегося в центре изображения мастер-камеры, у которой угол зрения 30° и дальность наблюдения $L = 100\text{м}$, при условии, что он двигается с большим ускорением и скоростью, но скорость, тем не менее, определена точно, погрешность наведения может складываться из величин 15° (движение с ускорением) и 5° ($\Delta T = 0,2\text{с}$), т.е.

$$\|\Delta \vec{\tau}\| < 20^\circ.$$

При погрешности наведения поворотной камеры 20° и угле зрения сопряжённой неподвижной камеры 30° увеличение изображений объектов более, чем в 1,5 раза, невозможно при условии требования высокой достоверности сопоставления изображений. Либо необходимо ограничить угол зрения поворотной камеры снизу, то есть получать изображения невысокого разрешения по сравнению с изображением мастера, либо учитывать высокую долю промахов при наведении на объект с углом зрения, в несколько раз меньшим, чем погрешность наведения.

Данная оценка приведена для наихудших условий, и выполняется на практике редко. Кроме того, промахи наведения, обусловленные погрешностью определения скорости объекта и времени наведения, можно частично компенсировать многократными попытками наведения, поскольку погрешности $\Delta\vec{v}$ и ΔT носят случайный характер.

3.5. Определение критерия идентичности объектов на изображениях видеокамер

Одной из актуальных задач видеосистем семантического сжатия является задача повторной идентификации объектов, обнаруженных на одной из видеокамер, при появлении в поле зрения других видеокамер. Повторная идентификация позволяет представить операторам видеосистемы важную семантическую информацию о перемещении объектов в зоне контроля видеосистемы в целом. Если области обзора видеокамер не пересекаются, то повторная идентификация затруднена и не может быть выполнена с высокой достоверностью. Однако для видеокамер с пересекающимися зонами обзора в данной работе представлен простой и достаточно надёжный алгоритм повторной идентификации на основе сопряжения и синхронизации.

Алгоритм сопоставления объектов в общей зоне зрения на изображении неподвижных видеокамер основан на том, что известна функция преобразования $\vec{u}_{ds}(\vec{v})$ (формула (2.8), см. п. 2.3), которая позволяет получить для точки $\vec{v} = (x, y)$ мастера № s соответствующую ей точку на мастере № d , при условии, что точка $\vec{v} = (x, y)$ находится в зоне пересечения на изображении. В противном случае функция не определена. Можно было бы задать функцию $\vec{u}(\vec{v})$ табличным способом, но в этом случае настройка системы оказалась бы очень громоздкой. В п. 2.3 указан способ вычисления функции $\vec{u}_{ds}(\vec{v})$, если заданы её значения в небольшом количестве точек (x, y) . При этом необходимо учитывать погрешность вычисления преобразования из-за неточности данных исходной сетки преобразования и из-за несовершенства метода подбора преобразования. На практике отклонение преобразования $\vec{u}(\vec{v})$ не превышает двух градаций координат изображения мастер-камеры.

Рассмотрим критерий сопоставления объектов, выделенных семантическими обработчиками мастер-камер в общей зоне обзора (см. [145]). Для этого будем обозначать объекты, выделенные обработчиком неподвижной видеокамеры, в виде набора параметров $(n, x_{min}, y_{min}, x_{max}, y_{max})$, где n — порядковый номер объекта за всё время работы обработчика мастер-камеры, а $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$ — минимальный прямоугольник, ограничивающий объект № n на изображении мастер-камеры. Пусть обработчик s обнаружил множество объектов $\{(n^s, x_{min}^s, y_{min}^s, x_{max}^s, y_{max}^s)\}$, а обработчик d обнаружил множество объектов $\{(n^d, x_{min}^d, y_{min}^d, x_{max}^d, y_{max}^d)\}$. При помощи сообщения **ПЕРЕДАЧА_ОБЪЕКТОВ** обработчик d может передать информацию об объектах в обработчик s , синхронизируя таким образом информацию о положении объектов

между обработчиками. Пусть

$$\begin{aligned} x^s &= \frac{x_{min}^s + x_{max}^s}{2}, & x^d &= \frac{x_{min}^d + x_{max}^d}{2}, \\ y^s &= \frac{y_{min}^s + y_{max}^s}{2}, & y^d &= \frac{y_{min}^d + y_{max}^d}{2}, \\ \vec{v}^s &= (x^s, y^s), & \vec{v}^d &= (x^d, y^d), \end{aligned} \quad (3.28)$$

тогда для каждой пары объектов (n^s, n_{db}^d) мы проверим условие

$$\begin{cases} \vec{u}_{ds}(\vec{v}^s) \in \{(x, y) | x_{min}^d \leq x \leq x_{max}^d, y_{min}^d \leq y \leq y_{max}^d\}, \\ \vec{u}_{sd}(\vec{v}^d) \in \{(x, y) | x_{min}^s \leq x \leq x_{max}^s, y_{min}^s \leq y \leq y_{max}^s\}. \end{cases} \quad (3.29)$$

Если условие (3.29) выполняется, то мы считаем, что объекты n^s и n^d являются на самом деле одним объектом. Таким образом мы устанавливаем соответствие $n^s \leftrightarrow n^d$.

Для того, чтобы условие (3.29) можно было применять, необходимо, чтобы за время синхронизации данных о положении объекта не происходило сильного смещения изображений движущихся объектов: сумма смещения объекта за время синхронизации и погрешности вычисления преобразований \vec{u}_{sd} и \vec{u}_{ds} не должна превышать характерных размеров объекта в координатах изображения. Покажем применимость критерия (3.29).

Сначала оценим время синхронизации данных между обработчиками, которое влияет на величину рассогласования данных о положении объектов. Пусть обработчик мастер-камеры анализирует кадры с частотой $f = 15\text{с}^{-1}$. Поскольку сообщения от других обработчиков принимаются с промежутком времени $T_f = 1/f = 1/15\text{с}$ (T_f — время между обработкой последовательных кадров), то для времени синхронизации δt справедливы неравенства

$$T_m < \delta t < T_m + T_f,$$

где $T_m < 30\text{мс} < T_f/2$ — время пересылки сообщения от обработчика к обработчику. Оценка 30мс для времени T_m следует из способа передачи данных между вычислительными процессами в многозадачных операционных системах с вытесняющей многозадачностью, см., например [146]. Временем передачи данных по сети ЛВС с пропускной способностью 100Мбит/с в данном случае можно пренебречь. Считая, что δt — равномерно распределенная величина, получим, что в среднем

$$\delta t = \frac{1}{T_m + T_f - T_m} \int_{T_m}^{T_m + T_f} t \cdot dt = \frac{1}{T_f} \frac{t^2}{2} \Big|_{T_m}^{T_m + T_f} = T_m + T_f/2,$$

то есть $\delta t \approx 63\text{мс}$, учитывая $T_m = 30\text{мс}$.

Зная время синхронизации δt , можно установить скорости объектов, при которых смещение не превышает характерные размеры, однако нестрогое доказательство применимости критерия (3.29) можно провести с помощью иллюстрации: на рис. 3.12



а) начальный момент б) изображение через 40мс в) разность изображений

Рисунок 3.12. Иллюстрация величины смещения объекта за один кадр. Слева — первый кадр, посередине — второй кадр, справа — изображение с выделенной разностью кадров.

изображены два последовательных кадра и разность между ними, причём видно, что смещения и автомобилей и людей, подсвеченные на правом рисунке, не превышают 1–2 точек изображения.

Приведём формальное обоснование того, что критерий (3.29) можно применять, несмотря на погрешность вычисления преобразований \vec{u} , несмотря на задержку δt при передаче объектов от обработчика к обработчику, а также несмотря на то, что в качестве информации о положении объекта используются лишь координаты центра ограничивающей рамки, а сама рамка определяется с некоторой погрешностью, связанной с вероятностным характером результата работы алгоритма выделения движения.

На практике алгоритм выделения движения не выдаёт изображения движущихся объектов менее четырёх точек по ширине и длине, поскольку данное ограничение даёт возможность отделить шумовые движения от действительных объектов. Из всех типов объектов, а именно: транспорта, групп людей и отдельных людей, представляющих интерес с точки зрения обнаружения, только человек обладает наименьшим размером по ширине, измеряемым точками изображения. Это обусловлено тем, что видеокамеры всегда установлены так, чтобы люди на изображении занимали вертикальное положение при ходьбе. Такие планы наиболее привычны для оператора аналитической системы, изображение камера «под углом», наоборот, трудно воспринимать.

Рассмотрим предельный случай, когда изображение человека по ширине составляет 4 точки. Поскольку физически этот размер соответствует в среднем половине метра, мы можем оценить скорость движения человека, при которой время обмена сообщениями между обработчиками не повлияет на выполнение условия (3.29). Условие (3.29) может не сработать, если координаты «центра» объекта $\vec{v}^s = (x^s, y^s)$ на текущем обработчике s и координаты $\vec{v}^d = (x^d, y^d)$, полученные от обработчика d и

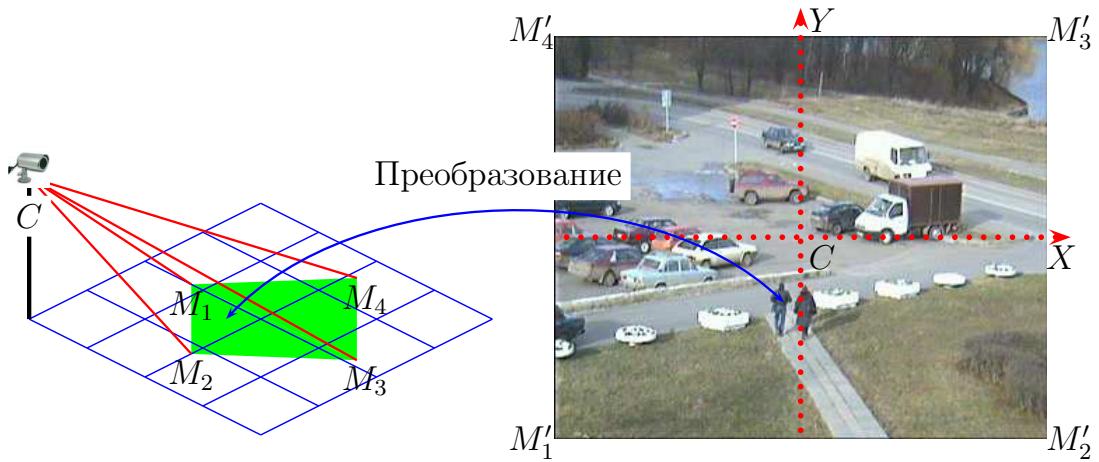


Рисунок 3.13. Проективное преобразование физических координат в координаты изображения.

«устаревшие» за счёт задержки передачи сообщений на время δt , будут отличаться более, чем на 2 точки. Следовательно, условие (3.29) применимо, если скорость движения изображения человека

$$v_{im}^h < \frac{2 \text{ точки}}{\delta t}. \quad (3.30)$$

Чтобы оценить предельную физическую скорость v_{ph}^h человека, которая соответствует скорости v_{im}^h в формуле (3.30), нужно выяснить, какому минимальному физическому смещению соответствует смещение изображения на 2 точки в рассматриваемом случае. Для этого обратимся к рис. 3.13.

Изображение на плоскости $M'_1M'_2M'_3M'_4$ является центральной проекцией поверхности $M_1M_2M_3M_4$, образованной конусом с прямоугольным сечением, который ограничивается плоскостями CM_1M_2 , CM_2M_3 , CM_3M_4 , CM_1M_4 , где C — точка закрепления камеры. Прямые пересечения плоскостей $CM_1M_2 \cap CM_3M_4$ и $CM_2M_3 \cap CM_1M_4$ перпендикулярны (введем обозначения $CX = CM_1M_2 \cap CM_3M_4$ и $CY = CM_2M_3 \cap CM_1M_4$), а двугранные углы между этими плоскостями связаны соотношением

$$(CM_1\widehat{M_2}, CM_3M_4) = \frac{3}{4}(CM_2\widehat{M_3}, CM_1M_4).$$

Биссекторные плоскости данных углов пересекаются по прямой, которая является оптической осью камеры. Все перечисленные свойства обусловлены оптической схемой объектива и прямоугольной формой ПЗС-матрицы, стороны которой соотносятся как 3:4.

Очевидно, что физическое смещение при фиксированном смещении на изображении будет минимальным, если оно параллельно плоскости CXY . В силу уже упомянутых эргономических ограничений, диктующих «естественное» положение камеры, соответствующие нашему зрительному восприятию, смещение параллельно плоскости CXY возможно лишь вдоль линий, параллельных линиям M_1M_2 и M_3M_4 , если

$M_1M_2M_3M_4$ — плоская поверхность. Во всех остальных случаях смещение увеличивается за счёт дополнительной проекции на оптическую ось. Для рассматриваемого нами человека, ширина изображения которого составляет 4 точки, а его профиль в ширину примерно половина метра, минимальное смещение, соответствующее смещению на 2 точки, происходит вдоль линии $M'_3M'_4$, то есть вдоль «ширины», и поэтому это минимальное смещение примерно соответствует 25 см, и предельная физическая скорость

$$v_{ph}^h = \frac{25\text{см}}{\delta t} = \frac{25}{63}\text{см/мс} \approx 4\text{м/с} \approx 14\text{км/ч.}$$

Такая скорость выше средней скорости пешехода, хотя и ниже рекордной скорости бега. В подавляющем большинстве случаев это ограничение выполняется.

Если ширина человека больше четырёх точек, то ограничение предельной скорости v_{ph}^h остается неизменным, поскольку, несмотря на увеличение допуска смещения изображения в точках изображения, этому допуску соответствует неизменное минимальное физическое смещение в половину профиля человека — 25 см.

На сопоставление групп людей погрешность задержки передачи координат влияет даже меньше, чем на сопоставление отдельных людей, поскольку скорость перемещения групп людей редко бывает высокой, а допуск на смещение обычно выше, так как ширина изображения двух и более идущих рядом людей по крайней мере не меньше ширины изображения одного человека.

Для транспортных средств оценка предельной скорости v_{ph}^v основана на том, что мгновенная скорость машин всегда направлена вдоль кузова, а длина малогабаритной машины составляет около 3 метров. Следовательно, центр может изменить своё положение на 1,5 метра за время δt , и при этом условие (3.29) ещё останется в силе. Отсюда

$$v_{ph}^v = \frac{1,5\text{м}}{\delta t} = \frac{1,5\text{м}}{63\text{мс}} \approx 24\text{м/с} \approx 86\text{км/ч,}$$

что не является максимумом для всех транспортных средств, но выполняется в большинстве случаев в тех случаях, когда применяется аналитическая видеосистема.

Осталось рассмотреть последний фактор, потенциально противоречащий применению условия (3.29) на практике — это неточное выделение «рамок» объектов, и ограничение информации о положении объекта до уровня прямоугольника, причем со сторонами, параллельными вертикальным и горизонтальным осям. Последнее может привести к тому, что два разных объекта будут обладать пересекающимися рамками, и условие (3.29) будет выполнено для пары *различных* объектов. И так действительно может случиться, например, для двух крупногабаритных грузовиков, которые двигаются параллельным курсом под углом 45° к осям CX и CY на изображении. Однако такие случаи на практике редки.

Частично ложные сопоставления отсеиваются за счёт двусторонней направленности проверки (3.29). Неточность выделения рамок также не является препятствием. С помощью специального метода (см. [147, 148]) можно оценить два показателя: математическое ожидания и дисперсию площади пересечения «идеального» ограничивающего прямоугольника объекта с прямоугольником, который выдает алгоритм выделения движения, волях от площади идеальной рамки. Эти показатели составляют, соответственно, 0,9 и 0,1. Из предположения, что доля площади перекрытия распределена нормально, следует вывод о том, что практически всегда центр не отклоняется от своего действительного положения более чем на половину от ширины или от длины в самом худшем случае, поскольку с вероятностью, превышающей 0,99, площадь перекрытия прямоугольника, выданного алгоритмом, с действительным прямоугольником, превышает половину. Таким образом, условие (3.29) сработает в обе стороны с большой вероятностью.

Все три фактора: неточность вычисления функции \vec{u} , погрешность за счёт задержки δt , и неточное выделение ограничивающего прямоугольника алгоритмом выделения движения могут сложиться вместе так, что простая проверка на совпадение объектов не сработает на *одном* кадре. Однако подобная проверка проводится на *каждом* кадре, и, хотя не все из приведенных погрешностей являются независимыми по отношению как друг к другу, так и сами к себе на разных кадрах, всё же эксперимент показывает, что в течение 5 кадров, то есть за треть секунды при частоте $f = 15$ кадров/с, связь между объектами обязательно будет установлена.

Для того, чтобы сохранить факт сопоставления $n^s \leftrightarrow n^d$ в базе данных, обработчик неподвижной камеры применяет специальный алгоритм. Информация об объектах хранится в таблице, в колонках у которой записываются различные свойства объекта, в том числе свойство n_{db} — порядковый номер объекта в рамках всей совокупности обработчиков. Данное свойство является ключевым полем таблицы объектов. Как только обработчик мастер-камеры выдаёт объект n (по нумерации обработчика) в первый раз, в базе создаётся новая запись с ключём n_{db} . Для отображения факта связи объектов существует специальное свойство c , называемое идентификатором связи, также записанное в таблице описания объектов. Если $c = <\text{служебное значение}> = <\text{сл. знач.}>$, то считается, что объект не был ни с кем связан, иначе все объекты, которых в таблице записано значение c , на самом деле являются одним физическим объектом, появлявшимся на разных камерах. Дополнительно вводится ограничения $c < C$ и $n_{db} < C$, поскольку значения чисел на ЭВМ не могут быть неограниченными. Наиболее логично поставить ограничение 32-х разрядного слова: $C = 2^{32}$. Исходя из этого определения идентификатора связи можно составить уникальный идентификатор u для физических объектов, т.е. число, которое будет различно для различных

объектов, но одинаково для изображений одного объекта с пересекающихся мастер-камер:

$$u = \begin{cases} n_{db} + C, & \text{если } c = \langle \text{служебное значение} \rangle, \\ c & \text{в противном случае.} \end{cases}$$

Алгоритм, поддерживающий подобную структуру идентификаторов, выполняется следующим образом: допустим, установлен факт соответствия $n^s \leftrightarrow n^d$, причём объекту n^d соответствует значение ключевого поля в таблице объектов n_{db}^d , а объекту n^s соответствует значение ключевого поля n_{db}^s . В этом случае необходимо выполнить следующую процедуру: пусть для значения n_{db}^s в колонке c содержится значение A , а для ключа n_{db}^d — значение B , тогда алгоритм распадается на четыре ветви:

$A = \langle \text{сл. знач.} \rangle$	$A \neq \langle \text{сл. знач.} \rangle$	$A = \langle \text{сл. знач.} \rangle$	$A \neq \langle \text{сл. знач.} \rangle$
$B \neq \langle \text{сл. знач.} \rangle$	$B = \langle \text{сл. знач.} \rangle$	$B = \langle \text{сл. знач.} \rangle$	$B \neq \langle \text{сл. знач.} \rangle$
Переносим в A значение из B .	Переносим в B значение из A .	Выделяем идентификатор связи между объектами, отличный от служебного значения и от уже существующих связей, и заносим его в поля A и B .	Если $A \neq B$, то пусть $C = \min(A, B)$, $D = \max(A, B)$. Для любых объектов в базе данных со значением D в поле c заменяем это значение на C .

Таблица 3.1. Таблица алгоритма назначения идентификаторов связи.

Для корректной работы алгоритма требуется, чтобы во время выполнения каждой из этих четырёх веток другие обработчики не могли выполнять аналогичные операции в базе данных.

Таким образом, имея в наличии идентификатор u , возможно построить запрос к базе данных, который выводит список физических объектов, а не список изображений объектов. Последний список может быть значительно длиннее, чем список объектов в поле зрения камеры, особенно в случае, когда несколько мастер-камер наблюдают за общей зоной обзора, в которой есть оживлённое движение.

3.6. Теоретическая оценка погрешности сопряжения видеокамер на основе достоверности сопоставления семантических элементов изображений

Для оценки погрешностей сопряжения видеокамер может быть использован статистический подход. На основе многократных повторов сопоставления изображений на поворотной и неподвижной видеокамерах или на паре неподвижных видеокамер, предполагая нормальное распределение отклонений фактических значений преобразований сопряжения от теоретических расчётных значений, можно оценить текущий уровень значимости с помощью простого подсчёта соотношения «промахов» и

«попаданий», при этом необходимо определиться, какое именно отклонение в абсолютном выражении соответствует «промаху». Общепринятое определение погрешности — это тройная дисперсия распределения отклонений фактического значения от расчётного, поэтому, зная значение, соответствующее разделению промахов и попаданий, и уровень значимости, который этому значению соответствует, мы сможем определить дисперсию, и, следовательно, погрешность.

Рассмотрим конкретные примеры определения погрешности при помощи статистического подхода.

Погрешность сопряжения поворотной и неподвижной видеокамер. Предположим, наведение производится на движущегося человека на расстоянии 20 метров от объектива камеры (см. рис. 3.11). «Промахом» мы будем считать такое наведение поворотной камеры на объект, что человек после завершения движения поворотной камеры оказывается вне границ изображения, т.е. его не видно. Если наведение производить с таким увеличением, что изображение человека в кадре примерно во весь рост, то вертикальный угол зрения слева равен $2/20\text{рад} \approx 6^\circ$, что соответствует горизонтальному углу $\frac{4}{3} \cdot 6^\circ = 8^\circ$. Считая, что горизонтальный угловой размер человека на расстоянии 20 метров равен примерно 1° , получаем, что горизонтальному промаху $\Delta\alpha$ (см. обозначения п. 3.4) соответствует примерное значение $8^\circ/2 + 1^\circ = 5^\circ$.

Пусть вертикальное наведение по углу склонения β производится без «промахов». Тогда, если общая доля промахов среди всех попыток наведения есть p , то, обозначив функцию ошибок как $F^{-1}(p)$, где

$$F(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^t e^{-\frac{1}{2}x^2} dx \text{ — функция распределения } N(0, 1),$$

получим с учётом $F(t) = F(-t)$ и $F(+\infty) = 1$, что

$$\begin{aligned} p &= P(|\Delta\alpha| > 5^\circ) = P(\Delta\alpha < -5^\circ) + P(\Delta\alpha > 5^\circ) = \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-5^\circ/\sigma} e^{-\frac{1}{2}x^2} dx + \frac{1}{\sqrt{2\pi}} \int_{5^\circ/\sigma}^{+\infty} e^{-\frac{1}{2}x^2} dx = 2 \frac{1}{\sqrt{2\pi}} \int_{5^\circ/\sigma}^{+\infty} e^{-\frac{1}{2}x^2} dx = 2 - 2F\left(\frac{5^\circ}{\sigma}\right), \end{aligned}$$

где σ — дисперсия отклонений $\Delta\alpha$. Отсюда получаем, что

$$\sigma = \frac{5^\circ}{F^{-1}(1 - p/2)},$$

т.е. оценкой погрешности является величина тройной дисперсии

$$|\Delta\alpha| = 3 \cdot \frac{5^\circ}{F^{-1}(1 - p/2)}. \quad (3.31)$$

Чтобы воспользоваться формулой (3.31), требуется провести значительный объём испытаний, чтобы набрать статистику «промахов» p .

Погрешность сопряжения неподвижных видеокамер. Проведя такую же последовательность рассуждений, как и для случая статистической оценки погрешности сопряжения поворотных и неподвижных видеокамер, получим формулу оценки погрешности горизонтальной координаты при преобразовании сопряжения:

$$|\Delta x| = 3 \cdot \frac{\Delta}{F^{-1}(1 - p/2)}, \quad (3.32)$$

где Δ — количество пикселей, которые соответствуют «промаху», а p — доля «промахов» от общего числа сравнений объектов при сопоставлении изображения одного и того же объекта на разных камерах, наблюдающих частично общую зону обзора.

Если объекты — это люди на расстоянии 20 метров от камеры, дальний план наблюдения которой находится на расстоянии 100 метров от камеры, а высота закрепления камеры над поверхностью 5 метров (см. рис. 3.11), то для обычного человека можно оценить «ширину» его изображения как

$$\frac{0,4}{20} \cdot \left(30^\circ \cdot \frac{\pi}{180^\circ}\right)^{-1} \cdot 352 \approx 15 \text{ пиксел},$$

где горизонтальный угловой размер человека $\frac{0,4}{20}$ поделен на горизонтальный угол зрения камеры 30° , и умножен на ширину изображения в пикселях 352. Исходя из ширины изображения объекта положим $\Delta = 15/2 \approx 8$. Количество «промахов» p , выраженное в долях от общего количества сопоставлений изображений, строго говоря, должно определяться по количеству попыток сопоставления на каждом кадре потока изображений. Однако для оценки погрешности по порядку величины можно взять просто долю таких объектов в общей зоне наблюдения, изображения которых не были сопоставлены системой друг другу по критерию (3.29).

Достоверность сопоставления изображений $1 - p$ является одной из наиболее важных характеристик аналитических видеосистем, которые могут быть объявлены производителями системы, хотя автор и не встречал систем с подобными характеристиками. Поэтому формулы для вычисления погрешностей сопряжения (3.31) и (3.32) позволяют оценить погрешность на основе известных технических характеристик, без проведения сложных измерений углов наведения и отклонений преобразований изображений.

3.7. Адаптация алгоритмов семантической обработки изображений для систем и приборов видеонаблюдения

Классификация и определение семантических признаков выделенных объектов — основная отличительная особенность аналитических видеосистем, которая позволяет

выполнять операции семантического сжатия. В работе реализованы следующие типы классификации:

- 1) классификация выделенного движущегося объекта по цвету и типам: человек, транспортное средство, группа людей;
- 2) классификация выделенного объекта по категориям: остановившийся, оставленный, обычный;
- 3) сопоставление изображений с разных камер одногого и того же объекта (изображение объекта с одной камеры классифицируется как похожее или отличное от изображения объектов с другой камеры);
- 4) классификация крупного снимка с поворотной камеры по категориям: снимок содержит интересующий нас объект, объект на снимке отсутствует из-за промаха при наведении;
- 5) классификация типа кресла в кинозале (пустое, занятое человеком, занятое вещами) для специальной аналитической видеосистемы, применяющейся для контроля зрителей в кинотеатре.

Рассмотрим опробованные в диссертационной работе методы распознавания и результаты классификации, достигнутые на данный момент.

Триангуляция[149]. Триангуляция — это метод распознавания категорий изображений объектов, введённый в данной диссертационной работе для случая, когда размерность пространства признаков равна двум. Для описания данного метода классификации применяется терминология, введенная в п. 1.4. На этапе обучения проводится триангуляция Делоне для точек в признаковом пространстве, соответствующих обучающему множеству L . На этапе распознавания алгоритм $A(I_0, I(S))$ находит треугольник триангуляции, в который попадает точка $I(S)$, и если все вершины найденного треугольника принадлежат классу K_i , то алгоритм выдает вектор, соответствующий истинности утверждения $S \in K_i$ ($\alpha_j^A(S) = \delta_{ij}$ для всех $j \in \overline{1, c}$), иначе $A(I_0, I(S)) = (\Delta, \dots, \Delta)$.

Простые методы триангуляции и поиска треугольника в триангуляции требуют выполнения $O(l^2)$ операций на этапе обучения и $O(l)$ операций при распознавании. Для ускорения этих процедур, во-первых, при триангуляции использовался алгоритм построения диаграммы Вороного, выполняемый за $O(l \log_2 l)$ операций (см. описание алгоритма в [138]), во-вторых, при поиске использовался алгоритм, подобный алгоритму поиска ближайшего соседа при помощи kD -дерева. В среднем реализованный алгоритм поиска требует $O(\log_2 l)$ операций.

Реализация поиска состоит из двух этапов. При обучении строится бинарное дерево, вершинами которого являются подмножества треугольников триангуляции. При

построении сначала в корень дерева заносятся все треугольники триангуляции. Затем в правую ветвь переносятся треугольники, у которых хоть одна вершина по координате x больше медианы всех координат x точек триангуляции, в левую ветвь записываются треугольники, у которых хотя бы одна вершина по координате x меньше медианы. Множества треугольников, находящихся на одном уровне дерева, таким образом, могут пересекаться. На следующем шаге вершины дерева разделяются по такому же принципу, только используются координаты y , а не x . Далее треугольники рекурсивно разделяются до тех пор, пока либо останется только один треугольник в вершине, либо разность множеств одного из наследников и родителя является пустым множеством, т.е. разделения не произошло. При поиске треугольника, в который попадает исходная точка, координаты точки соотносятся с медианами, которые делили треугольники дерева при переходе от родителей к наследникам. За $O(\log_2 l)$ происходит спуск к «листу», в котором вершины треугольников расположены также по отношению к медианам, как и у исходной точке. Далее происходит проверка, принадлежит ли точка одному из треугольников «листа». Если да, то поиск завершён, если нет, то необходимо подняться на уровень выше и продолжить поиск. В редких случаях возможна ситуация, когда придётся вернуться к корню дерева.

В целом алгоритм классификации при помощи триангуляции аналогичен алгоритму классификации по правилу k ближайших соседей. Тем не менее, результаты получаются разные, и иногда триангуляция срабатывает надёжнее. Применение триангуляции обосновано тем, что для неё легко спрогнозировать результаты классификации, опираясь только на визуальное представление векторов-признаков на плоскости. В случае применения нейросетевого подхода или любого другого метода распознавания, описанного в п. 1.4, конечный результат можно получить только после эксперимента. Однако ограничение 2, накладываемое на размерность пространства признаков, существенно ограничивает возможность использования триангуляции как подхода к распознаванию объектов.

Наборы признаков и способы их перебора. Перечислим сначала признаки объектов, которые выделяются на мастер-камерах. Каждый такой объект представляет ограниченное множество точек Ω на изображении с целочисленными координатами (x, y) . Введём обозначения

$$x_{min} = \arg \min_{x \in \Omega} x,$$

$$y_{min} = \arg \min_{y \in \Omega} y,$$

$$x_{max} = \arg \max_{x \in \Omega} x,$$

$$y_{max} = \arg \max_{y \in \Omega} y,$$

тогда можно ввести «ширину» объекта $w = x_{max} - x_{min} + 1$ и «высоту» объекта $h = y_{max} - y_{min} + 1$.

Пусть $\omega(x, y)$ — характеристическая функция множества Ω , т.е.

$$\omega(x, y) = \begin{cases} 1, & (x, y) \in \Omega, \\ 0, & (x, y) \notin \Omega. \end{cases}$$

Эту функцию для каждого объекта находят алгоритмы выделения движения, упомянутые в пункте 1.3. Носитель функции $\omega(x, y)$ (множество точек с отличными от нуля значениями) содержится во множестве $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$ (декартово произведение двух отрезков), что используется в перечисленных ниже формулах.

В качестве признаков выделенных на мастер-камерах объектов используются:

- 1) отношение $\frac{h}{w}$;
- 2) отношение площади объекта $S = \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)$ к площади прямоугольника $[x_{min}, x_{max}] \times [y_{min}, y_{max}]$, равное $\frac{S}{hw}$;
- 3) центр масс (Mx, My) , вычисляемый по формулам

$$Mx = \frac{1}{wS} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)x,$$

$$My = \frac{1}{hS} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)y;$$

- 4) вторые моменты

$$Mx^2 = \frac{1}{w^2 S} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)(x - wMx)^2,$$

$$My^2 = \frac{1}{h^2 S} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)(y - hMy)^2,$$

$$Mxy = \frac{1}{hwS} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)(x - wMx)(y - hMy);$$

- 5) третьи моменты

$$Mx^3 = \frac{1}{w^3 S} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)|x - wMx|^3,$$

$$My^3 = \frac{1}{h^3 S} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)|y - hMy|^3,$$

$$Mz^3 = \frac{1}{S^{5/2}} \sum_{x=x_{min}}^{x_{max}} \sum_{y=y_{min}}^{y_{max}} \omega(x, y)((x - wMx)^2 + (y - hMy)^2)^{3/2};$$

- 6) несколько функций от определённых выше величин: $\frac{Mx}{My}, \frac{(wMx)^2}{S}, \frac{(hMy)^2}{S}$;
- 7) отношение H разности максимума и минимума функции $H(x) = \sum_{y=y_{min}}^{y_{max}} \omega(x, y)$ на отрезке $[x_{min}, x_{max}]$ к высоте объекта h ;
- 8) время t_s , в течение которого объект не двигался к текущему моменту;
- 9) флаг разделения объектов f_s , равный единице, если во время появления объекта рядом с ним находился другой объект, в противном случае равный нулю;
- 10) матрица-шаблон силуэта размерности $N \times N$, элементы T_{nm} которой рассчитываются по формуле

$$T_{nm} = \frac{1}{whN^2} \sum_{i=nw}^{(n+1)w-1} \sum_{j=mh}^{(m+1)h-1} \omega(x_{min} + \left\lfloor \frac{i}{N} \right\rfloor, y_{min} + \left\lfloor \frac{j}{N} \right\rfloor).$$

Если пользоваться последней формулой буквально, то сложность вычисления матрицы-шаблона будет $O(whN^2)$, однако в данной работе используется алгоритм со сложностью $O((w + N)(h + N))$. Суть алгоритма в том, что суммирование ведется не по каждой точке прямоугольника $[0, wN - 1] \times [0, hN - 1]$, а только по клеткам неравномерной сетки этого прямоугольника, узлами которой служат точки, вертикальные координаты которых кратны h или N , а горизонтальные координаты кратны w или N . Внутри каждой клетки такой сетки функция $\omega(x_{min} + \lfloor i/N \rfloor, y_{min} + \lfloor j/N \rfloor)$ тождественно равна либо нулю либо единице, так что в цикле по клеткам сетки к соответствующему элементу прибавляется либо 0, либо площадь клетки.

В работе системы контроля зрителей [206] для каждого кресла ведётся подсчёт двух признаков:

- 1) корреляция r_1 между текущим изображением кресла и изображением пустого кресла, заготовленным заранее;
- 2) корреляция r_2 между текущим изображением кресла и изображением кресла, сглаженного в течении нескольких десятков секунд (параметр оценивает «подвижность» изображения внутри кресла).

Чтобы распознать тип снимка со слейв-камеры, используются следующие признаки:

- 1) координаты $x'_{min}, x'_{max}, y'_{min}, y'_{max}$, преобразованные из соответствующих координат объекта на мастере в координаты слейва (доступны только в случае, когда после поворота слейва объект ещё не потерян на мастере);
- 2) отношение Λ длин контура картинки, «почти параллельных» другим линиям контура, к общей длине контура на картинке.

Для выбора наилучшего набора параметров при распознавании изображений объектов, выделенных на мастере, был произведён перебор методов распознавания и наборов параметров. После первых шагов выяснилось, что нейросетевой алгоритм при фиксированном наборе параметров в большинстве случаев даёт наибольшую долю верных ответов. Однако обучение нейросети слишком долгий процесс, чтобы можно было провести полный перебор наборов параметров. В дальнейшем планируется применить генетический алгоритм для этой задачи, однако на данный момент произведён перебор только «наиболее перспективных» наборов параметров, перспективность определялась, либо с помощью метода линейного дискриминантного анализа, либо умозрительно. Тем не менее, опробовано несколько сотен наборов, что не так уж и мало для того, чтобы надеяться на близость полученного результата к оптимальному.

Выводы

1. Проведён анализ различных схем синхронизации программных модулей семантической обработки сигналов от видеоприборов. При помощи анализа сложности алгоритмов и численных методов оптимизации автором предложен наиболее эффективный из рассмотренных алгоритмов синхронизации, основанный на обработке в цикле приёма сообщений от модулей поворотных и неподвижных камер.
2. Выведены соотношения для параметров управления поворотными видеокамерами при наведении на движущиеся объекты для различных моделей управления поворотными видеокамерами.
3. Исследованы составляющие погрешности при наведении на движущийся объект. Получены численные оценки погрешности, которые в рассмотренных условиях наблюдения составляют $0,5^\circ$ при наведении на неподвижный объект и 20° при наведении на движущийся объект для наихудшей оценки предсказанного положения объекта.
4. Установлено, что при погрешности наведения поворотной камеры 20° и при угле зрения сопряжённой неподвижной камеры 30° увеличение изображений объектов более, чем в 1,5 раза невозможно, если требуется высокая достоверность сопоставления изображений.
5. Разработан критерий-условие идентичности объектов (формула (3.29)), если изображения объектов получены несколькими видеокамерами с общей зоной обзора. Предложен алгоритм объединения записей, соответствующих одному физическому объекту, в базе данных.
6. С помощью модели физических перемещений объектов проанализированы ограничения для применения критерия идентичности: для надёжного сопоставления

скорость человека должна быть по модулю менее 15км/ч, скорость автомобиля не должна превышать по модулю 90км/ч при любых направлениях движения объектов по отношению к оптической оси камеры. В других условиях критерий применим, но верное сопоставление менее вероятно.

7. Разработан способ оценки погрешностей сопряжения на основе статистики сопоставления изображений во время работы системы. Установлена связь достоверности сопряжения и погрешности сопряжения.

8. Исследованы способы применения алгоритмов семантической классификации изображений по различным категориям в аналитических видеосистемах. Впервые предложено использовать триангуляцию при классификации объектов по двум признакам.

4. Разработка, исследование и анализ методик сравнения алгоритмов семантического сжатия видеинформации

Проблема проверки эффективности алгоритмов семантического сжатия относится к области тестирования программного обеспечения, которое призвано обрабатывать большие массивы видеинформации. В области компьютерного зрения прослеживается определённая тенденция: чем больший объём видеинформации задействован при тестировании программных компонент аналитической видеосистемы и чем качественнее автоматизирован процесс проверки, тем более эффективно работают конечные программные модули.

В данной главе сначала будет приведён обзор текущих методов оценки качества семантической обработки, а потом будут представлены новые методики оценки качества алгоритмов семантического сжатия, которые позволяют существенно улучшить достоверность выделения событий методами компьютерного зрения.

4.1. Сравнение алгоритмов семантической сегментации изображений

Основная задача семантической сегментации изображения заключается в выделении отдельных элементов на изображении и классификация данных элементов по категориям (см. рис. 1.5). Отдельным элементами изображения могут быть автомобили, люди, стационарные объекты на сцене и изображения других категорий объектов. Выделить изображение объекта семантически — значит определить множество точек A изображения, из которых состоит изображение объекта в обрабатываемом кадре.

Типичным подходом к оценки качества алгоритма сегментации является сравнение результата с шаблонной сегментацией [122], которая определяется либо вручную оператором, либо с помощью другого точного алгоритма. Будем обозначать множество точек шаблонной сегментации объекта как T . Обозначим $|\mathcal{M}|$ мощность (количество элементов) множества \mathcal{M} . Распространённым показателем качества сегментации изображения является отношение

$$\varrho(A, T) = \frac{|A \cap T|}{|A \cup T|}. \quad (4.1)$$

Графическая иллюстрация к вычислению меры $\varrho(A, T)$ представлены на рис. 4.1. Очевидно, что

$$\forall A, T : \varrho(A, T) \leq 1,$$

а также

$$\forall A, T : \varrho(A, T) = 1 \text{ равносильно } A = T,$$

то есть чем больше результат A повторяет форму шаблона T , тем ближе значение ϱ к единице.



Рисунок 4.1. Иллюстрация к вычислению меры качества сегментации $\varrho(A, T)$.

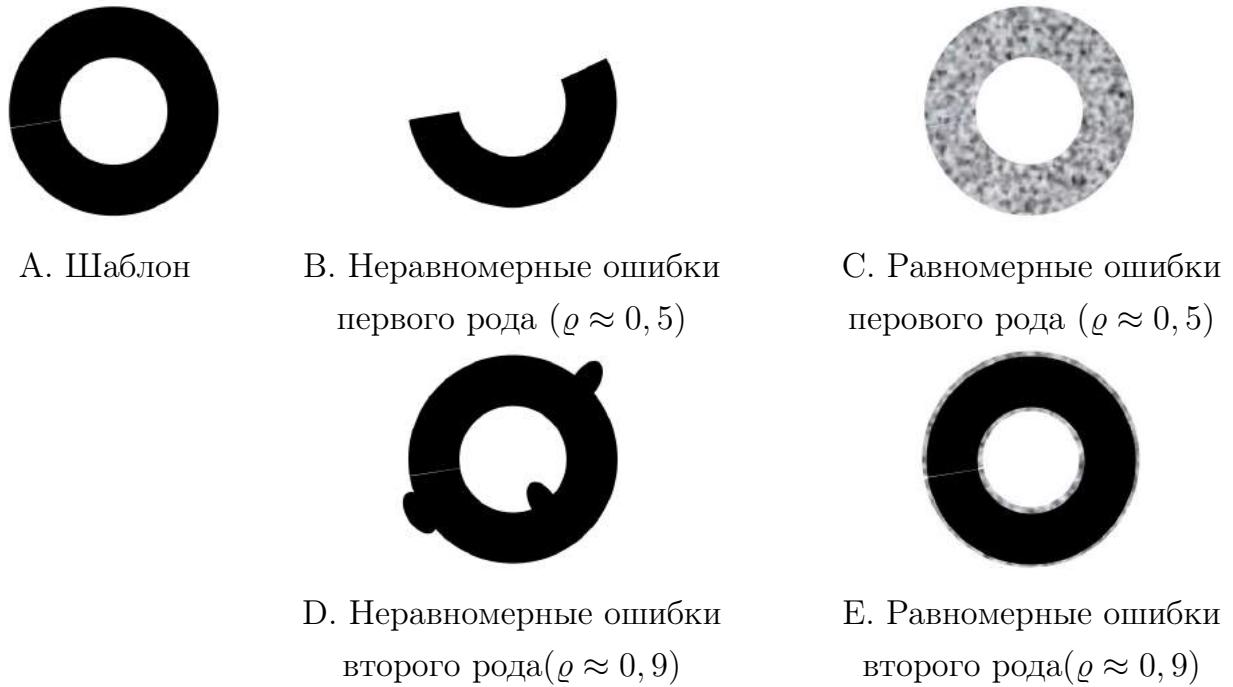


Рисунок 4.2. Примеры некорректной оценки качества выделения шаблона с помощью меры $\varrho(A, T)$

В работе [150] выявлены недостатки приведённой выше оценки качества алгоритма семантической сегментации. Проиллюстрируем данные недостатки примерами на рисунке 4.2.

На рисунке 4.2.А представлен пример множества точек T , соответствующих шаблонному выделению изображения кольца. Рисунок 4.2.В представляет собой пример множества точек A_B , которое является результатом работы алгоритма, выделившего лишь половину кольца T , а на рисунке 4.2.С изображено множество точек A_C как результат работы алгоритма, выделившего не все точки кольца, а только, допустим, половину точек, случайно и равномерно распределённых по площади множества T . На рисунках 4.2.Д и 4.2.Е представлены результаты работы алгоритмов как множества точек A_D и A_E , соответственно. В данных множествах выделены все точки кольца T , однако дополнительно в конечный результат ошибочно включены некоторые артефакты в случае множества A_D и шумовые точки вдоль границ кольца в случае

множества A_E , причём мы будем считать, что площадь артефактов на рисунке 4.2.D соответствует числу шумовых точек на рисунке 4.2.E и равна примерно девятой части площади кольца T .

С точки зрения меры качества ϱ результат A_B эквивалентен результату A_C , поскольку мы ранее определили, что $A_B \cup T = T$, $A_B \cap T = A_B$, $A_C \cup T = T$, $A_C \cap T = A_C$ и $|A_B| \approx |A_C| \approx 0,5|T|$. Следовательно, $\varrho(A_B, T) \approx \varrho(A_C, T) \approx 0,5$ по формуле (4.1).

Аналогично, результат A_D обладает одинаковой характеристикой качества совпадения ϱ с результатом A_E , поскольку по нашему определению $A_D \cap T = T$, $A_E \cap T = T$, $|A_D \cup T| \approx |T| + 1/9|T| = 10/9|T|$ и $|A_D \cup T| \approx 10/9|T|$. Подставив эти значения в формулу (4.1), получим $\varrho(A_D, T) \approx \varrho(A_E, T) \approx 0,9$.

Несмотря на одинаковую меру совпадения, визуально множества A_C и A_E лучше повторяют форму множества T , чем, соответственно, множества A_B и A_D . Более того, результаты A_C и A_E предпочтительней с точки зрения устойчивости алгоритмов, которые используют полученные множества A_\bullet как входные данные. Например, алгоритмы поиска окружностей на изображении, построенные на основе преобразования Хафа (один из таких алгоритмов описан в [151]) во многих случаях неустойчивы к неоднородным ошибкам первого и второго рода при отделении объектов переднего плана от фона.

В источнике [150] представлены идеи, как ввести меру совпадения ϱ^* шаблонного множества T и результирующего множества A таким образом, чтобы она лучше дискриминировала алгоритмы семантической сегментации. Т.е. для примеров, представленных на рис. 4.2 должно выполняться $\varrho^*(A_B, T) < \varrho(A_C, T)$ и $\varrho(A_D, T) < \varrho(A_E, T)$. Далее будет описан способ вычисления меры ϱ^* , несколько обобщённый по сравнению со способом из упомянутой работы, с привлечением теории нечётких множеств [152].

Детерминированному множеству T мы поставим в соответствие нечёткое множество T' , являющееся парами из точек изображения p и числовой меры принадлежности точки множеству в диапазоне $[0, 1]$. В качестве меры принадлежности к нечёткому множеству шаблонных точек T' мы введём бинарную функцию

$$\mu^{T'}(p) = \begin{cases} 1, & \text{если } p \in T, \\ 0 & \text{в противном случае.} \end{cases}$$

Таким образом подчёркивается, что принадлежность точек к шаблону определена точно. Для множества A , выдаваемого в результате алгоритма сегментации, мы определим соответствующее нечёткое множество A' , мера принадлежности каждой точки

изображения будет определяться на основании функции $\mu^{T'}$:

$$\mu^{A'}(p) = \begin{cases} 1, & \text{если } \mu^{T'}(p) = 1 \text{ и } p \in A, \\ g(p), & \text{если } \mu^{T'}(p) = 1 \text{ и } p \notin A, \\ b(p), & \text{если } \mu^{T'}(p) = 0 \text{ и } p \in A, \\ 0, & \text{если } \mu^{T'}(p) = 0 \text{ и } p \notin A. \end{cases}$$

Функция $g(p)$ определяется как функция сглаживания с весами распределения Гаусса:

$$g(p) = \sum_{q \in A} \frac{\mu^{T'}(q)}{\sqrt{2\pi\sigma^2}} e^{-\frac{(p-q)^2}{2\sigma^2}},$$

где под $(p - q)^2$ обозначено скалярное произведение вектора из разности координат точек самого на себя. Благодаря сглаживанию в A' с некоторым весом входят даже те точки, которые не вошли в результат алгоритма A , причём вес определяется величиной окрестности суммирования σ^2 и количеством точек из пересечения $A \cap T$, которые находятся близко к точке p — чем больше таких точек, тем выше вес. Принято использовать небольшие окрестности, например, $\sigma^2 = 5$.

Функция $b(p)$ определяет вес точек A' , находящихся вне T , в зависимости от того, насколько они удалены от множества T , например, следующим образом:

$$b(p) = \frac{2 - e^{\alpha d(T,p)}}{2},$$

где α — отрицательная константа, а $d(T, p)$ — расстояние от точки p до множества T , определяемое как

$$d(T, p) = \min_{q \in T} \|p - q\|.$$

Вместо экспоненциальной функции можно использовать любую монотонно возрастающую функцию $f(x)$ с областью значений $[0, 1]$ для того, чтобы определить функцию $b(p)$ как $f(d(T, p))$, однако экспоненциальная функция $f(x) = \frac{2-e^{\alpha x}}{2}$ удобна тем, что достаточно быстро возрастает до 1, и зону пониженного вклада в ошибку первого рода можно задать показателем α .

После того, как нечёткие множества T' и A' определены, меру совпадения исходных множеств T и A можно определить как

$$\varrho^*(A, T) = \frac{|A' \cap T'|}{|A' \cup T'|}, \quad (4.2)$$

где объединение, пересечение, и мощность множеств определяются в рамках теории нечётких множеств, т.е.

$$\begin{aligned} \mu^{A' \cap T'}(x) &= \min(\mu^{A'}(x), \mu^{T'}(x)), \\ \mu^{A' \cup T'}(x) &= \max(\mu^{A'}(x), \mu^{T'}(x)), \\ |M| &= \sum_{x \in M} \mu^M(x) \text{ для } M = A' \cap T' \text{ и } M = A' \cup T'. \end{aligned}$$

Ввиду того, что принадлежность точек множеству A' задана более приближенно к шаблону T в тех случаях, когда это не мешает субъективному восприятию результата A , мера совпадения ϱ^* более устойчива к небольшим шумовым искажениям в результатах алгоритма сегментации. Например, за счёт функции сглаживания $g(p)$ мера совпадения $\varrho^*(A_C, T)$ значительно меньше $\varrho^*(A_B, T)$, а за счёт уменьшенного веса $b(p)$ точек возле границ шаблона T мера совпадения $\varrho^*(A_E, T)$ значительно больше меры $\varrho^*(A_D)$ (см. рис. 4.2).

Как показывает практика (см. [150]), предложенная мера качества сегментации (4.2) не только лучше упорядочивает результаты в соответствии с субъективным представлением по сравнению с мерой (4.1), но и по объективным критериям оценивать качество с помощью меры ϱ^* предпочтительней, например, потому, что использование результатов сегментацией с более высокой мерой ϱ^* для последующих алгоритмов семантического анализа обычно приводит к более точным конечным результатам. Мера ϱ подобным свойством не обладает.

4.2. Сравнение алгоритмов выделения движущихся объектов

Выделение движущихся объектов — широко распространённый приём семантического сжатия видеопотока. Особенно часто анализ движущихся объектов производится в различных системах охранного видеонаблюдения, поскольку обычно при эксплуатации подобных систем интерес представляет не весь объём видеинформации, а только изображения с существенными изменениями на наблюдаемой сцене. Существуют методики оценки эффективности работы алгоритмов семантического сжатия, выделяющих движение объектов [147, 148, 153], которые мы рассмотрим в данном параграфе.

Вычитание фона — это категория алгоритмов обработки изображений, предназначенных для классификации объектов на последовательности изображений как подвижные и неподвижные. Чаще всего вычитание фона используется в детекторах движения на основе видеокамер в системах видеонаблюдения. Существует множество методов вычитания фона, данной проблеме посвящены тысячи публикаций, среди которых особо можно выделить работы [77, 154, 155, 156, 157], описывающие основные методы. Алгоритмы вычитания фона отличаются как по вычислительной сложности, так и по точности идентификации, поэтому возникает потребность в сравнении данных методов. В работе [123] предложено сравнивать методы вычитания фона с помощью тестов на фиксированных последовательностях изображений разного вида с заранее отмеченными точками движения в форме маски для каждого кадра последовательности. В качестве метрики сравнения используются различные показатели совпадения результатов обработки алгоритмов, классифицирующих точки кадра как движущиеся или неподвижные, с заранее введёнными шаблонами, считающимися эта-



Рисунок 4.3. Примеры изображений тестовой выборки changedetection.net (сверху) с результатом обработки одним из распространённых методов вычитания фона (снизу). Слева направо примеры изображений из категорий: основной набор, динамический фон, тряска камеры, тени объектов, перемещение элементов фона, тепловизионные изображения.

лонами.

Изображения тестовой выборки changedetection.org состоят из нескольких десятков видеороликов, в которых содержится в общей сложности более 100 тысяч кадров изображений. Видеоролики разделены на различные категории, среди которых: базовый тестовый набор, изображения с динамически изменяющимся фоном (водоёмы, ветви кустарника и деревьев на ветре и т.п.), изображения с не закреплёнными жёстко трясущихся камер, изображения с тенями, тесты на обработку изменений фона благодаря перемещаемым на изображениях предметам и изображения тепловизионных камер. Примеры изображений из разных категорий приведены на рис. 4.3 (рисунок заимствован из работы [123]).

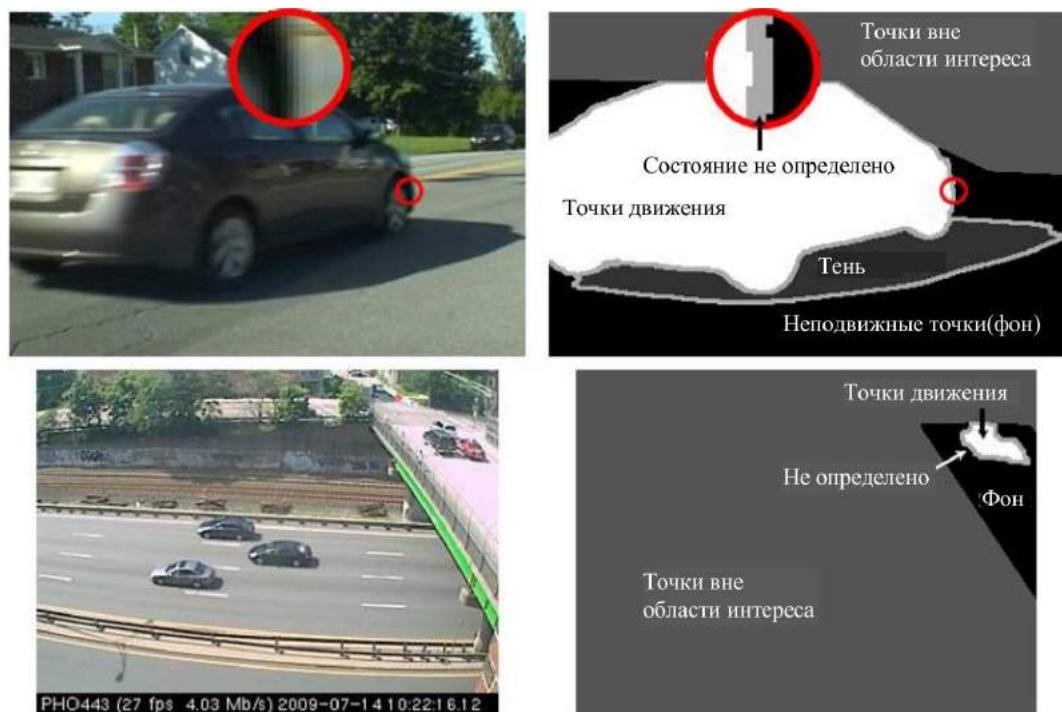


Рисунок 4.4. Пример разметки кадров из тестового набора.

В каждой точке все изображения тестового набора размечаются, то есть каждая точка может принадлежать одному из пяти классов: подвижная точка, неподвижная точка, состояние точки неопределено, точка находится вне области интереса, и точка неподвижна, но находится в тени от движущегося объекта. Если обозначить эти состояния, соответственно, как $C = \{m, s, u, o, h\}$, то разметкой является характеристическая функция $\chi_T(x, y)$, которая каждой точке изображения (x, y) сопоставляет её класс C . Пример разметки, заимствованный из [1] приведён на рис. 4.3. Состояние o (вне области интереса) введено для того, чтобы лучше разграничить категории тестовых наборов изображений. Состояние u (не определено) введено для устранения неоднозначности разметки в граничных точках изображений движущихся объектов: в граничных точках любому оператору, который проводит разметку, сложно однозначно отнести точку к классу движущаяся(m) или неподвижная(s). Наконец, состояние h (тень подвижного объекта) специально введено для отдельной метрики, позволяющей оценить возможность метода вычитания фона классифицировать именно такие точки как неподвижные точки фона. Тестируемые алгоритмы вычитания фона выдают результат в форме индикаторной функции $\chi_A(x, y)$ для каждой точки каждого кадра, в отличие от функции χ_T , данная функция может принимать только два значения: m (точка движения) или s (неподвижная точка). Примеры данной функции для одного из наиболее простых алгоритмов вычитания фона приведены на рис. 4.3.

$$TP = \{(x, y) | \chi_T(x, y) = m \text{ при условии, что } \chi_A(x, y) = m\}.$$

Точки, корректно определённые алгоритмом, как неподвижные, обозначаются

$$TN = \{(x, y) | \chi_T(x, y) = s \text{ или } \chi_T(x, y) = h \text{ при условии, что } \chi_A(x, y) = s\}.$$

Множество точек, в которых алгоритм допустил ошибку первого рода, обозначается как

$$FP = \{(x, y) | \chi_T(x, y) = s \text{ или } \chi_T(x, y) = h \text{ при условии, что } \chi_A(x, y) = m\}.$$

Наконец, множество точек с ошибкой второго рода в результатах обозначается

$$FN = \{(x, y) | \chi_T(x, y) = m \text{ при условии, что } \chi_A(x, y) = s\}.$$

Далее для краткости под TP , TN , FP , FN подразумевается мощность однократных множеств. В рамках changedetection.net предлагается использовать следующие метрики для одной видеопоследовательности кадров в каждой из категорий.

1. $Re = TP / (TP + FN)$ — доля точек движения в шаблоне разметки, корректно определённых алгоритмом как движущиеся.

2. $Sp = TN / (TN + FP)$ — доля точек фона в шаблоне разметки, корректно определённых алгоритмом как неподвижные.
3. $FPR = FP / (FP + TN)$ — доля точек движения по результатам алгоритмов, ошибочно определённых движущимися точками.
4. $FNR = FN / (TN + FP)$ — доля точек фона по результатам алгоритмов, ошибочно определённых как неподвижные точки.
5. $PWC = (FN + FP) / (TP + FN + FP + TN)$ — доля ошибок классификации состояния движения точек в ответах алгоритмов вычитания фона.
6. $Pr = TP / (TP + FP)$ — доля точек движения по результатам алгоритмов, правильно определённых подвижными.
7. $Fm = 2 \frac{Pr \cdot Re}{Pr + Re}$ — среднее гармоническое показателей Re и Pr .

Значения Re , Sp , FNR характеризуют ошибки первого рода, показатели FPR , Pr характеризуют ошибки второго рода, и показатели PWC и $Fmeasure$ тем или иным способом оценивают среднюю ошибку.

Для усреднения этих метрик по всем видеороликам используются следующие правила. Обозначим значение одной из 7 метрик, приведённых выше, в категории c для видеоролика v как $M_{c,i}^v$ (расчёт данного показателя ведётся на основе величин TP , TN , FP , FN , на всех кадрах видеоролика, i — индекс метрики), а количество видеороликов в категории c как N_c . Значение метрик в рамках одной категории рассчитывается как среднее значение метрики по видеороликам:

$$M_{c,i} = \frac{1}{N_c} \sum_v M_{c,i}^v.$$

Значение показателя по всем категориям видеороликов вычисляется с помощью ещё одного усреднения

$$M_i = \frac{1}{6} \sum_c M_{c,i}. \quad (4.3)$$

С помощью операции ранжирования вводится два различных обобщённых показателя качества работы сравниваемых алгоритмов вычитания фона. Операция ранжирования — это операция расстановки по порядку методов по одной из введённых ранее метрик в порядке убывания качества, рангом метода называется порядковый номер метода в данной расстановке. Если метрики у двух и более методов совпадают, то при ранжировании им всем назначается среднее арифметическое разделённых методами позиций. Если в сравнении участвует Ω методов, то для каждого m -го метода из числа Ω можно определить его ранг по метрике $M_{c,i}$ в каждой категории последовательности изображений как $rank_{M_{c,i}}(m)$. Усреднённая позиция методов в каждой

категории с учётом всех метрик рассчитывается по формуле

$$RM_{c,m} = \frac{1}{7} \sum_i rank_{M_{c,i}}(m),$$

Альтернативной обобщённой метрикой сравнения методов вычитания фона служит средний ранг при упорядочивании методов по средним метрикам, рассчитанным по формуле (4.3):

$$R_m = \frac{1}{7} \sum_i rank_{M_i}(m).$$

В таблице 4.2 приведены выборочные результаты сравнения 16 различных методов, приведённых в работе [123]. Данные методы соответствуют, во-первых, наиболее известным методам вычитания фона, а во-вторых, наилучшим методам по результатам, указанным в рассматриваемой работе.

Метод	RC	R	Re	Sp	FPR	FNR	PWC	Fm	Pr
PBAS[156]	3,00	3,29	0,79	0,990	0,010	0,009	1,77	0,75	0,82
ViBe+[154]	4,83	5,00	0,69	0,993	0,007	0,017	2,18	0,72	0,83
GMM[77]	11,50	10,14	0,71	0,986	0,014	0,027	3,68	0,52	0,72
Euclidean distance	16,67	14,00	0,70	0,969	0,031	0,017	4,35	0,61	0,62

Таблица 4.2. Результаты сравнения методов вычитания фона, приведённые в работе [123].

Из таблицы 4.2 следуют следующие выводы. Во-первых, наиболее простые методы вычитания фона, основанные на сравнении средней интенсивности в точке с текущей интенсивностью, обладают наименьшей устойчивостью к ошибкам первого и второго рода. Во-вторых, наиболее известный метод вычитания фона, основанный на анализе распределения интенсивности в точке в предположении, что это распределение является суммой распределений нормальных величин (подробнее см. в работе [77]), занимает средние позиции в сравнении с остальными методами, рассмотренными в работе [123]. Также следует отметить тенденцию, что в целом методы вычитания фона, использующие непараметрический подход для оценки распределения интенсивности точек переднего и заднего плана, т.е. движения и фона, показывают лучшие результаты по сравнению с методами, которые используют параметрический подход для оценки аналогичных распределений интенсивности.

Информация о том, что в кадре есть движение, и что оно происходит в конкретных точках изображения, является слишком большой по объёму, чтобы представлять интерес с точки зрения систем так называемого семантического сжатия, или семантической обработки изображений. Обычно, эту информацию подвергают дальнейшей

обработке, выделяя из полученных областей движения отдельные движущиеся объекты, одновременно происходит восстановление траектории движения объектов в кадре. Информация о масках движения на каждом кадре по объёму составляет сотни тысяч байт, информация о траекториях объектов составляет уже порядка тысячи байт, причём не на одном кадре, а на сотнях кадров. В описание траектории объекта может входить либо координаты ограничивающей изображение объекта прямоугольной рамки, либо координаты ограничивающего объект многоугольника на всех кадрах последовательности, на которых объект присутствовал.

Наиболее известный алгоритм выделения траекторий объектов на основе метода вычитания фона представлен в статье [38], а также в параграфе 1.3, рисунок 1.9. В результате работы данного алгоритма мы получаем траектории объектов движения. Возможны различные ошибки в работе данного алгоритма: могут определяться ложные движущиеся объекты в тех местах, где движения нет, могут быть пропущены объекты движения, возможны ошибки в решении задачи о назначении, когда траектория одного объекта переходит в результатах алгоритма на траекторию другого объекта. В зависимости от того, какой метод вычитания фона применяется на первом этапе, данные ошибки будут различаться. Поскольку в данной работе мы оцениваем влияние вычитания фона на конечный результат в виде описания траекторий объектов, необходимо ввести определения, позволяющие вычислять метрики, характеризующие качество полученного результата.

Алгоритмы выделения движущихся объектов принято проверять с помощью записанного видеоматериала, у которого есть так называемая аннотация или разметка. Аннотация заключается в том, что перед проверкой работы алгоритмов на некоторой видеопоследовательности каждый кадр вручную размечается. Разметка может содержать области движения, и их связь с областями движения на предыдущем кадре. Область движения обычно размечается с помощью минимального ограничивающего прямоугольника со сторонами, параллельными границам кадра, поскольку ввести такой прямоугольник достаточно просто с помощью современных устройств ввода. Есть связь с областями движения на предыдущих кадрах, с помощью разметки всех кадров можно восстановить траектории движения объектов во времени. Обладая подобной аннотацией для конкретной видеопоследовательности, можно сопоставить её с аннотацией, которые выдают различные алгоритмы при обработке этой видеопоследовательности. Разметка, или аннотация, сделанная вручную, является эталоном для работы алгоритмов. Поэтому мы будем называть её иногда «идеальной разметкой» или «идеальной аннотацией», имея в виду, что ожидается совпадение результатов работы алгоритмов с идеальной разметкой. Существуют системы аннотации видео в сети Интернет [158], ими можно свободно пользоваться в качестве web-приложений.

Рассмотрим основные показатели качества работы алгоритмов выделения движущихся объектов и их траекторий. Для этого введём обозначения:

- $V^n = \{V_i^n\}$ для последовательности кадров видео, где i — номер кадра, n — номер видеопоследовательности в тестовой базе;
- $\{I_l^n\}$ для объектов идеальной разметки, где l — номер идеального объекта, n — номер видеопоследовательности, на которой данный объект отмечен;
- $\{A_m^{an}\}$ для объектов автоматической разметки, где m — номер объекта, n — номер видеопоследовательности, a — номер тестируемого алгоритма;
- $P_{il}^n = P(V_i^n, I_l^n)$ и $P_{im}^{an} = P(V_i^n, A_m^{an})$ — функции, возвращающие положение объектов I_l^n и A_m^{an} на кадре V_i^n , где положение P_{il}^n и P_{im}^{an} — это множество точек объектов в кадре, которое, в связи с трудоёмкостью поточечной разметки изображений объектов, обычно представляется в виде прямоугольника наименьшего размера с параллельными границами кадра сторонами и содержащего все точки изображения объекта;
- $\rho(P_1, P_2)$ — мера совпадения двух областей кадра P_1 и P_2 , далее мы будем использовать $\rho(P_1, P_2) = \frac{|P_1 \cap P_2|}{|P_1 \cup P_2|}$ (отношение мощности пересечения множеств к мощности объединения, которое изменяется в пределах от 0 для непересекающихся множеств до 1 для совпадающих множеств);
- $\rho(P_{il}^n, P_{im}^{an}) = \rho(P_{im}^{an}, P_{il}^n) = \rho_{lm}^{ian}$ — мера совпадения положений автоматического объекта A_m^{an} и идеального объекта I_l^n на кадре i последовательности V^n ;
- $\mu(\omega)$ — это число элементов в конечном множестве ω , т.е. мощность;
- $\mu(A_m^{an}) = \mu(\{i | P_{im}^{an} \neq \emptyset\})$ — число кадров, на которых есть объект A_m^{an} ;
- $\mu(I_l^n) = \mu(\{i | P_{il}^n \neq \emptyset\})$ — число кадров, на которых есть объект I_l^n ;
- $\chi(A_m^{an}), \chi(I_l^n)$ — функция, результат которой — некоторое свойство объектов A_m^{an} и I_l^n , например, категория объекта (человек, машина, группа людей).

Для сокращения обозначений далее будем рассматривать фиксированный алгоритм обработки и фиксированную видеопоследовательность, что позволит нам опустить индексы алгоритма a и видеопоследовательности n .

На основе введённых обозначений можно рассчитать следующие метрики работы алгоритмов.

Число ошибок первого рода можно рассчитать по формуле

$$F_P^{\alpha\beta} = \mu(\left\{ A_m \mid \forall l : \frac{\mu(\{i \mid \rho_{lm}^i < \alpha\})}{\mu(A_m)} > \beta \right\}), \quad (4.4)$$

где $\alpha \in (0, 1]$ — порог меры совпадения автоматического и идеального объекта на одном кадре для того, чтобы иметь основания эти объекты сопоставлять на данном кадре, а $\beta \in [0, 1]$ — порог на долю кадров автоматических объектов, на которых идеальный объект должен быть сопоставлен автоматическому для того, чтобы автоматический объект не считался ложным срабатыванием. Чем ближе пороги α и β к единице, тем строже критерий для отбора объектов, не являющихся ложными, чем ближе эти пороги к 0, тем критерий мягче. Например, при $\beta = 0$ и достаточно малых α , например, $\alpha = \alpha_m = \frac{1}{2} \rho_{lm}^i$, автоматический объект будет считаться ложным только тогда, когда он ни разу не пересекался по положению ни с каким из идеальных объектов.

Число ошибок второго рода можно рассчитать по формуле, подобной (4.4):

$$F_N^{\alpha\beta} = \mu(\left\{ I_l \mid \forall m : \frac{\mu(\{i \mid \rho_{lm}^i < \alpha\})}{\mu(I_l)} > \beta \right\}), \quad (4.5)$$

т.е. идеальный объект считается пропущенным, если не нашлось ни одного автоматического объекта, который совпадал бы по положению лучше, чем на α , на более, чем на $(1 - \beta)\mu(I_l)$ кадрах. При $\beta = 0$ и $\alpha = \alpha_m$ идеальный объект считается обнаруженным, если хотя бы на одном кадре его положение пересекается с положением одного из автоматических объектов.

Качество сопровождения объекта можно оценить с помощью показателей

$$Q_T^\alpha = \sum_l Q_{Tl}^\alpha, \text{ где } Q_{Tl}^\alpha = \mu(\{i \mid \exists m : \rho_{lm}^i \geq \alpha\}), \quad (4.6)$$

$$E_T^\alpha = \sum_m E_{Tm}^\alpha, \text{ где } E_{Tm}^\alpha = \mu(\{I_l \mid \exists i : \rho_{lm}^i \geq \alpha\}), \quad (4.7)$$

где величина Q_{Tl}^α характеризует, насколько постоянно идеальный объект I_l сопровождается автоматическими объектами, а величина E_{Tm}^α показывает, сколько раз автоматический объект A_m ошибочно меняет сопровождаемые идеальные объекты. Для оценки качества выделения траекторий интересны интегральные соотношения $Q^\alpha = \frac{Q_T^\alpha}{\sum_l \mu(I_l)}$, $E^\alpha = \frac{E_T^\alpha}{\mu(\{A_m\})}$, которые, соответственно, показывают долю кадров, на которых идеальные объекты сопровождались автоматическими объектами, и число ошибок смены траектории по отношению к общему числу автоматических объектов.

На сайте www.changedetection.net, кроме видеопоследовательностей, доступны все результаты обработки всех изображений методами вычитания фона, упомянутыми в работе [123], в форме маски. Эти результаты можно использовать для эмуляции

работы алгоритма вычитания фона на первом этапе алгоритма выделения траекторий движущихся объектов, что позволит для всех методов вычитания фона получить автоматическую аннотацию, а затем метрики качества для всех категорий видеоизображений в виде

1. $F_p = F_P^{\alpha\beta}$ при $\alpha = 0.5$ и $\beta = 0.3$.
2. $F_n = F_P^{\alpha\beta}$ при $\alpha = 0.5$ и $\beta = 0.3$.
3. $Q = Q^\alpha$ при $\alpha = 0.5$.
4. $E = E^\alpha$ при $\alpha = 0.5$.

Данные метрики могут быть вычислены с помощью специального программного обеспечения, которое сопоставляет идеальную разметку с автоматической, полученной на основе алгоритмов сопровождения движущихся объектов. Используя эти метрики, можно рассчитать обобщённые показатели качества, применив тот же подход ранжирования, который производился при расчёте метрик RC_m и R_m для методов вычитания фона. Изменится только состав и количество исходных метрик. Вычисленные ранги методов конечного алгоритма слежения за движущимися объектами обозначим RTC_m и RT_m , соответственно. Итоговые результаты сравнения алгоритмов сопровождения движущихся объектов, в основе которых лежат разные алгоритмы вычитания фона, приведены в таблице 4.4. В таблице представлены все методы, рассмотренные в работе [123], в порядке убывания результатов по метрике RTC . Кроме того, отдельной строкой «Baseline» представлены модельные результаты, полученные для случая, когда в алгоритм сопровождения движущихся объектов подставляются шаблонные точки движения, полученные в результате ручной разметки видеоданных.

Качество алгоритмов вычитания фона напрямую влияет на конечное качество сопровождения отдельных объектов: это видно из того, что работа алгоритмов сопровождения на основе введённых вручную точек движения получила наивысшую оценку качества в таблице 4.4.

С другой стороны, отсутствует прямая корреляция между результатами по сравнению алгоритмов вычитания фона и результатами работы алгоритмов выделения траекторий объектов на их основе. Если не учитывать шаблонную разметку, то с точки зрения метрики RTC лучше всего работает алгоритм, являющийся средним по метрике R , поэтому за счёт дополнительной обработки можно получить хороший результат по алгоритму отслеживания объектов, используя быстрые методы вычитания фона. Возможно, отсутствие корреляции связано в том числе с недостатками вычисления меры совпадения множеств точек при оценке качества сегментации, рассмотренные в предыдущем параграфе.

Метод	RTC	RT	RC	R	F _p	F _n	Q	E
Baseline	6,1875	4,25	-	-	11	80	0,35	0,31
GMM KaewTraKulPong	7,29	7,75	9,5	9,43	363	94	0,40	0,35
ViBe+	7,48	6	4,83	5	309	91	0,40	0,39
ViBe	7,58	7,75	9,33	10,17	296	102	0,39	0,35
SOBS-SC	8,5	6,5	6	6,14	246	98	0,31	0,26
Bayesian Back	8,77	5,625	11	12,57	268	109	0,40	0,27
KDE Nonaka	8,875	8,25	9,17	8,43	422	102	0,35	0,27
SOBS	9,15	7,75	8,17	8,57	295	108	0,32	0,261
KDE Elgamal	9,67	10,875	9,67	11,43	295	107	0,32	0,36
GMM Stauffer Grimson	9,99	9,5	11,5	10,14	341	110	0,36	0,29
PBAS	10,04	9,75	3	3,29	287	102	0,31	0,34
GMM Zivkovic	10,19	9,875	13,67	10,86	320	108	0,34	0,34
KDE Yoshinaga	10,94	9,75	10,67	9,29	417	113	0,39	0,28
Euclidian distance	11,06	13,125	16,67	14	351	109	0,34	0,42
PSP-MRF	11,17	12,25	4,83	5	189	122	0,30	0,35
GMM RECTGAUSS-Tex	12	13,75	13,67	13	125	165	0,20	0,44
Mahalonobis distance	12,13	15	15,5	13,43	483	116	0,33	0,36
Chebyshev probability	12,83	14,25	6,67	5,86	186	143	0,18	0,48
Local Self-similarity	16,19	18	14,67	13,14	427	191	0,10	0,42

Таблица 4.4. Характеристики алгоритмов сопровождения движения на основе различных методов вычитания фона.

Анализ таблицы 4.4 также показывает, что даже для «идеальной» разметки точек движения предложенный алгоритм трекинга объектов совершаet довольно много ошибок.

Таким образом, необходимо комплексно подходить к проблеме автоматического выделения событий на изображении видеокамер. Недостаточно получить хороший результат только для одной из вычислительных подсистем при идентификации событий в системах компьютерного зрения, поскольку существует взаимное влияние подсистем друг на друга при формировании конечного результата.

4.3. Анализ методик оценки точности алгоритмов классификации

При оценке точности алгоритмов семантической классификации изображений необходимо сравнить категорию, к которой отнёс изображение объекта тот или иной алгоритм семантической обработки, с действительной категорией объектов. Метрика качества подобных алгоритмов строится на основе доли ошибок классификации по

отношению ко всем попыткам автоматического распознавания класса, и вычисление подобных осуществляется с использованием функции категории объекта χ , введённой в предыдущем параграфе. Качество классификации событий можно оценить с помощью метрики

$$F_S^{\alpha\beta} = \mu \left(\left\{ I_l \middle| \exists m : \frac{\mu(\{i | \rho_{lm}^i < \alpha\})}{\mu(I_l)} < \beta \vee \chi(I_l) \neq \chi(A_m) \right\} \right) / \mu(\{I_l\}), \quad (4.8)$$

где $F_S^{\alpha\beta}$ – доля неправильно классифицированных идеальных объектов. В знаменателе выражения (4.8) вычисляется количество таких идеальных объектов, для которых не выполнен предикат из формулы (4.5), т.е. для них существуют некие автоматические объекты, которые хорошо пересекаются по положению на большом количестве кадров, но при этом их свойства χ не совпадают. Аналогично можно построить матрицу из показателей по каждому из значений χ : насколько часто алгоритмы выдают, например, объекты типа «группа людей» для объектов, которые на самом деле автомобили.

Проблемой при анализе точности алгоритмов классификации является оценка достоверности полученных по формуле (4.8) ошибок. Рассмотрим методику оценки достоверности классификации на примере классификатора символов в модуле распознавания автомобильных номеров транспортных средств [159].

На текущий момент существует много систем, которые выполняют автоматическую компьютерную обработку видеоизображений потока транспорта с целью выделения и идентификации государственных регистрационных знаков транспортных средств. Подобная обработка может быть как в составе мониторинговых систем дорожного движения, предназначенных для оценки транспортного спроса в ключевых узлах дорожной сети, так и в системах контроля безопасности дорожного движения. Очевидно, что эффективность алгоритма распознавания номерных знаков является критичной для таких систем, поскольку ошибки распознавания приводят к погрешностям измеряемых статистических характеристик транспортного потока и к неприменимости системы в целом из-за экономической нецелесообразности её использования. Считается, что порог применимости отдельно взятого алгоритма идентификации номеров примерно равен 95% по числу номеров, идентифицируемых алгоритмом без ошибок.

Несмотря на то, что многие системы идентификации номеров преодолели порог применимости, разработка новых или усовершенствование существующих алгоритмов является актуальной проблемой по двум основным причинам. Во-первых, любой существующий алгоритм совершает ошибки, которые необходимо устранять. Во-вторых, очень часто с целью удешевления вычислительных приборов, которые исполняют алгоритм, приходится модифицировать или полностью перерабатывать уже

существующий алгоритм, чтобы он мог исполняться менее мощным вычислителем, не сильно теряя при этом в качестве распознавания.

Существуют два основных подхода при построении алгоритмов идентификации изображений регистрационных знаков транспортных средств. Первый подход заключается в том, что на основе государственного стандарта строятся модели изображений номера, и потом каждая из этих моделей сопоставляется с изображением номерного знака на входе алгоритма, результат выводится по наилучшему совпадению моделей.

Второй, более распространённый подход заключается в том, что для построения модели номерного знака используется не государственный стандарт, а значительная выборка изображений автомобильных номеров, которые предварительно подвергаются разметке: каждое изображение сопоставляется с набором ожидаемых результатов распознавания номерного знака, включающим строку с символами знака и их положение на изображении. Для построения модели на основе большой выборки изображений используется либо нейросетевой подход, либо приёмы, подобные методам построения нейросетей, включающие в себя построение признаковых пространств и решающих правил [25, 160, 161, 162]. Широкое распространение данного подхода с использованием машинного обучения объясняется тем, что, во-первых, государственные стандарты время от времени подвергаются пересмотру, и во-вторых, существуют страны, которые не чётко регламентируют внешний вид и структуру номерных знаков. К таким странам относится, например, США. Наконец, даже при использовании модели, основанной на стандартах, невозможно создать качественный алгоритм идентификации номерного знака без системы верификации этого алгоритма, а на текущий момент единственным достоверным способом верификации является проверка алгоритмов на большой выборке изображений.

Таким образом, при построении любого алгоритма распознавания автомобильных номеров требуется выборка изображений большого объёма. Далее приведена оценка количества изображений номеров, необходимых для построения эффективного алгоритма идентификации номерных знаков.

Для примера рассмотрим номера, соответствующие ГОСТ Р 50577-93[163]. Наиболее распространённые на данный момент номера соответствуют шаблону вышеуказанного ГОСТ, в котором буква предшествует 3 цифрам, потом следуют ещё две буквы, и в конце расположены две или три цифры, относящиеся к коду региона. Для простоты ограничим нашу задачу только распознаванием знаков номера, ошибки локализации изображений номеров внутри кадра мы рассматривать не будем. Таким образом, проверять будем ту часть алгоритма идентификации номеров, на вход которой поступает изображение с локализованными элементами номерного знака, а на выходе алгоритма строка символов, соответствующая знакам номера.

Введём следующие обозначения: N — число символов в номере ($N = 8$ для номеров с двумя цифрами в коде региона и $N = 9$ для номеров с тремя цифрами в коде региона); S — множество символов, используемых в номерах ($S = L \cup D$, где $L = \{A, B, E, K, M, H, O, P, C, T, Y, X\}$ — буквы, а $D = \overline{0..9}$ — цифры); E_l^s — вероятностное событие, заключающееся в неверном распознавании символа $s \in S$, находящегося в позиции l относительно первого символа номера; H_l^s — вероятностное событие возникновения символа $s \in S$, находящегося в позиции l относительно первого символа, в произвольно взятом номере. В предположении независимости ошибок E_l^s вероятность правильного распознавания номера можно представить как

$$P = \prod_{l=1}^N (1 - P(E_l^s)) = \prod_{l=1}^N \left(1 - \sum_{s \in S} P(E_l^s | H_l^s) P(H_l^s)\right). \quad (4.9)$$

Введём упрощение в модель распределения ошибок, сделав предположение, заключающееся в том, что ошибка $P(E_l^s)$ зависит только от s , и не зависит от положения символа в номере l , более того,

$$P(E_l^s) = \begin{cases} p_L, & \text{если } s \in L, \\ p_D, & \text{если } s \in D. \end{cases} \quad (4.10)$$

Ограничимся рассмотрением номеров с двумя цифрами в коде региона, т.е. номерами с 8 символами, так как для номеров с 9 символами рассуждения аналогичны. Тогда, считая, что в коде региона могут быть все цифры¹, мы получим в рамках упрощённой модели

$$P = (1 - p_D)^5 \cdot (1 - p_L)^3, \quad (4.11)$$

где показатели степеней соответствуют вероятности не сделать ошибки ни в одной из 5 цифр и ни в одной из 3 букв номерного знака.

Введём обозначение $\tilde{p} = \max(1 - p_D, 1 - p_L)$. Тогда из формулы (4.11) следует

$$P \leq \tilde{p}^8. \quad (4.12)$$

Для того чтобы оценить уровень распознавания P с погрешностью $\pm 3\sigma$, необходимо оценить уровень распознавания отдельного символа \tilde{p} с погрешностью $\pm 3\tilde{\sigma}$, причём согласно приближению формул ряда Тейлора из формулы (4.12) следует

$$\tilde{\sigma} \approx \sigma/8. \quad (4.13)$$

¹На данный момент это не так: из 100 возможных сочетаний двух цифр несколько пар в кодах регионов не применяются. Пренебрежение этим фактом несущественно смешает оценку количества примеров в сторону увеличения

Разумно ограничить неопределённость оценки уровня распознавания минимальным уровнем

$$3\sigma = (1 - P)/2. \quad (4.14)$$

В этом случае при $P = 0,95$ неопределённость результата будет примерно в пределах от 0,93 до 0,97. На основании (4.13) и (4.14) получаем

$$\tilde{\sigma} = (1 - P)/48. \quad (4.15)$$

Согласно [164], \tilde{p} , $\tilde{\sigma}$, и количество необходимых примеров M изображения конкретного символа связаны формулой

$$\tilde{\sigma} = \frac{\sqrt{\tilde{p}(1 - \tilde{p})}}{\sqrt{M}}. \quad (4.16)$$

Таким образом, на основании (4.12), (4.15) и (4.16), имея ввиду убывание функции $\tilde{p}(1 - \tilde{p})$ при $\tilde{p} \geq 0,5$, получаем количество примеров

$$M \geq \frac{2304\sqrt[8]{P}\left(1 - \sqrt[8]{P}\right)}{(1 - P)^2}, \quad M \geq 5853 \text{ при } P = 0,95. \quad (4.17)$$

Итак, для того, чтобы достоверно верифицировать факт работы алгоритма распознавания номеров с уровнем ошибок 95%, требуется, чтобы в примерах изображений, на которых происходит верификация, было $M \geq 5853$ каждого из символов множества S . При этом изображений номеров требуется несколько меньше, поскольку на одном номере может быть несколько одинаковых символов.

Обозначим количество номеров, которые содержат M конкретных символов $\alpha \in S$, как N , и будем исходить из того, что наша выборка номеров объёма N случайна, причём вероятность попадания каждого из существующих номеров в нашу выборку одинакова. Введём вероятность $P(\alpha, n, k)$ для события, заключающегося в том, что n -значный номер из нашей выборки содержит ровно k символов $\alpha \in S$. Тогда справедливо следующее соотношение:

$$M = \sum_{k=1}^n k \cdot P(\alpha, n, k) \cdot N. \quad (4.18)$$

Вероятность $P(\alpha, n, k)$ можно рассчитать комбинаторными методами. Понятно, что в номерах, соответствующих ГОСТ Р 50577-93, $P(\alpha, n, k) = 0$ для цифр $\alpha \in D$ при $n > 5$ и для букв $\alpha \in L$ при $n > 3$, так как стандарт не подразумевает наличие такого количества цифр и букв в номере. Для остальных троек (α, n, k) расчёт ведётся отдельно для букв и цифр.

Пусть n_α — количество мест, отведённых в номере под символы типа α , m_α — мощность множества символов типа α , т.е.

$$n_\alpha = \begin{cases} 3, & \alpha \in L, \\ 5, & \alpha \in D, \end{cases} \quad m_\alpha = \begin{cases} |L| = 12, & \alpha \in L, \\ |S| = 10, & \alpha \in D. \end{cases} \quad (4.19)$$

Количество слов K длины n , составляемых из m символов, в которых символ α встречается ровно k раз, определяется формулой

$$K = C_n^k \cdot (m - 1)^{n-k}. \quad (4.20)$$

Рассуждения, доказывающие эту формулу, следующие. Если k позиций заняты символом α , то остальные $n - k$ позиций заняты остальными символами, количество которых $m - 1$. Т.е. всего есть $(m - 1)^{n-k}$ вариантов расстановки оставшихся символов. Число расстановок оставшихся k символов в n позиций соответствует C_n^k . Отсюда следует формула (4.20).

Всего слов длины n , составляемых из m символов, как известно, равно m^n , откуда на основании формул (4.19) и (4.20) следует, что

$$P(\alpha, n, k) = \frac{K}{m_\alpha^{n_\alpha}} = \frac{C_{n_\alpha}^k \cdot (m_\alpha - 1)^{n_\alpha - k}}{m_\alpha^{n_\alpha}}. \quad (4.21)$$

Пользуясь (4.17), (4.18) и (4.21), получаем итоговую оценку необходимого количества изображений номеров для верификации достоверности алгоритмов P :

$$N \geq \frac{m_\alpha^{n_\alpha} \cdot 2304 \sqrt[8]{P} \left(1 - \sqrt[8]{P}\right)}{\left[\sum_{k=1}^{n_\alpha} k \cdot C_{n_\alpha}^k \cdot (m_\alpha - 1)^{n_\alpha - k}\right] \cdot (1 - P)^2}. \quad (4.22)$$

Подставляя в эту формулу значения для цифр, получаем требуемое количество номеров $N > 11704$. Для букв требуемое количество номеров $N > 23409$, соответственно. Окончательно именно последнее ограничение определяет требуемое количество номеров для подтверждения достоверности распознавания алгоритмов на уровне 0,95.

Может показаться парадоксом тот факт, что всего трёхбуквенных комбинаций в номерах, соответствующих ГОСТу РФ, равно $12^3 = 1728$, а примеров требуется более 20 тысяч. Однако на основании формулы (4.20) мы можем получить, что в этих трёхбуквенных комбинациях содержится только 498 символов A , например, поэтому для набора требуемого количества примеров (около 6 тысяч) придётся использовать повторы комбинаций.

На практике к основным источникам ошибок алгоритмов можно отнести шумовые явления (шум видеосигнала, загрязнение номеров из-за погодных условий), изменения

условий освещения в течение суток (смена дня и ночи, смена режимов уличного освещения), изменение условий в течение года при смене сезонов, смена погодных условий (переменная облачность, туман, дождь), изменение углов съёмки видеозображений номеров, различная скорость транспортных средств (влияет на чёткость изображения номера из-за экспозиции камеры). Для того чтобы выборка, на которой верифицируется алгоритм, была репрезентативной, желательно для каждого типа условий иметь более-менее равное количество примеров.

Сбор необходимой выборки изображений в этих условиях представляется сложной задачей. Можно упростить этап разработки алгоритмов, введя синтез изображений номеров. Практически любой фактор можно промоделировать с помощью фильтров, преобразующих изображение. Например, смену освещённости можно промоделировать на основе данных о свойстве материалов и красок, из которых производятся автомобильные номера, и на основании коэффициентов отражения типичных номеров из одного реального примера с известной освещённостью синтезировать некоторое количество искусственных примеров, применив фильтры преобразования уровней яркости. Смена угла съёмки эмулируется применением проективного преобразования картинки, параметры которого задаются исходя из параметров съёмки исходной картинки (угол зрения камеры и угол оптической оси к плоскости таблицы номера). Влияние экспозиции на потерю чёткости движущегося номера можно эмулировать с помощью изображения номера, полученного от неподвижной машины с дальнейшим сложением последовательных изображений номеров, полученных моделированием перемещения с определённой скоростью. Сами изображения номеров тоже можно синтезировать на основе правил ГОСТ.

Теоретически, набрать необходимый объём тестовых примеров можно вообще без реальных съёмок, используя исключительно синтезируемые примеры. К сожалению, синтезированные примеры могут лишь помочь разработке алгоритмов. Верификация же алгоритмов всё равно останется под вопросом: вряд ли модели, по которым получались синтезированные изображения, можно считать полностью эквивалентными реальным изображениям. Это значит, что для полноценной верификации требуется несколько десятков тысяч реальных, а не синтезированных примеров.

Методику оценки точности классификации с помощью алгоритмов семантического сжатия и расчёт объёмов тестовой выборки изображений, которые легли в основу приведенных выше материалов, можно применять не только для алгоритмов распознавания символов, но и для оценки достоверности семантической обработки других типов изображений.

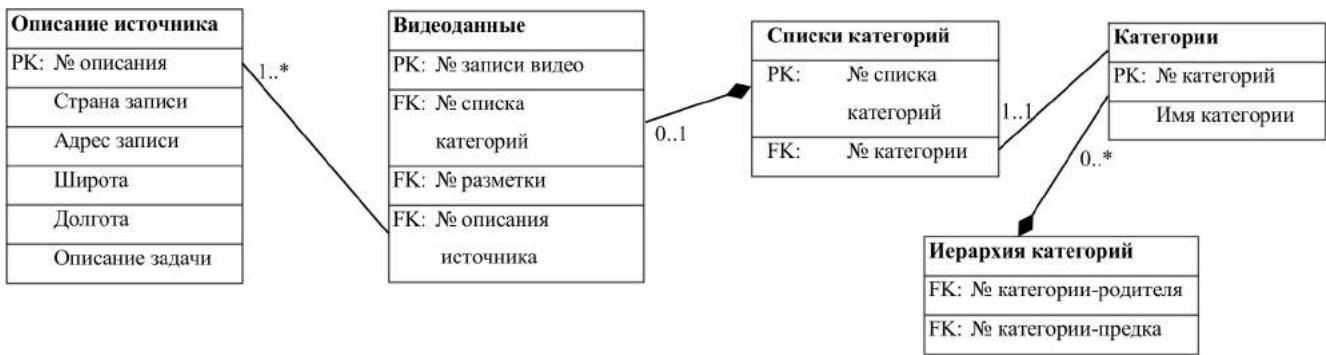


Рисунок 4.5. Архитектура базы данных информационной системы хранения видеоматериала

4.4. Разработка программного обеспечения для автоматизированной оценки достоверности алгоритмов обработки видеоизображений

Разработка любого алгоритма распознавания и идентификации в компьютерном зрении начинается с создания набора изображений или видео, на котором можно будет протестировать разрабатываемый алгоритм. Для обеспечения проверки качества работы требуется большой набор видеоматериала, см. [159, 164]. Практически каждая проблема требует для решения около 1000 примеров как минимум. Поскольку алгоритмы компьютерного зрения обычно являются составной частью целых комплексов систем видеонаблюдения, то для поддержания процессов разработки и тестирования данных комплексов требуется система хранения видеинформации с особой архитектурой, отличающейся от древовидной файловой структуры, поддерживаемых большинством современных операционных систем [165]. Система файловых каталогов отражает связь один ко многим, но в случае тестовых видеоданных информационные связи более сложные: одна видеозапись может применяться и для тестирования детектора оставленных предметов, и для оценки детектора движущихся объектов. В данной статье рассматриваются требования к архитектуре хранения видеоданных, предназначенных для разработки алгоритмов компьютерного зрения.

Система хранения видеоданных должна поддерживать возможность отнести изображения и видео сразу к нескольким категориям: по категории решаемой задачи (например, идентификация автомобильных номеров, детектирование возгорания на изображении, детектирование исчезновения и появления предметов, детектирование лиц, распознавание жестов руками, и т.д.), по категории условий места (помещение, улица), по категории географического положения локации, в которой получен материал, по категории погодных условий и условий освещения (постоянное или переменное освещение, ночное или дневное время, солнечная или пасмурная погода и т.д.), по категории насыщенности сцены событиями (разреженная частота событий от 1 в минуту, средняя частота от 1 в 10 секунд, плотная частота событий менее 1 в секунду). Видеохранилище должно поддерживать возможность расширять как состав катего-

рий, так и состав элементов внутри категорий, см. рис 4.5. Должна быть возможность строить иерархии внутри каждой категории, например, категория «переменное освещение» может подразделяться на «переменное естественное освещение» (из-за переменной облачности, например) и «переменное искусственное освещение» (из-за нестабильной работы осветительных приборов).

Кроме набора категорий, с каждым набором видео и изображений может быть связано покадровое описание присутствующих на них объектов (см. [166, 167]), так называемая разметка или аннотация видеоматериалов. С помощью данной разметки, в том числе, можно оценивать плотность событий на каждом видеоролике или наборе изображений.

Материал, собранный для разработки интеллектуальных подсистем видеонаблюдения, и подвергнутый обработке для получения разметки, обладает большой стоимостью. Поэтому система хранения этого материала должна предусматривать резервирование и отказоустойчивость. С учётом того, что практически каждая из современных проблем в компьютерном зрении требует около 100 гигабайт аннотированных видеоданных для своего решения, а также с учётом того, что перед компаниями, занимающимися компьютерным зрением, стоит более 100 таких проблем, совокупный объём видеоданных в таких системах составляет десятки терабайт. Резервирование с полным дублированием данных в таком случае является крайне дорогостоящим, поэтому применяются системы частичного резервирования наподобие RAID-5 или RAID-6. Для повышения отказоустойчивости системы хранения данных всё же дублируются.

Информационная система хранения данных должна легко строить ответы на вопросы: сколько есть в системе видеороликов, которые предназначены для тестирования систем контроля проникновения в запрещённую зону, которые записаны при этом в ночное время суток и на которых присутствуют помехи в виде засветки от нестационарных искусственных источников освещения. Система должна обладать возможностью детализировать ответ на данный запрос: она должна предоставить информацию, какова длительность данных видеозаписей по времени, и сколько объектов интереса присутствует в аннотации к данным записям. В частности, для изображений с автомобильными номерами система должна обладать возможностью ответить на запросы следующего типа: сколько автомобильных номеров, заканчивающихся на цифры 777, присутствует в хранилище данных, сколько из них сняты днём, а сколько ночью, и т.п.

Информационная система хранения видеоданных — это самый важный ресурс компаний, выпускающих продукцию в области компьютерного зрения. Чем эффективнее она устроена, чем быстрее с её помощью можно получить ответ о составе ви-

деоданных, которые система хранит, тем быстрее и эффективнее будет проводиться разработка функциональных программных подсистем интеллектуальных анализаторов видеоизображения, повышающих эффективность и достоверность автоматических детекторов событий на изображении.

Оценка качества систем распознавания в компьютерном зрении может быть основана на проведении натурных испытаний: необходимо установить сравниваемые системы, например, систему подсчёта с помощью одной камеры и систему подсчёта с помощью стереокамер, в одном и том же месте, затем одновременно запустить эти системы и провести натурные испытания: например, записать в журнале, сколько человек прошло на самом деле, и каковы были результаты работы обоих систем. Несмотря на концептуальную простоту, у подобных натурных испытаний есть серьёзные недостатки. Во-первых, в качестве выходных параметров видеосистем при испытаниях мы должны получить не только вероятностную точность работы системы распознавания, например, в виде достоверности, но и получить доверительный интервал для этого показателя. Доверительный интервал должен быть достаточно узким для того, чтобы можно было достоверно сравнить показатели различных систем. Если результат для одной системы $87\% \pm 5\%$, а результат другой системы $91\% \pm 5\%$, то однозначно судить о том, какая система лучше, нельзя. Существует вероятность, что лучше именно та система, у которой результат 87% , а не та, у которой результат 91% , поскольку важное значение приобретает пересечение доверительных интервалов у данных показателей. Но для того, чтобы сузить доверительные интервалы, в нашем примере это $[87\%-5\%, 87\%+5\%]=[82\%, 92\%]$ и $[91\%-5\%, 91\%+5\%]=[86\%, 96\%]$, необходимо провести испытания с очень большой выборкой. Как показано в работе автора [164], чтобы ширина полуинтервала была 2% , необходимо провести порядка 1000 испытаний, а для ширины полуинтервала $0,5\%$ требуются уже десятки тысяч испытаний. И поскольку число испытаний столь велико, невозможно обеспечить ручной контроль точности тех показателей, с которыми мы должны сравнивать результаты работы систем после проведения испытаний. Практика показывает, что записи в журналах, например, о проходах людей через турникеты будут содержать ошибки, обусловленные человеческим фактором, если требуется записать в журнал факты о 1000 или о 10000 таких событий. Второй существенный недостаток натурных испытаний заключается в том, что их невозможно повторить в точности заново: тысяча событий не произойдёт перед камерами во второй раз совершенно так же, как это произошло в первом испытании, к тому же освещённость наблюдаемой сцены, скорее всего, тоже будет изменяться во времени. Поэтому сравнение различных систем с помощью натурных испытаний надо проводить одновременно, т.е. невозможно сначала провести испытание одной системы, а потом другой. Из-за этого процесс проведения испытаний превращается в

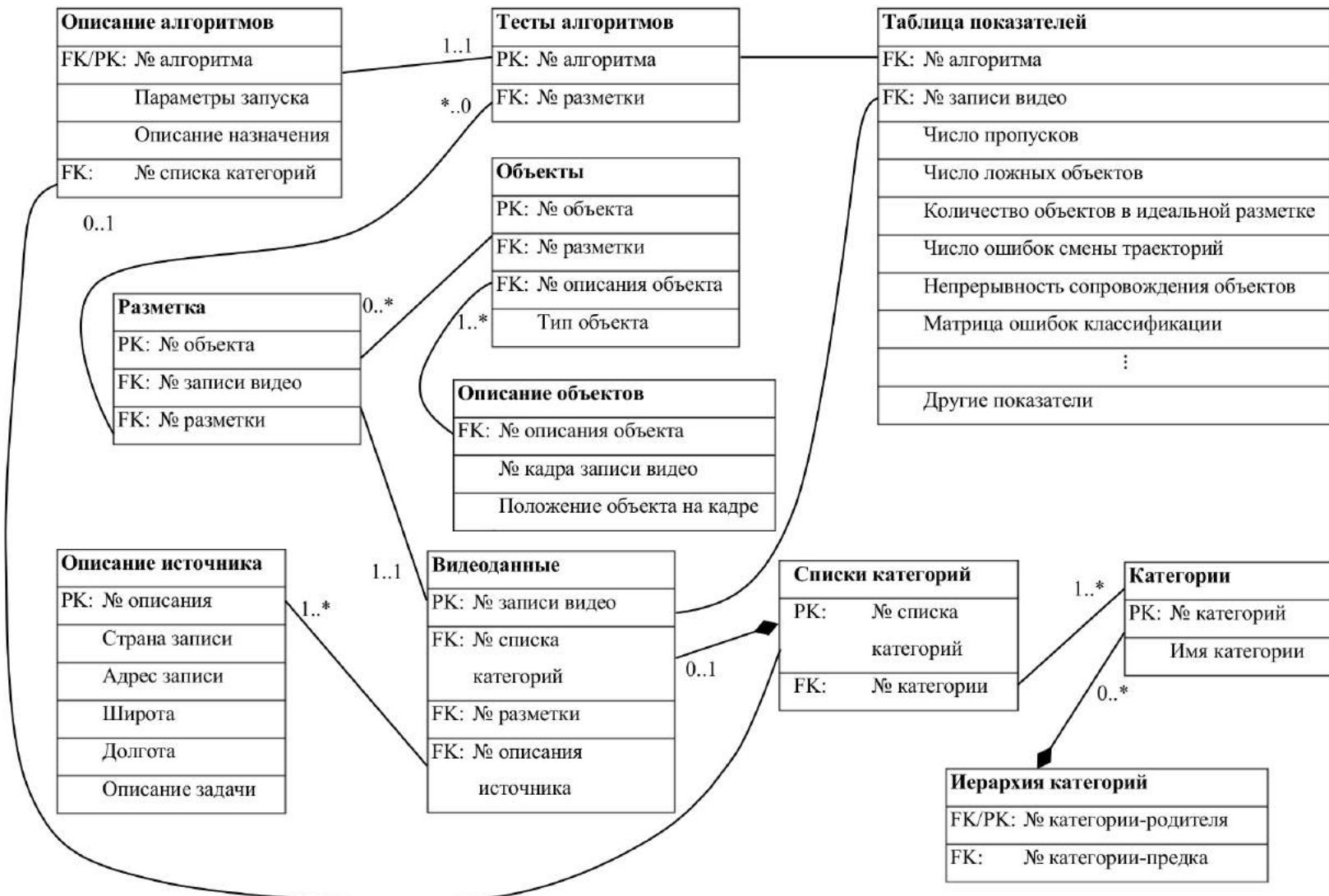


Рисунок 4.6. Схема базы данных для хранения разметки видео и результатов тестов (FK — внешний ключ, PK — индекс таблицы)

ресурсоёмкую по трудовым затратам задачу.

Учитывая вышесказанное, испытание алгоритмов семантической обработки видеинформации проводят на основе собранных данных в информационной системе хранения видеинформации. Каждый алгоритм семантической обработки тестируется: после запуска на изображениях в составе отдельных файлов и файлов с видеопотоком формируют автоматическую разметку видеоматериала (см. определение термина автоматическая разметка в параграфе 4.2), а затем сравнивают её с идеальной разметкой, хранящейся в информационной системе хранения видеоматериалов, используя для этого формулы (4.4)–(4.7) и (4.8). В результате система хранения видео становится частью системы базы данных, предназначенных для информации, описывающей тестирования алгоритмов семантической обработки, см. рис. 4.6. База данных включает в себя как видеоматериалы и аннотацию (идеальную разметку), так и результаты работы различных версий алгоритмов семантической обработки (автоматическую разметку) совместно с рассчитанными результатами сравнения идеальной и автоматической разметки.

Запуск тестирования различных алгоритмов на большом объёме видеоданных является требовательной к вычислительным ресурсам операцией. Например, в процессе разработки автором алгоритмов для систем ВАРШ.00002-01, ВАРШ.00003-01, ЛЦКБ.464412.002 и подобных им необходимо было обрабатывать видеофайлы длиной несколько суток (при средней частоте кадров 25к/с) алгоритмами семантической обработки, например, выделение движущихся объектов, отслеживание транспортных средств с распознаванием автомобильных номеров, детектирование оставленных предметов, детектирование огня и др., с разными входными параметрами, определяющими различный уровень чувствительности алгоритмов (баланс ошибок первого и второго рода). По мере роста объёма видеоданных в тестовой информационной системе, однопоточная обработка с помощью одного ядра современного компьютера с многоядерной структурой стала длиться около сотни часов, т.е. около 4 суток.

Поскольку скорость тестирования очень важна при активной разработке с внесением изменений в программный код реализации алгоритмов семантической обработки, были внедрены параллельные вычисления в несколько потоков на каждом ядре многоядерной вычислительной системы с разделяемой памятью: видеофайлы допускают обработку алгоритмами с одним из вариантов входных параметров (настроек), поскольку такая обработка не использует общих ресурсов в информационной тестовой системе. Каждый запуск формирует данные для отдельных строк в системе БД, изображённой на рис. 4.6. На уровне физических ресурсов общим является жёсткий диск с файлами видеохранилища, однако разграничения доступа к нему осуществляются на уровне операционных систем, установленном на тестовом компьютере. В результате

использования многопоточных вычислений время тестирования с сотни часов сократилось до 15 на ЭВМ с процессором, обладающим 8-ю ядрами, что также довольно долго.

На текущий момент существует много способов выполнить распределённые вычисления, разделив вычислительную нагрузку на некоторое количество ЭВМ. По большей части технологии распределённых вычислений базируются на обмене данных с помощью передачи сообщений между различными вычислительными узлами. Наиболее распространён способ, основанный на библиотеке MPI (Message passing interface) [168]. Можно также использовать для организации параллельных вычислений распределённые системы непрерывной интеграции, например, систему [169]. В рамках данной работы была использована система облачных вычислений HTCondor [170]. Данная система позволяет сравнительно просто организовать вычислительное облако на рабочих местах сотрудников современных предприятий. Системой предоставляются простые механизмы для управлением запуском и остановкой вычислительных потоков на узлах облака, при этом поддерживается динамическое состояние вычислительных узлов: отдельные ЭВМ облака могут выключаться и включаться в процессе расчётов, все невыполненные задачи при выключении узла будут назначены на активные ЭВМ. В результате применения системы [170] время тестирования в облаке из 7 ЭВМ на базе 8-ядерных процессоров сократилось с 15 до 3,5 часов, т.е. в итоге со 100 до 3,5 часов. Ожидаемое сокращение времени тестирования в $7 \cdot 8 = 56$ раз, т.е. до менее, чем 2 часа, не происходит в соответствии с законом Амдала [171] о доле последовательных вычислений.

Таким образом, неотъемлемыми частями современных систем тестирования алгоритмов семантической обработки изображений являются приложения по аннотации видеоматериала, система хранения данных с видеинформацией и её аннотацией, совмещённая с системой хранения результатов семантической обработки, а также механизмы облачных распределённых вычислений.

Выводы

1. Установлено, что современные методики тестирования не всегда точно устанавливают достоверность результатов семантической обработки.
2. На основе математического аппарата теории нечётких множеств предложен способ оценки качества семантической сегментации, устойчивый к случайным помехам.
3. Представлена новая теория тестирования алгоритмов семантической обработки и выведены формулы для новых характеристик оценки качества детекторов движущихся объектов в поле зрения камеры. Показано, что существующие системы оценки

качества сравнивают различные детекторы не вполне в соответствии с функциональными требованиями к алгоритмам семантической обработки данного класса.

4. Разработаны методики оценки объёма видеоматериала, необходимого для разработки новых алгоритмов семантической идентификации изображений.

5. Приведено описание системы тестирования алгоритмов семантической обработки, лежащей в основе ряда полученных в диссертации новых результатов.

5. Исследование эффективности алгоритмов семантической обработки видеоинформации при анализе стереоизображений

Стереосистемы всё более широко применяются в современных системах компьютерного зрения. В рамках данной главы будет рассмотрен ряд новых подходов к обработке стереоизображений. Суть данных подходов заключается в комбинировании результатов обработки изображения от одной камеры с картой глубины, вычисленной по синхронным изображениям двух видеокамер. В начале главы будет проведён анализ эффективности обработки изображений без использования стереоэффекта. Затем будет проведено исследование способов повысить достоверность результатов за счёт комбинированной обработки изображений с двух камер стереосистемы с применением автоматически вычисленной оценки расстояний до изображённых объектов сцены.

5.1. Основы теории построения и анализа алгоритмов калибровки стереоизображений и восстановления глубины

В современных системах технического зрения всё чаще применяется стереоэффект для получения дополнительной информации о сценах, наблюдаемых в поле зрения камеры. Типичная видеосистема на основе стереокамер состоит вычислительного устройства и подключенных к нему пары камер, направленных в одну сторону и расположенных на небольшом расстоянии друг от друга. Расстояние между камерой называется базой стереосистемы.

Предположим, что параметры камер со значением базы b в стереосистеме одинаковы, т.е. у камер совпадают фокусное расстояние у камер f , ширина чувствительных матриц w' , высота h' и разрешение $W \times H$. Также совпадают фокальные плоскости двух оптических систем, в которых расположены чувствительные матрицы, из чего следует, что оптические оси двух камер сонаправлены, и горизонтальные стороны соответствующих прямоугольных элементов чувствительной матрицы находятся на одной прямой. Если выполнены все вышеперечисленные ограничения на параметры камер в стереосистемы, то мы назовём эти камеры канонической стереопарой, а саму систему — канонической стереосистемой. Схема канонической стереосистемы изображена на рис. 5.1.

Представленная схема иллюстрирует, что объект T , находящийся на расстоянии L от стереопары камер с оптическими центрами O_1 и O_2 , проектируется в точки I_1 и I_2 фокальных плоскостей, принадлежащих камерам, которые мы будем условно называть левой и правой, соответственно. Если мы проведём из точки O_1 прямую, параллельную отрезку O_2I_2 , то она пересечёт фокальную плоскость в точке I'_2 . Эта точка соответствует точке, в которую переходит точка I_2 при параллельном переносе, накладывающим изображение правой камеры на изображение левой камеры. Обо-

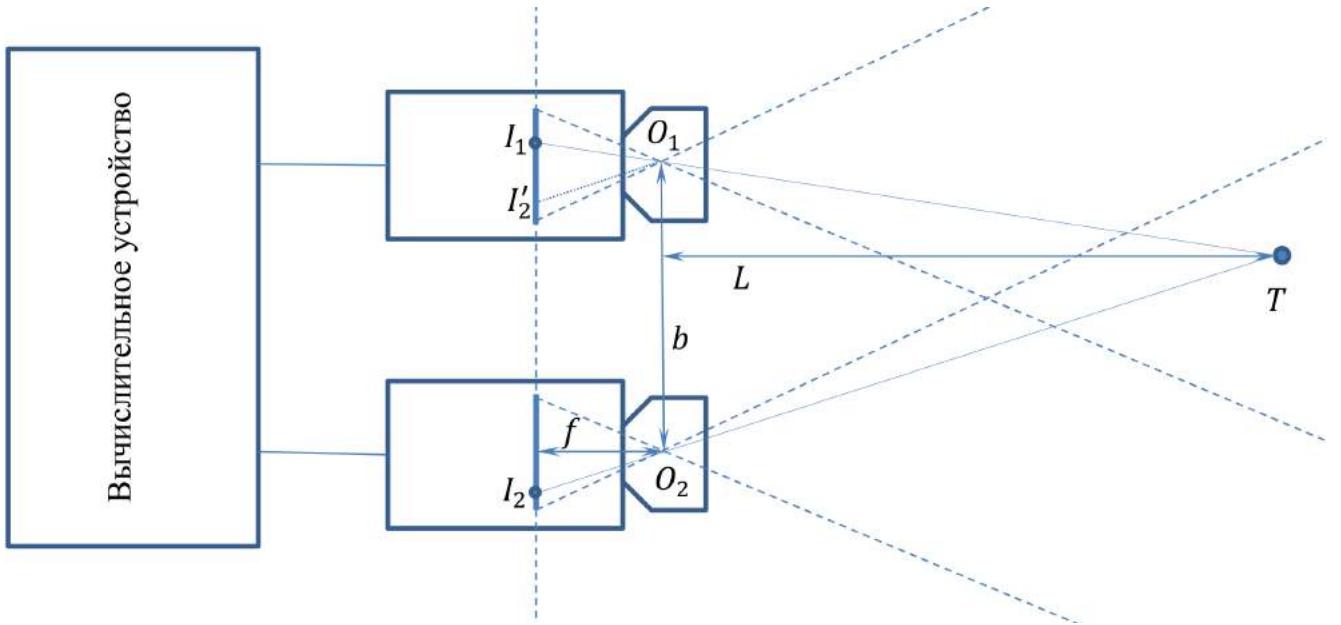


Рисунок 5.1. Схема видеосистемы на основе стерео

значим расстояние $d' = |I_1I'_2|$, тогда из подобия треугольников $\triangle TO_1O_2$ и $\triangle O_1I_1I'_2$ следует соотношение

$$\frac{d'}{b} = \frac{f}{L}. \quad (5.1)$$

Если $F_r(x, y)$ — кадр с изображением правой камеры, $F_l(x, y)$ — кадр с изображением левой камеры, и объект T изображён в точке (x_T^l, y_T^l) левой камеры, то с помощью методов компьютерного зрения можно найти на кадре F_r точку (x_T^r, y_T^r) , в которой изображение соответствует изображению окрестности точки (x_T^l, y_T^l) на кадре F_l . При этом можно воспользоваться, например, одним из методов сопоставления изображений, описанным в параграфе 1.3. По построению нашей системы, вертикальные координаты объекта T на правом изображении будут совпадать с координатами на левом изображении, т.е. $y_T^r = y_T^l$, так что при поиске окрестности изображения достаточно двигаться вдоль горизонтальной координаты. Обозначим $d = \Delta x = |x_T^r - x_T^l|$ — разность горизонтальной координаты изображения точки T на левой и правой камере. Эту величину будем называть диспаритетом. Поскольку измеряемая в метрах величина d' связана с пиксельной разностью d пропорцией $d/W = d'/w'$, то завершив поиск диспаритета, можно восстановить расстояние от объекта до камер на основании формулы (5.1):

$$L = \frac{fbW}{dw'} = \frac{f'b}{d}, \quad (5.2)$$

где $f' = fW/w'$ — коэффициент, измеряемый в пикселях изображения, характеризующий угол зрения и разрешение камеры. Таким образом, восстановив по цифровым изображениям диспаритет в точке объекта, при известных параметрах стереопары камер возможно оценить расстояние от точки наблюдения до этой точки. Т.е. по изображениям $F_l(x, y)$ и $F_r(x, y)$ можно восстановить расстояния до изображаемых точек

$L(x, y)$. Функцию $L(x, y)$ называют картой глубины, а соответствующую ей функцию $d(x, y)$ — картой диспаратностей. Данные функции широко используются в системах технического зрения.

Проблема построения карты диспаратностей $d(x, y)$ на основе двух изображений $F_l(x, y)$ и $F_r(x, y)$ состоит из двух частей. Во-первых, предположения, в которых выведена формула (5.1), о расположении камер в пространстве и о совпадении их параметров, на практике обычно не выполняются. Поэтому изображения левой и правой камеры сначала подвергают так называемой ректификации, являющейся преобразованием, которое подбирают в процессе калибровки стереопары таким образом, что получившиеся в результате преобразования кадры $F'_r(x, y)$ и $F'_l(x, y)$ соответствуют изображениям некоторой канонической стереопары с известными параметрами. Во-вторых, необходимо либо выбрать существующий метод вычисления $d(x, y)$, поскольку на данный момент исследователями предложено множество методов расчёта диспаратности по двум ректифицированным изображениям со своими достоинствами и недостатками, либо разработать новый метод, если ни один из существующих методов не подходит.

Для калибровки стереопары применяется множество различных методов [100]. Чаще всего для построения преобразования ректификации изображений применяется метод, предложенный в работе [101], поскольку идеи, изложенные в этой статье, реализованы в общедоступных программных кодах, а именно в библиотеке OpenCV [102], и в среде Matlab [103].

Для видеокамеры, модель объектива которой соответствует идеальной тонкой линзе, трёхмерные координаты объектов сцены в системе координат $OXYZ$, которую мы будем называть мировой, связаны с координатами изображения объектов в системе Ouv посредством следующих матричных формул:

$$\alpha \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = M \cdot [R|P] \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \text{ где } M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \text{ — матрица параметров камеры,} \quad (5.3)$$

а именно: фокусные расстояния f_x и f_y для, соответственно, горизонтального и вертикального угла зрения, измеренные в единицах пикселей изображения, и центр проективного преобразования (c_x, c_y) в координатах изображения камеры; $[R|P]$ — составная матрица размером 3×4 из матрицы 3×3 поворота камеры R и столбца $P = [p_x \quad p_y \quad p_z]^T$, определяющего позицию камеры. Матрица M определяет внутренние параметры видеокамеры, зависящие только от свойств оптической системы и чувствительной матрицы, и не зависящие от положения видеокамеры в простран-

стве, матрицы R и P определяют внешние параметры, связанные с ориентацией и положением камеры в пространстве.

Уравнение (5.3) применимо для камер, в которых отсутствуют оптические aberrации. Часто в оптических системах присутствуют радиальная и тангенциальная дисторсия, и матричные формулы (5.3) необходимо дополнить моделью дисторсии, представленной в [104, 105, 106]. Согласно этой модели, координаты (x', y') , полученные после преобразования для идеальной линзы

$$\begin{aligned} \begin{bmatrix} x \\ y \\ z \end{bmatrix} &= R \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + P, \\ x' &= x/z, \\ y' &= y/z, \\ u &= f_x \cdot x' + c_x \\ v &= f_y \cdot y' + c_y \end{aligned}$$

должны быть подвергнуты преобразованию, обратному к преобразованию устранения радиальной и тангенциальной дисторсии (см. [106]):

$$\begin{aligned} x' &= x''(1 + K_1 r^2 + K_2 r^4 + \dots) + (T_2(r^2 + 2x''^2) + 2T_1 x'' y'')(1 + T_3 r^2 + T_4 r^4 \dots), \\ y' &= y''(1 + K_1 r^2 + K_2 r^4 + \dots) + (T_1(r^2 + 2y''^2) + 2T_2 x'' y'')(1 + T_3 r^2 + T_4 r^4 \dots), \end{aligned} \quad (5.4)$$

где $r^2 = x''^2 + y''^2$, K_i — коэффициенты радиальной дисторсии, T_i — коэффициенты тангенциальной дисторсии. Коэффициентами высоких порядков, обычно, пренебрегают. Если учитывать только коэффициенты первого порядка T_1 и T_2 для тангенциальной дисторсии, и первые шесть коэффициентов K_1, K_2, \dots, K_6 для радиальной дисторсии, то проведя расчёты, аналогичные расчётом в [107], получим, что приближённо обратное преобразование к преобразованию, представленному в формулах (5.4), можно записать в виде

$$\begin{aligned} x'' &= x' \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + 2t_1 x' y' + t_2(r^2 + 2x'^2), \\ y'' &= y' \frac{1 + k_1 r^2 + k_2 r^4 + k_3 r^6}{1 + k_4 r^2 + k_5 r^4 + k_6 r^6} + t_1(r^2 + 2y'^2) + 2t_2 x' y', \end{aligned} \quad (5.5)$$

где $r^2 = x'^2 + y'^2$, k_1, k_2, \dots, k_6 и t_1, t_2 — коэффициенты обратного преобразования от неискажённых дисторсий координат кискажённым. После преобразования $(x', y') \rightarrow (x'', y'')$ необходимо использовать преобразование с коэффициентами матрицы камеры M из соотношения (5.3):

$$\begin{aligned} u &= f_x \cdot x'' + c_x, \\ v &= f_y \cdot y'' + c_y. \end{aligned}$$

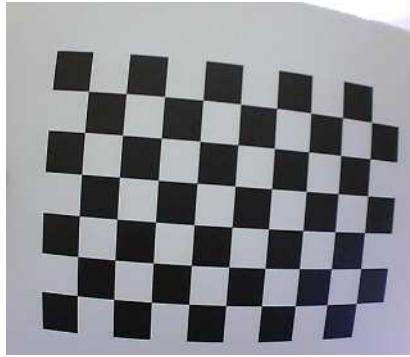


Рисунок 5.2. Пример изображения шаблона в виде шахматной сетки для калибровки внутренних параметров камер.

В процессе калибровки используют шаблон в форме шахматной клетки с известным количеством строк m и рядов n , а также с заранее измеренным размером клетки δ . Пример шаблона см. на рис. 5.2. Выбор сетки в виде шахматной клетки обусловлен тем, что существуют алгоритмы автоматического поиска внутренних узлов данной сетки, которые работают достаточно устойчиво. В результате работы такого детектора угловые точки клеток внутри шаблона автоматически выделяются на изображении в точках S_{ij} , $i = \overline{1, m}$, $j = \overline{1, n}$. Если мировую систему координат ориентировать так, что плоскость Oxy совпадает с плоскостью доски, причём оси параллельны сторонам доски, а точка O совмещена с самой левой и самой нижней внутренней угловой точкой шаблона. Тогда соответствующие точкам изображения S_{ij} координаты исходных точек \hat{S}_{ij} в ведённой мировой системе координат можно вычислить как $(l \cdot i, l \cdot j, 0)$, поскольку свойства калибровочного шаблона нам известны. С другой стороны, координаты исходных точек и проективные координаты на изображении связаны преобразованиями (5.3) и (5.5), т.е.

$$S_{ij} = \mathcal{T}(\hat{S}_{ij}, M, R, P, D), \quad (5.6)$$

где \mathcal{T} — рассмотренное ранее преобразование мировых координат сцены в координаты изображения, зависящее от матриц внутренних параметров камеры M , матрицы поворота R , позиции камеры P и вектора коэффициентов дисторсии $D = (k_1, k_2, \dots, k_6, t_1, t_2)$. Если вектор D состоит из нулевых значений, то, как следует из уравнения (5.5), преобразование \mathcal{T} приобретает форму (5.3).

Для калибровки стереопары выбранный шаблон помещают в поле зрения обоих стереокамер, варьируя угол плоскости шаблона по отношению к осям камер стереопары, и расстояние от шаблона до камеры. В результате обработки K синхронных изображений левой и правой камеры автоматически выделяются угловые точки шаблона на изображении $\{S_{ij}^{rk}\}_{k=1}^{k=K}$ и $\{S_{ij}^{lk}\}_{k=1}^{k=K}$, где индексы i и j соответствуют горизонтальным и вертикальным индексам углов шаблона, индекс k определяет номер синхронной пары изображений, на которых выделен шаблон шахматной доски с уг-

ловыми точками, и индексы l и r определяют, какой камере, соответственно левой или правой, принадлежит изображение, на котором выделены угловые точки шаблона. Поскольку координаты \hat{S}_{ij} привязаны к системе координат, в которой шаблон неподвижен, то для обоих камер стереопары справедливо

$$S_{ij}^{\bullet k} = \mathcal{T}(\hat{S}_{ij}, M^\bullet, R_k^\bullet, P_k^\bullet, D^\bullet),$$

где вместо знака \bullet подразумевается индекс l или r . В данной формуле зафиксировано, что \hat{S}_{ij} не зависит от положения шаблона по отношению к камере в системе координат, связанной с шаблоном, что внутренние параметры левой камеры M^l и D^l и правой камеры M^r и D^r также не зависят от положения шаблона по отношению к камере, и что матрицы ориентации оптической оси и положение камер R_k^l , R_k^r , P_k^l , P_k^r зависят от положения шаблона в силу выбора мировой системы координат.

Поскольку взаимное расположение камер в стереопаре фиксировано, то существует такая матрица поворота C и такой вектор \vec{b} , что при смещении на вектор \vec{b} правой камеры её оптический центр совпадёт с центром левой камеры, и при применении поворотов C относительно оптического центра правой камеры оптическая ось правой камеры станет параллельна оптической оси левой камеры. Это означает, что $P_k^l = P_k^r + \vec{b}$ и $R_k^l = CR_k^r$. В процессе калибровки стереопары для построения преобразования ректификации требуется найти элементы матриц и векторов M^l , M^r , D^l , D^r , C , \vec{b} . Одним из способов найти решение данной задачи является поиск значений, в которых достигается

$$\min_{\begin{array}{l} R_k^l = CR_k^r \\ P_k^l = P_k^r + \vec{b} \end{array}} \sum_{l,r} \sum_k \sum_{i,j} \|S_{ij}^{\bullet k} - \mathcal{T}(\hat{S}_{ij}, M^\bullet, R_k^\bullet, P_k^\bullet, D^\bullet)\|. \quad (5.7)$$

Поиск минимума в задаче (5.7) производится методом Левенберга-Марквардта [108, 109, 110], поскольку целевая функция является суммой квадратов нелинейных непрерывно дифференцируемых функций с совпадающими аргументами. В основе метода Левенберга-Марквардта лежит регуляризированный метод Ньютона, поэтому для сходимости метода важно выбрать достаточно близкое к глобальному минимуму приближение. Конкретный способ выбора начального приближения зависит от реализации калибровки стереопары. В частности, существует подход, при котором сначала отдельно оцениваются матрицы камер M^l и M^r , используя построение проективного преобразования между изображением шаблона и самим шаблоном, при этом полагается $D^l = D^r = 0$ (см. [101]). Затем при фиксированных параметрах M^l и M^r и найденных матрицах R_k^l , R_k^r , P_k^l , P_k^r с помощью метода Левенберга-Марквардта оцениваются параметры D^l и D^r . После этого при фиксированных D^l и D^r вновь уточняются параметры M^l и M^r . Далее запускается

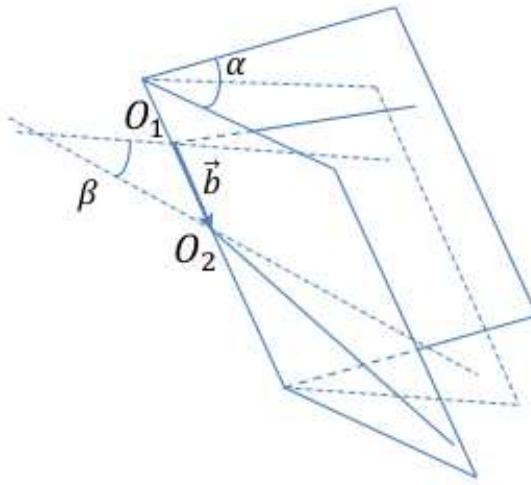


Рисунок 5.3. Поворот оптических осей камер для ректификации

поиск решения задачи (5.7), где в качестве начального приближения используются найденные ранее параметры $M^l, M^r, D^l, D^r, R_k^l, P_k^r$, а начальные матрицу C и вектор \vec{b} вычисляют на основе метода наименьших квадратов и набора равенств из условия исходной задачи.

Получив внутренние параметры камер M^l, M^r, D^l, D^r , а также параметры взаимного расположения C и \vec{b} , можно построить преобразование ректификации. Ректификация состоит из устранения дисторсии с помощью коэффициентов D^l и D^r , выравнивания разности масштабов в коэффициентах матриц M^l и M^r , и в цифровом преобразовании изображений, соответствующем повороту камер так, чтобы их оптические оси стали перпендикулярны вектору \vec{b} и параллельны друг другу, а оси системы координат на обоих изображениях были сонаправлены. Такие преобразования поворота для обоих камер определяются на основе взаимного расположения, задаваемого параметрами C и \vec{b} . Для этого определяют преобразования поворота R_1 и R_2 , которые поворачивают оптические оси обоих камер стереопары. Сначала поворот производится для плоскостей, проходящих через оси и через точки закрепления камер O_1 и O_2 ($\vec{b} = \overrightarrow{O_1 O_2}$), в сторону бисекторной плоскости на угол $\alpha/2$, если угол между плоскостями α (см. рис. 5.3). Затем оптические оси уже не являются скрещенными прямыми, и если между ними ненулевой угол β , необходимо развернуть их в вышеупомянутой бисекторной плоскости на угол $\beta/2$ друг от друга. Восстановлением перпендикулярности к вектору \vec{b} обычно пренебрегают, имея ввиду малость углов α и β . Для ректификации можно было бы поворачивать только одну камеру, но тогда на одной из камер пришлось бы наблюдать много таких точек изображения, которые невозможно установить после виртуального поворота, поскольку точки, которые должны быть отображены в этих областях, находятся вне поля зрения камеры. Подвергая преобразованию поворота обе камеры, можно более оптимально распределить невидимые зоны между левой и правой камерами стереопары.

Таким образом, применяя формулы (5.3) и (5.4) с найденными матрицами и коэффициентами для левой и правой камеры, получаем из исходных изображений $F_r(x, y)$ и $F_l(x, y)$ ректифицированные $F'_r(x, y)$ и $F'_l(x, y)$. Далее для простоты мы не будем применять штрихи в обозначении кадров, считая, что исходные изображения ректифицированы.

Существует множество способов найти функцию диспаратности $d(x, y)$, определение которой дано ранее. Базовые методы осуществляют для каждой ε -окрестности $U_\varepsilon^r(x_r, y)$ на правом изображении F^r самую похожую ε -окрестность $U_\varepsilon^l(x_l, y)$ на изображении F^l (координата y окрестностей совпадает, т.к. изображения ректифицированы). Для оптимальной работы вычислительных систем ε -окрестности формируются в пространстве с нормой L_0 , т.е.

$$U_\varepsilon^\bullet(x_\bullet, y) = \{(x', y) \mid \max(|x' - x_\bullet|, |y' - y|) < \varepsilon\} —$$

квадратная окрестность с центром в точке (x_\bullet, y) . Обозначим вектор из значений интенсивности изображений F_\bullet в окрестностях $U_\varepsilon^\bullet(x, y)$ как $v_\varepsilon^\bullet(x, y)$, компоненты вектора упорядочены по строкам и столбцам изображений. Тогда функция диспаратностей рассчитывается по формуле

$$d(x, y) = \arg \min_{x'} \|v_\varepsilon^r(x', y) - v_\varepsilon^l(x, y)\| - x. \quad (5.8)$$

Методы, основанные на формуле (5.8), описаны в работах [111, 112, 113]. Преимущество данных методов в небольшой вычислительной сложности: если количество точек на изображениях N , а максимальная диспаратность на изображениях D , то сложность поиска минимума (5.8) определяется как $O(ND\varepsilon^2)$. Для некоторых типов норм сравнения векторов v_ε^r и v_ε^l сложность алгоритма не зависит от размера окрестности и равна $O(ND)$. Однако данные методы не работают в точках разрыва функции $d(x, y)$ на границах объектов ближнего плана, так как в сопоставляемую окрестность попадают и ближние, и дальние точки сцены, более того, точки, видимые одной камерой, могут быть невидимы другой камерой стереопары.

Методы, основанные не на локальном сопоставлении изображений, а на глобальном поиске, более успешно решают проблему перекрытия и разрывов, но они являются более вычислительно сложными. Согласно [120], глобальный поиск, ускоренный с помощью метода динамического программирования [114, 115, 116], обладает вычислительной сложностью $O(ND^2)$ или, при некоторых допущениях, $O(ND \log D)$. Сопоставим по вычислительной сложности метод, основанный на распространении вероятности в марковской сети [119]. Более сложный метод поиска минимального разреза в специальном потоковом графе, вершины которого соответствуют описанию функции $d(x, y)$ и соединены с источником, стоком и между собой определённым спо-

собом [117, 118], в среднем обладает сложностью $O(N^{1,2}D^{1,3})$, в наихудшем случае их сложность $O(N^2D^2 \log(ND))$.

Для практической оценки эффективности того или иного метода восстановления глубины можно пользоваться источником [121]. Данный источник предоставляет набор пар изображений с заранее сформированной эталонной диспаратностью. Различные методы могут сравниваться на этом наборе. В критерии сравнения входит погрешность определения значений функции $d(x, y)$, полнота восстановления глубины и время обработки кадров. Данный источник показывает, что на данный момент в системах реального времени могут использоваться лишь алгоритмы локального сопоставления блоков, основанные на уравнении (5.8), поскольку остальные алгоритмы обрабатывают кадры за время от одной секунды и более.

5.2. Исследование эффективности алгоритмов выделения движущихся объектов на основе изображений одной видеокамеры

В рамках данной работы уже приводилось описание алгоритмов, которые часто применяются для идентификации и сопровождения подвижных объектов, см. п. 1.3 и п. 2.1. Данные алгоритмы являются достаточно эффективными, если применяются для сцен с невысокой плотностью движения объектов в тех случаях, когда важно не пропустить сам факт наличия движущихся объектов на сцене.

В случае плотного движения на сцене в поле зрения видеокамер (см. пример на рис. 5.4) рассмотренные ранее алгоритмы сопровождения объектов на основе моделирования фонового изображения сцены недостаточно эффективны, поскольку фон постоянно закрыт подвижными объектами и затруднительно построить модель изображения неподвижной части сцены. Тем не менее, область применения систем видеонаблюдения не ограничивается только сферой обеспечения безопасности, часто видеонаблюдение используется для сбора статистики, необходимой для хозяйственной деятельности организаций. Для оценки статистики не столько важна достоверность определения событий, сколько важно с некоторой точностью определить плотность движения и направление потоков движения на видеокамере. Иными словами, в системах сбора статистики показатели Q^α и E^α , которые оценивают точность сопровождения движущихся объектов и рассчитываются по формулам (4.7) параграфа 4.2, важны более, чем количество пропусков и ложных срабатываний $F_N^{\alpha\beta}$ и $F_P^{\alpha\beta}$, рассчитываемых по формулам (4.4) и (4.5), соответственно.

К примеру, на рис. 5.4 в области наблюдения видеокамеры находится вход в кинозал, и для организации контроля продажи билетов важно, сколько людей проходит в зал во время начала сеанса, и сколько людей выходит из зала во время конца сеанса. В рамках диссертационной работы был внедрён метод оценки плотности движения объектов, который позволяет дать оценку количества перемещающихся объектов при



Рисунок 5.4. Пример плотного движения в поле зрения видеокамеры.

помощи одной камеры даже тогда, когда точную модель фона оценить не удаётся. Правда в условиях плотного движения, относительная погрешность оценки количества людей довольно велика, и составляет около 20%, тем не менее, в некоторых случаях при оценке посещаемости достаточно такой точности.

Рассмотрим алгоритм оценки плотности движения с помощью одной видеокамеры.

Перед описанием алгоритма условимся считать, что для видеокамер, на которых мы отслеживаем движущиеся объекты, проведена процедура калибровки, описанная в работе автора [172], т.е. в каждой точке изображения (x, y) нам известен характерный размер объекта $H(x, y)$. Данная функция является входом всех нижеперечисленных алгоритмов.

Систему координат на изображении кадра мы введём таким образом, чтобы единицы координат соответствовали пикселам изображения, координата $(0,0)$ соответствует левому верхнему углу, ось y направлена вниз, ось x — вправо.

Пусть $\{F_i(x, y)\}$ — последовательность изображений, а значениями функций F_i являются интенсивности изображения в точках (x, y) . Данная последовательность, как и функция $H(x, y)$, относится к входным данным для алгоритма отслеживания движения. К выходным данным алгоритма относится множество объектов $\mathbf{A} = \{A_m\}$, где для каждого объекта A_m определено его множество положений $\{P_{im}^a\}$ на i -ом кадре видеопоследовательности, индекс a показывает, что положение объекта определено

автоматически. Само положение объекта определяется координатами минимального по площади прямоугольника со сторонами, параллельными координатным осям. Примером алгоритма обработки видеопоследовательности изображений, достаточно достоверно выделяющего движущиеся объекты в поле зрения одной камеры, при этом обладая невысокой вычислительной сложностью, является следующая последовательность шагов.

1. Осуществим подбор проективного преобразования координат

$$(x', y') = \mathbf{h}(x, y) = (h_x(x, y), h_y(x, y)),$$

такого, что на кадрах $F_i^p(x, y) = F_i(h_x(x, y), h_y(x, y))$ характерный размер изображения объектов в каждой точке примерно равен фиксированному значению $n_h = 16$. Преобразование $\mathbf{h}(x, y)$ представляет собой две дробно-линейные функции

$$\begin{aligned} x' = h_x(x, y) &= \frac{Ax + By + C}{Gx + Fy + 1}, \\ y' = h_y(x, y) &= \frac{Dx + Ey + F}{Gx + Fy + 1}, \end{aligned}$$

где коэффициенты A, B, C, D, E, F, G, H необходимо подобрать таким образом, чтобы в любой точке кадра выполнялось условие

$$h_y(x, y - H(x, y)) - h_y(x, y) \approx n_h.$$

При этом коэффициенты проективного преобразования могут быть заданы заранее с помощью графической утилиты подбора преобразования кадра, либо их можно вычислить, решив следующую задачу

$$\min_{A, B, C, D, E, F, G, H} \sum_{(x, y)} (h_y(x, y - H(x, y)) - h_y(x, y) - n_h)^2,$$

например, методом градиентного спуска. Начальное приближение можно получить с помощью решения несколько упрощённой задачи

$$\min_{A, B, C, D, E, F, G, H} \sum_{(x, y)} (h_y(x, y - H(x, y)) - h_y(x, y) - n_h)^2 \cdot (Gx + Fy + 1)^2,$$

например, методом наименьших квадратов, поскольку в данном случае целевая функция состоит из слагаемых, являющихся квадратами линейных функций относительно пространства, в котором ведётся поиск оптимальных значений.

2. Введём множество объектов T_i , отслеживаемых на кадре i . Элементами множеств T_i будем считать точки t , характеризующие положение объектов. Индекс начального кадра видеопоследовательности мы будем считать единичным, и положим $T_0 = T_1 = \emptyset$. Множество результирующих объектов в начале работы алгоритма положим $\mathbf{A} = \emptyset$.

3. Следующие шаги алгоритма повторяются по порядку для каждого кадра последовательности $\{F_i(x, y)\}$.

4. Кадр $F_i(x, y)$ преобразуется в кадр $F_i^p(x, y)$ с помощью преобразования, полученного на шаге 1.

5. Кадр $F_i^p(x, y)$ подвергается операции Гауссова размытия путём свёртки с функцией $g(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{\frac{x^2+y^2}{2\sigma^2}}$, где $\sigma = \frac{n_h}{3}$. В результате получается кадр $F_i^{pg}(x, y) = g(x, y) * F_i^p(x, y)$.

6. Для кадра $F_i^{pg}(x, y)$ производится поиск точек локальных нестрогих экстремумов, положение которых запоминается с помощью характеристических функций-таблиц $\chi_{\min}(x, y)$ и $\chi_{\max}(x, y)$:

$$\begin{aligned}\chi_{\min}(x, y) &= \begin{cases} 1, & \text{если} \\ & \forall x' \in [x - \frac{n_h}{2}; x + \frac{n_h}{2}] \forall y' \in [y - \frac{n_h}{2}; y + \frac{n_h}{2}] : \\ & F_i^{pg}(x', y') \geq F_i^{pg}(x, y), \\ 0 & \text{в противном случае,} \end{cases} \\ \chi_{\max}(x, y) &= \begin{cases} 1, & \text{если} \\ & \forall x' \in [x - \frac{n_h}{2}; x + \frac{n_h}{2}] \forall y' \in [y - \frac{n_h}{2}; y + \frac{n_h}{2}] : \\ & F_i^{pg}(x', y') \leq F_i^{pg}(x, y), \\ 0 & \text{в противном случае.} \end{cases}\end{aligned}$$

7. С помощью алгоритма волновой раскраски выделяются множества соседних точек, помеченные 1 в области значений функций $\chi_{\min}(x, y)$ и $\chi_{\max}(x, y)$. Множества центров масс выделенных областей обозначим как $R'_i = \{(c_x^r, c_y^r)\}_{r=1}^{n'_r} = \{c^r\}_{r=1}^{n'_r}$.

8. Из множества точек R'_i выделяются точки R_i с наиболее различимой текстурой. Это производится с помощью критерия, введённого в работе [173]: в каждой точке оцениваются градиенты $\nabla F_i^p = \left(\frac{\partial F_i^p}{\partial x}, \frac{\partial F_i^p}{\partial y}\right)$ функции $F_i^p(x, y)$, из них составляется матрица $(\nabla F_i^p)^T \cdot \nabla F_i^p$ размером 2×2 . Для устранения шума в градиентах данная матрица в каждой точке (x, y) усредняется по окрестности

$$\left[x - \frac{n_h}{2}; x + \frac{n_h}{2}\right] \times \left[y - \frac{n_h}{2}; y + \frac{n_h}{2}\right],$$

а затем для усреднённой матрицы рассчитывается минимальное собственное значение. Если это собственное значение больше некоторой величины, значит найденный центр масс точек нестрогого экстремума на изображении $F_i^p(x, y)$ является достаточно существенным, и поэтому легко идентифицируемым на предыдущих и последующих кадрах. Именно эти точки из множества R'_i отбираются во множество R_i .

9. Из элементов множества $T_{i-1} = \{t^o\}$ и элементов множества $R_i = \{c^r\}$ строится соответствующий двудольный граф, каждому ребру (t^o, c^r) сопоставляется вес $w_o^r =$

$\alpha \|t^o - c^r\| + \beta \varepsilon(t^o, c^r)$, где α и β — положительные весовые коэффициенты, $\|t^o - c^r\|$ — расстояние между соответствующими точками, $\varepsilon(t^o, c^r)$ — разность изображений по норме L_2 в окрестности точки (t_x^o, t_y^o) на предыдущем кадре F_{i-1}^p и в окрестности точки (c_x^r, c_y^r) на текущем кадре F_i^p . С помощью метода, описанного в работе [174], решается задача об оптимальном паросочетании

$$\min \sum_{(r,o)} (C - w_0^r),$$

где C — константа, превышающая все веса w_o^r , а $E = \{(r, o)\}$ — множество пар индексов, однозначно определяющих рёбра построенного двудольного графа, входящие в оптимальные паросочетания, это подразумевает что каждый индекс вершин r и o может входить в найденное множество E только один раз. Используя найденное множество E , строится множество новых положений объектов на T_i текущем кадре: если $(r, o) \in E$, то включаем во множество T_i элемент t^o из множества T_{i-1} , при этом его положение обновляем новым положением c^r . Если существуют элементы R_i , которые не входят в рёбра, задаваемые множеством E , то в множество T_i включаются новые объекты, соответствующие расположению областей R_i без пары. Все объекты $t \in T_{i-1}$, которые не обладают парой среди R_i , переносятся в T_i , при условии, что

$$\exists n < n_t : (\text{на кадре } T_{i-n-1} \text{ у объекта } t \text{ была пара в } R_{i-n}).$$

Значение n_t является постоянным параметром, определяющим время сохранения в памяти информации об объекте, измеряемое в количестве кадров видеопоследовательности.

10. Обновляются выходные данные $\mathbf{A} = \{A_m\}$. Информация о вновь появившихся объектах t множества T_i переносится в качестве элементов A_m при условии, что

$$\exists n_0 \forall n < n_b : (\text{на кадре } T_{i-n_0-n-1} \text{ у объекта } t \text{ была пара в } R_{i-n_0-n}),$$

где n_b — постоянный параметр, определяющий длину отрезка последовательности изображений, на каждом кадре которого рассматриваемый объект должен сопоставляться с какой-либо точкой экстремума, чтобы попасть в конечный результат работы алгоритма. Положения объектов отмечаются прямоугольными рамками, получить которые можно с помощью построенного преобразования $\mathbf{h}(x, y)$: для каждого объекта A_m на кадре i известно его текущее положение $t^m = (t_x^m, t_y^m)$. Окрестность положения объекта в координатах кадра $F_i^p(x, y)$ можно определить как $P_{im}''a = [t_x^m - \frac{n_h}{2}; t_x^m + \frac{n_h}{2}] \times [t_y^m - \frac{n_h}{2}; t_y^m + \frac{n_h}{2}]$. Прообраз данного множества при преобразовании $\mathbf{h}(x, y)$ является оценочным положением объекта на исходном кадре $F_i(x, y)$, т.е. $P_{im}^a = \mathbf{h}^{-1}(P_{im}''a)$ определяет четырёхугольник, приближающий положение объекта. В качестве конечного результата, используя координаты вершин P_{im}^a , строится

P_{im}^a — минимальный по площади прямоугольник со сторонами, параллельными осям координат исходного кадра.

Таким образом, получается результат $\mathbf{A} = \{A_m\}$. В системах видеонаблюдения поток обрабатывается непрерывно, поэтому формально представленный метод обработки изображений не удовлетворяет требованию для алгоритмов, по которому любой алгоритм должен состоять из конечной последовательности действий. Однако, если рассматривать конечные подпоследовательности кадров от начала работы до текущего момента, то количество действий также будет конечным, поэтому термин алгоритм применять в данном случае корректно. Представленный алгоритм работает в реальном времени, после обработки каждого кадра у него на выходе есть текущий результат с траекториями объектов. Для краткости дальнейших ссылок на данный алгоритм, обозначим его \mathcal{A}_{mono} .

Алгоритм \mathcal{A}_{mono} с высокой достоверностью сопровождает объекты при малой плотности движения не более одного объекта в течение 10 секунд в поле зрения камеры. В этом случае показатели $F_N^{\alpha\beta}$ и $F_P^{\alpha\beta}$ (см. параграф 4.2) не выше 5% по отношению к общему числу объектов в кадре. Однако данные ошибки значительно возрастают до 20% при увеличении потока движения до нескольких объектов в секунду (подобный случай изображён на рис. 5.4). В рамках данной работы установлено, что использование стереоэффекта позволяет существенно повысить достоверность детектирования подвижных объектов. Далее будет приведён анализ разработки систем выделения движущихся объектов в стереосистемах.

5.3. Разработка алгоритмов синхронизации кадров стереокамер на основе теории графов

В настоящее время стереосистемы, т.е. системы, состоящие из двух видеокамер, которые расположены на известном расстоянии друг от друга, находят всё большее применение. Например, весьма перспективные автоматизированные системы подсчёта количества людей, вошедших и вышедших из помещения, аналогичные [175], основаны на алгоритмах восстановления глубины по стереоизображению, например, см. [111, 112, 113, 114, 115, 116, 117, 118, 119, 120]. Прежде, чем рассмотреть алгоритм отслеживания движущихся изображений с помощью стереокамер, рассмотрим решение одной из проблем применения алгоритмов восстановления глубины, а именно: обеспечение синхронности изображений, полученных видеокамерами, см. статью автора [176].

С одной стороны, нет большой технической проблемы в том, чтобы синхронизировать захват изображения на ПЗС матрицах, если обе видеокамеры специально спроектированы и сконструированы с возможностью подобной синхронизации. С другой стороны, в настоящее время существует немного таких устройств, и все они в десятки

раз дороже, чем пара обычных видеокамер. И если цель производителя заключается в сборе недорогой стереосистемы (особенно в рамках проведения предварительных экспериментов), необходимо использовать стереосистемы с несинхронным захватом изображений в видеокамерах. Даже если частота захвата кадров на двух камерах стереосистемы одинакова, всё равно возможна некоторая рассинхронизация, как из-за разности во времени старта захвата видеопотока, так и из-за погрешности генерирующего частоту захвата кварца.

В рамках данной работы проведено модельное тестирование влияния рассинхронизации на достоверность алгоритма подсчёта посетителей на основе стереоизображений. Для этого искусственно в снятые изображения перед обработкой вводилось отставание кадров видеопотоков, на 1 и на 2 кадра. Если точность детектирования движущихся объектов при синхронной обработке была на уровне 95%, то при сдвиге на кадр (что соответствует 40мс при частоте 25к/с) результат был 87%, при сдвиге на 2 кадра (80мс) – 72%.

Данный тест показывает необходимость устранять, насколько это возможно, задержку синхронизации, обусловленную как аппаратными причинами, так и несовершенством программного обеспечения для захвата видеоданных в операционных системах. Для обеспечения синхронности в дальнейших исследованиях обработка стерео потоков видео построена в виде трёх асинхронных процессов. Первые два процесса асинхронно формируют две очереди из кадров

$$\{(F_{n-i+1}^l, t_{n-i+1}^l), (F_{n-i+2}^l, t_{n-i+2}^l), \dots, (F_n^l, t_n^l)\},$$

$$\{(F_{m-j+1}^r, t_{m-j+1}^r), (F_{m-j+2}^r, t_{m-j+2}^r), \dots, (F_m^r, t_m^r)\},$$

где F_s^\bullet — кадры левой или правой камеры, а t_s^\bullet — соответствующее данному кадру время захвата. Имеет смысл поддерживать такую длину очереди, чтобы время между первым и последним кадром было раза в два выше, чем возможная задержка времени поступления кадра между моментом захвата матрицы и моментом записи в очередь. В современных системах, основанных на сетевых камерах, это время может быть достаточно большим, порядка одной секунды.

Третий процесс заключается выборе наиболее подходящих пар кадров из очередей, формируемых остальными процессами, и отсылке их на обработку алгоритма анализа изображений на основе восстановления глубины. Для выбора оптимальной пары построим двудольный граф $G = \{V, E\}$, $V = V^r \cup V^l$, см. рис. 5.5, в котором вершинам соответствуют кадры из сформированной очереди, и рёбрами соединены только кадры от левой камеры V^r и правой камеры V^l , причем такие, что разность времени захвата между ними не более определённого порога, например, $T_{\max} = 50\text{мс}$. Также мы определим вес для каждого ребра $e = (F_p^l, F_s^r) \in E$ как $W_e = T_{\max} - |t_p^l - t_s^r|$,

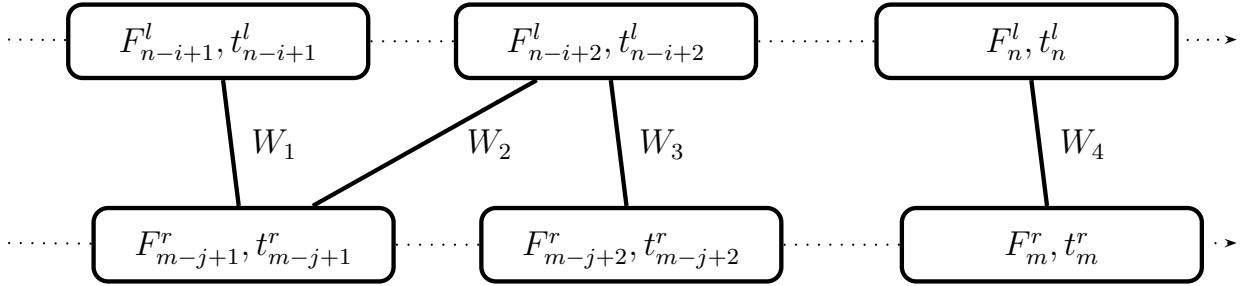


Рисунок 5.5. Двудольный граф из асинхронных кадров, веса рёбер равны $W_1 = T_{\max} - |t_{n-i+1}^l - t_{m-j+1}^r|$, $W_2 = T_{\max} - |t_{n-i+1}^l - t_{m-j+1}^r|$, $W_3 = T_{\max} - |t_{n-i+2}^l - t_{m-j+2}^r|$, $W_4 = T_{\max} - |t_n^l - t_m^r|$.

т.е. чем ближе кадры по времени, тем больше вес. Далее необходимо решить задачу о максимальном парасочетании, т.е. надо найти подмножество $E' \subseteq E$, которое решает задачу

$$\min \sum_{e \in E'} W_e. \quad (5.9)$$

Для задачи (5.9) известно эффективное решение с помощью метода Эдмондса [177]. На данный момент известен алгоритм решения задачи (5.9), полученный как развитие метода «цветков», асимптотическая сложность которого $O(|V|(|E| + \ln |V|))$. Это позволяет выполнять поиск оптимальных парасочетаний в реальном времени. Получив решение E' для задачи (5.9), процесс отбора пар должен послать на обработку ту пару картинок, которая соответствует паре из E' с наименьшим временем захвата одного из кадров среди всех кадров в парах E' . После отбора пары по этому условию и обработки кадров из этой пары в алгоритме, процесс выбора пар должен удалить данные кадры из очередей.

Таким образом достигается синхронизация кадров, что существенно влияет на качество обработки изображений в стереосистемах. Если нет аппаратной поддержки синхронизации, можно получить некоторое приближение к синхронной обработке с помощью алгоритма, предложенного автором. Алгоритм позволяет обеспечить разность во времени захвата обрабатываемых пар кадров не выше, чем половина интервала времени между захватом изображений в двух камерах, что позволяет сохранить уровень достоверности конечного результата, основанного на вычислении глубины.

5.4. Разработка и анализ алгоритмов выделения движущихся объектов с помощью стереосистем

Алгоритм отслеживания движущихся объектов с помощью стереоизображений можно построить аналогично \mathcal{A}_{mono} (см. п. 5.2). На входе алгоритма, кроме функции характерных размеров объектов $H(x, y)$, полученных так, как описано в [172], также задаётся высота подвеса L для пары камер, угол наклона оптических осей к

наблюдаемой сцене γ , и калибровочные параметры стереопары, полученные, например, способом, описанным в [101]. К калибровочным параметрам относятся: смещение \vec{b} от одной камеры к другой, матрица поворота R , позволяющая привести положение оптической оси одной камеры к такому положению, при котором она параллельна положению второй камеры, векторы коэффициентов устранения радиальной дисторсии D_1 и D_2 объективов стереопары, матрицы M_1 и M_2 , состоящие из внутренних параметров камер, характеризующие фокусное расстояние и разрешение изображений. Кроме последовательности кадров $\{F_i(x, y)\}$ с левой камеры на вход алгоритмов поступает последовательность соответствующих кадров с $\{F'_i(x, y)\}$ с правой камеры.

Отслеживание движущихся объектов с помощью стереоизображений состоит из следующих шагов.

1. Повторяется шаг 1 алгоритма \mathcal{A}_{mono} , и находится преобразование $\mathbf{h}(x, y)$, приводящее все изображения объектов к изображению с характерным размером $n_h = 16$.
2. Проводится предварительная инициализация множеств T_i и \mathbf{A} аналогично шагу 2 алгоритма \mathcal{A}_{mono} .
3. Дальнейшие шаги повторяются для каждой пары кадров последовательностей $\{F_i(x, y)\}$ и $\{F'_i(x, y)\}$. В некоторых случаях требуется выбор синхронных пар кадров, как это описано в работе [176] и в п. 5.3.
4. На основании параметров \vec{b} , R , D_1 , D_2 , M_1 , M_2 проводим преобразование ректификации изображений, т.е. преобразуем $F_i(x, y)$ и $F'_i(x, y)$ в, соответственно, ректифицированные $F_i^r(x, y)$ и $F'_i^r(x, y)$. Далее либо с помощью метода [112], либо с помощью более качественного, но и более вычислительно сложного метода [116] на основе $F_i^r(x, y)$ и $F'_i^r(x, y)$ производится расчёт диспаратностей $F_i^d(x, y)$, т.е. расчёт смещений в каждой точке левого изображения к соответствующей точке правого изображения. Поскольку известны параметры сцены L и γ , то на основании $F_i^d(x, y)$ производится расчёт высот $F_i^h(x, y)$ наблюдаемых объектов на основе трёхмерной реконструкции сцены. Наконец, с помощью преобразования, полученного на шаге 1, получаем из изображения $F_i^h(x, y)$ изображение $F_i^p(x, y)$ с компенсацией проективных искажений размеров объектов.
5. Далее повторяются шаги 5–10 алгоритма \mathcal{A}_{mono} , за исключением шага 8, поскольку для карты высот дополнительной фильтрации максимумов проводить не требуется.

Приведённый алгоритм будем обозначать \mathcal{A}_{stereo} . На основе алгоритмов \mathcal{A}_{mono} и \mathcal{A}_{stereo} возможно создать комбинированный алгоритм \mathcal{A}_{comb} , приведённый ниже.

1. На последовательности кадров $\{F_i(x, y)\}$ запускается \mathcal{A}_{mono} . В результате получается набор объектов \mathbf{A}^{mono} .

	$F_P^{\alpha\beta}$ (ложные)	$F_N^{\alpha\beta}$ (пропуски)	Q^α (доля кадров с сопровождением)	E^α (смена траектории)
\mathcal{A}_{mono}	117 из 1080	150 из 1039	75%	187
\mathcal{A}_{stereo}	40 из 1020	35 из 1039	87%	63
\mathcal{A}_{comb}	35 из 1021	23 из 1039	90%	67

Таблица 5.1. Сравнение результатов работы алгоритмов. Результаты приведены для значений $\alpha = 0,3$ и $\beta = 0,5$.

2. На последовательности кадров $\{F_i(x, y)\}$ и $\{F'_i(x, y)\}$ запускается \mathcal{A}_{stereo} . В результате получается набор объектов \mathbf{A}^{stereo} .

3. Конечный результат \mathbf{A} формируется на основе \mathbf{A}^{mono} и \mathbf{A}^{stereo} по следующему принципу: в множество \mathbf{A} попадают те объекты из \mathbf{A}^{mono} и \mathbf{A}^{stereo} , которые достаточно долго, на протяжении n_I кадров подряд, были близки по положению. Если на текущем кадре существует два почти одинаковых по положению объекта из \mathbf{A}^{mono} и \mathbf{A}^{stereo} , то в конечный результат включается только положение одного объекта из \mathbf{A}^{stereo} . Приоритет отдан алгоритму \mathcal{A}_{stereo} , поскольку, как показано далее, он работает более достоверно.

Для сравнения эффективности алгоритмов \mathcal{A}_{mono} , \mathcal{A}_{stereo} и \mathcal{A}_{comb} необходимо провести анализ их работы на одинаковых последовательностях изображений, и сравнить результаты с помощью специальных методик, аналогичных приведённым в работах [166, 167, 178] и в параграфе 4.2. Результаты для данных трёх алгоритмов, приведённых выше, представлены в таблице 5.1. Видно, что по качеству сопровождения и достоверности \mathcal{A}_{stereo} значительно опережает \mathcal{A}_{mono} , а \mathcal{A}_{comb} показывает несколько лучшие результаты по сравнению с \mathcal{A}_{stereo} , за исключением показателя E^α .

Таким образом, дополнительный признак изображений, а именно — глубина, получаемая с помощью стереокамер, позволяет существенно улучшить результаты сопровождения движущихся объектов. Тривиальная комбинация подходов к сопровождению объектов с помощью стерео- и обычных изображений позволяет дополнительно улучшить приведённые показатели.

5.5. Исследование способов построения алгоритмов выделения движущихся объектов одиночными видеокамерами и стереосистемами

Алгоритмы \mathcal{A}_{mono} , \mathcal{A}_{stereo} и \mathcal{A}_{comb} , рассмотренные в предыдущих параграфах, применимы для камер, ракурс наблюдения которых достаточно специфичен: наиболее эффективно алгоритмы подсчёта движущихся объектов работают тогда, когда камеры расположены непосредственно над потоком движения. При наблюдении за движущимися объектами со стороны эффективность всех трёх представленных алгоритмов снижается. Далее проводится исследование и анализ возможных способов применения

стереосистем для детектирования движущихся объектов со стороны.

Проблема слежения за движущимися объектами с помощью видеокамер изучается уже давно. Как правило, слежение за объектами в системах реального времени построено на одном из методов поиска положения объекта на текущем кадре на основе известного положения этого объекта на предыдущем кадре. Среди наиболее используемых можно отметить следующие методы поиска:

- 1) по наибольшему совпадению изображений текущего и предыдущего кадра по норме L_2 [75], в частности, поиск нового положения по методу «mean shift» [180];
- 2) выделение контуров с последующим применением метода активных контуров [181];
- 3) фиксация изменяющихся положений особых точек изображения [182];
- 4) сравнение похожих сегментированных областей [183], в том числе, полученных с помощью вычитания фона.

Для каждого из приведённых выше методов существует много модификаций, связанных с оптимизацией вычислительной сложности или применением различных алгоритмов, например, алгоритма вычисления оптического потока по методу Лукаса-Канаде. Более того, существуют разработки, объединяющие несколько методов поиска и слежения, позволяющие получить более точное сопровождение объектов. Среди подобных работ наиболее интересна, на наш взгляд, статья [179], в которой предложено на основе обучения на большой выборке видеоданных выбирать метод вычисления следующего положения объекта, исходя из априорной оценки точности, полученной в процессе обучения для того или иного метода.

С одной стороны, приведённые выше наработки показывают хорошую достоверность при идентификации и сопровождении движущихся объектов, однако они обладают существенным недостатком: сложность данных алгоритмов не всегда позволяет обрабатывать изображения в реальном времени с помощью современных вычислительных средств. Хорошие результаты достигаются при обработке записанных видеопоследовательностей, причём частота обработки кадров существенно ниже частоты выдачи кадров камерами. Поэтому в рамках данной работы для выделения движущихся объектов используется алгоритм вычитания фона, рассмотренный в рамках обзора в п. 1.3. Простота данного метода позволяет реализовать его с помощью вычислительно-эффективных инструкций и микропрограмм современных ЭВМ. Дополнительные фильтры позволяют снизить число ошибок первого и второго рода до приемлемых уровней (одно ложное срабатывание в неделю на одной камере при детектировании 99,5% объектов, удовлетворяющих определённым паспортным требованиям). Однако и этого уровня в некоторых случаях бывает недостаточно с точки зрения требований к системе в конкретных условиях эксплуатации. В таких случаях

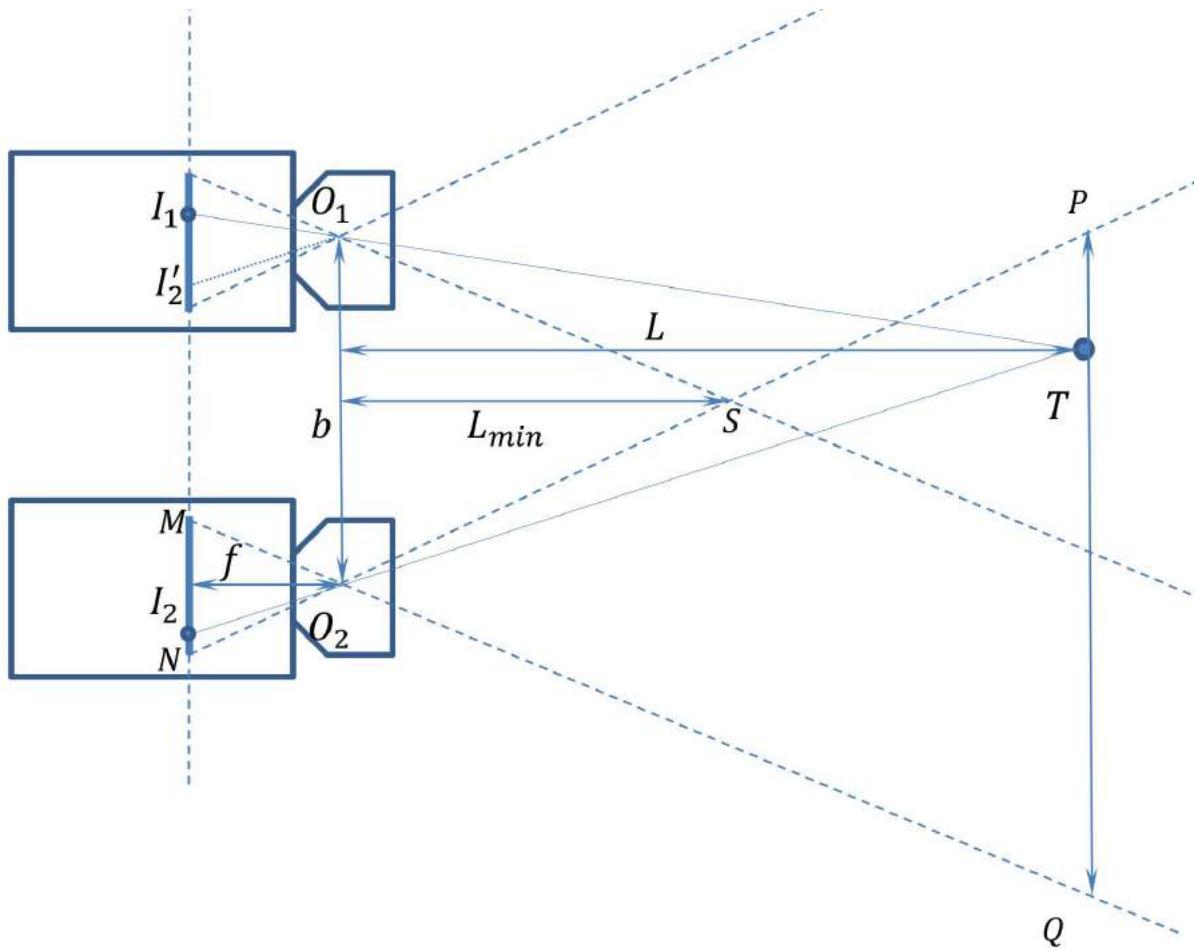


Рисунок 5.6. Схема расчёта диспаритности в стереосистемах.

возможно достигнуть качественного улучшения показателей достоверности за счёт использования стереосистем вместо моновидеокамер.

Рассмотрим границы применимости стереосистем при детектировании движущихся объектов, для определённости людей. Схема стереосистемы в горизонтальном разрезе представлена на рис. 5.6.

Рассматриваемая стереосистема состоит из двух одинаковых камер с совмещёнными фокальными плоскостями, причём стороны ПЗС-матриц камер параллельны друг другу. Таким образом, оптические оси камер параллельны. Введём пояснения к обозначениям на рисунке, а также некоторые новые обозначения:

- O_1 и O_2 – оптические центры, соответственно, левой и правой камеры,
- $b = |O_1O_2|$ – «база» стереосистемы,
- T – положение наблюдаемого в камеры стереосистемы объекта,
- L – расстояние от стереопары до наблюдаемого объекта,
- I_1 и I_2 – изображение объекта T на левой и правой камерах, соответственно,

- I'_2 – положение точки I_2 после параллельного переноса точки O_2 к точке O_1 так, чтобы оптические оси камер совпали,
- обозначим $= |I_1 I'_2|$ – диспаратность для изображений объекта T ,
- S – ближайшая к стереопаре точка, находящаяся на расстоянии L_{\min} от стереопары и попадающая в поле зрения обоих камер,
- M и N – края ПЗС-матрицы правой камеры, т.е. $W = |PQ|$ – ширина ПЗС-матрицы
- P и Q – края поля зрения правой камеры на расстоянии L , то есть $r = |PQ|$ – ширина зоны обзора камеры на расстоянии L ,
- f – фокусное расстояние стереокамер, которое вместе с шириной матрицы определяет угол обзора камеры $2 \arctg \frac{f}{2W}$.
- ω – горизонтальное разрешение ПЗС-матриц, и, соответственно, кадров стереокамер.

Основная формула для восстановления глубины следует из подобия треугольников $I_1 I'_2 O_1$ и $O_1 O_2 T$:

$$\frac{\Delta}{f} = \frac{b}{L} \implies L = \frac{bf}{\Delta}.$$

Величина диспаратности измеряется в метрических единицах. Чтобы перейти к диспаратности d , измеряемой в точках кадра, необходимо воспользоваться соотношением $\frac{\Delta}{W} = \frac{d}{\omega}$, то есть

$$L = \frac{\omega bf}{dW}, \quad (5.10)$$

данную формулу будем называть основной.

Практически в каждой задаче имеет значение ширина зоны обзора стереосистемы. Из подобия треугольников MNO_2 и PQO_2 следует

$$\frac{f}{W} = \frac{L}{r} \Rightarrow r = \frac{LW}{f}, \quad (5.11)$$

данную формулу будем называть формулой ширины зоны обзора. При оценке ширины обзора стереопары следует учитывать, что зона обзора стерео находится в пересечении зон обзора левой и правой камеры, поэтому из ширины зоны обзора одной камеры необходимо вычесть базу b , однако в большинстве случаев этим можно пренебречь.

Кроме ширины зоны обзора, решающее значение в задачах, основанных на восстановлении глубины с последующей 3d-реконструкцией, имеет точность измерений.

При фиксированных параметрах ω, b, f, W из основной формулы следует $Ld = \text{const}$, т.е. $L\partial d = -d\partial L$, и значит

$$|\partial L| = \frac{L |\partial d|}{d} = \frac{L^2 W |\partial d|}{\omega b f}, \quad (5.12)$$

данную формулу будем называть формулой погрешности восстановления глубины.

Из подобия треугольников O_1O_2S и MNO_2 следует, что $L_{\min} = \frac{bf}{W}$. Однако большинство алгоритмов расчёта диспаратности обладают ограничением на максимальную диспаратность d_{\max} (например, довольно часто диспаратность ограничивается значением $d_{\max} = 256$), поэтому минимальное расстояние, на котором стереосистема может оценивать глубину, определяется с помощью основной формулы 5.10:

$$L'_{\min} = \frac{\omega bf}{d_{\max} W}, \quad (5.13)$$

данную формулу будем называть формулой расчёта ближнего края рабочей зоны. Заметим, что при $d_{\max} = \omega$, т.е. при фактическом отсутствии ограничения на максимальную диспаратность, $L'_{\min} = L_{\min}$.

Рассмотрим доступные диапазоны параметров стереопары.

Разрешение ПЗС-матрицы ω может быть одним из следующих значений: 160, 320, 352, 640, 704, 720, 1280, 1920. Более высокие значения пока не поддерживаются современными вычислительными средствами расчёта диспаратности реального времени. С одной стороны, чем меньше данное разрешение, тем ближе к стереопаре рабочая зона согласно формуле расчёта для L'_{\min} . С другой стороны, чем разрешение больше, тем выше погрешность оценки расстояния $|\partial L|$ согласно формуле (5.12) восстановления погрешности глубины.

Фокусное расстояние $f > 2,8\text{мм}$ (при меньших фокусных расстояниях возникают большие оптические искажения), а размер матрицы W определяется согласно таблице 5.2.

Наиболее распространены на данный момент матрицы $1/3''$ и $1/4''$, т.е. фактически выбор ширины матрицы ограничен между двумя значениями — 4,8мм и 3,6мм. Отношение $\frac{W}{f}$ входит и в формулу расчёта ширины зоны обзора, и формулу расчёта погрешности, и в формулу расчёта ближнего края рабочей зоны. Чем больше отношение, тем ближе край рабочей зоны, тем шире зона обзора, но, с другой стороны, тем выше погрешность восстановления глубины.

Переходя к задаче распознавания движущихся людей на расстоянии порядка десятка и менее метров от стереокамеры, приходим к необходимости выбирать параметры так, чтобы отношение $\frac{W}{f}$ было как можно больше: ограничения на ширину зоны обзора на расстоянии нескольких метров от камеры получаются более строгие, чем на погрешность, тем более погрешность можно «отрегулировать» разрешением

Диагональ в видиконовых дюймах	Размер в мм
13/8" (плёнка типа 135)	36 × 24
APS-H	28,1 × 18,7
APS-C	23,7 × 15,6
4/3"	17,3 × 13,0
1"	12,8 × 9,6
2/3"	8,8 × 6,6
1/1,8"	7,2 × 5,3
1/2"	6,4 × 4,8
1/2,3"	6,16 × 4,62
1/2,33"	6,08 × 4,56
1/2,5"	5,8 × 4,3
1/2,7"	5,4 × 4,0
1/3"	4,8 × 3,6
1/3,2"	4,54 × 3,42
1/3,6"	4 × 3
1/4"	3,6 × 2,7
1/6"	2,4 × 1,8

Таблица 5.2. Таблица соответствия ПЗС-матриц различных стандартов их геометрическим размерам.

камер и стереобазой. Чтобы отношение было максимальным, необходимо выбирать минимальное фокусное расстояние $f \approx 3\text{мм}$, и наибольшую по ширине ПЗС-матрицу из доступных, а именно $1/4"$, $W = 4,8\text{мм}$. Получим $\frac{\omega W}{f} = 1,6$, и $r = 1,6L$ по формуле (5.11).

Произведение разрешения и стереобазы ωb участвует и в формуле погрешности (5.12), и в формуле расчёта ближнего края (5.13). Поэтому при выборе разрешения и стереобазы возможны некоторые вариации в пределах заданного ограничения $b < 30\text{см}$. Поскольку $d_{\max} = 256$, то при максимальном $\omega = 1920$ из формулы расчёта ближнего края получаем $L'_{\min} = \frac{1920}{256 \cdot 1,6} b \approx 4,7b$. С другой стороны, для лучших методов расчёта диспаратности $|\partial d| \approx 2 \div 3$ пикселя (см. [121]). Поэтому из формулы расчёта погрешности восстановления глубины следует, что на предельном по дальности в рамках поставленной задачи расстоянии $L = 10\text{м}$ $|\partial L| = \frac{100\text{м}^2 \cdot 3}{1920 \cdot 1,6b} \approx \frac{1000\text{см}^2}{b}$. Для того, чтобы фиксировать приближение человека, необходимо $|\partial L| < 33\text{см}$, т.е. несколько меньше характерной ширины человека. Отсюда $b = 30\text{см}$. Тогда $L'_{\min} \approx 1,5\text{м}$ — ближайшее расстояние для рабочей зоны. Для фиксации приближения человека с «квантом» чувствительности 1,5м достаточно $b = 6\text{см}$, и тогда $L'_{\min} \approx 30\text{см}$, т.е. ограничения



Рисунок 5.7. Пример изображения широкоугольной камеры кругового обзора.

почти нет. Однако необходимо помнить ещё про то, что ширина зоны наблюдения на ближнем плане $r = 1,6L'_{\min} \approx 50\text{см}$ будет при этом неприемлемо маленькой, и в действительности рабочая зона с шириной 3–4м начнётся с расстояния 2–3м.

Таким образом, при разрешении 1920 нужна стереобаза 30см, при 1280 — 45см, при 1024 — 60см, при 720 — 80см.

Заметим, что для устранения ограничения (5.11) следует использовать сферические ПЗС-сенсоры, чтобы, с одной стороны, изображение камер было примерно таким, как изображение на рис. 5.7, но, с другой стороны, не было бы потери оптического разрешения на краях (в случае плоских сенсоров такие потери неизбежно будут). После вырезания нужных участков изображений, устранения дисторсии и ректификации, фактически получаются изображения с нескольких стереопар сразу, и ограничения по ширине рабочей зоны будут сняты за счёт того, что сцена будет закрыта несколькими стереопарами.

Алгоритм детектирования движущихся объектов с повышенной достоверностью при соблюдении приведённых выше ограничений на область наблюдения строится следующим образом: любая движущаяся область считается объектом только тогда, когда изменяется глубина точек объекта на величину, превышающую погрешность определения расстояния от объекта до камеры, см. формулу (5.12). Данный алгоритм более достоверно детектирует движущиеся объекты, чем исходный алгоритм сопровождения движущихся объектов с помощью вычитания фона, однако вычислительная сложность алгоритма расчёта глубины велика, и для работы в реальном

времени необходимы специальные аппаратные ускорители расчёта карты глубины, которые на данный момент не являются широко доступными. Поэтому есть необходимость в разработке новых более быстрых алгоритмов расчёта глубины для детектирования подвижных объектов.

5.6. Исследование способов повышения быстродействия алгоритмов обработки стереоизображений за счёт семантического сжатия

Для оценки расстояния до объектов наиболее часто используются алгоритмы восстановления глубины [114, 115, 116], работающие с одной стороны достаточно быстро, а с другой стороны обладающие приемлемой погрешностью расчёта диспаритетности. Данные алгоритмы базируются на принципе динамического программирования при расчёте минимума целевой функции $E(x, y)$:

$$\begin{aligned} \min_{d(x,y)} E(x, y) = & \min \sum_{(x,y)} (C(x, y, d(x, y)) + \\ & + \sum_{(u,v) \in U(x,y)} P_1 T(|d(x, y) - d(u, v)| = 1) + \\ & + \sum_{(u,v) \in U(x,y)} P_2 T(|d(x, y) - d(u, v)| > 1)), \quad (5.14) \end{aligned}$$

где $d(x, y)$ — диспаритетность, или величина смещения от точки (x, y) на левом изображении $F^l(x, y)$ до соответствующей ей точки $F^r(x, y)$ на левом изображении (т.е. $F^l(x, y) \approx F^r(x + d(x, y), y)$), $C(x, y, d(x, y))$ — функция штрафа в точке (x, y) за несовпадение изображений $F^l(x, y)$ и $F^r(x, y)$ в окрестностях (x, y) и $(x + d(x, y), y)$, соответственно, функция $T(\cdot)$ является функцией над предикатами, и равна 1, если условие предикат в аргументе функции T выполняется, и 0, если предикат не верен, $U(x, y)$ — некоторая окрестность (обычно квадратная) точки (x, y) , а P_1 и P_2 — постоянные коэффициенты, регулирующие вклад в штраф за невыполнение условий плавного изменения функции $d(x, y)$ в окрестности $U(x, y)$.

Стоимость несовпадения $C(x, y, d(x, y))$ в целевой функции (5.14) определяется разными способами. В одной из популярных библиотек обработки изображений [102] используется функция штрафа следующего вида:

$$\begin{aligned} F_r^+(x, y) &= \frac{1}{2} (F_r(x, y) + F_r(x + 1, y)), \quad F_r^-(x, y) = \frac{1}{2} (F_r(x, y) + F_r(x - 1, y)), \\ F_{\min}(x, y) &= \min \{F_r^+(x, y), F_r^-(x, y), F_r(x, y)\}, \\ F_{\max}(x, y) &= \max \{F_r^+(x, y), F_r^-(x, y), F_r(x, y)\}, \\ C(x, y, d(x, y)) &= \max \{0, F_l(x - d(x, y)) - F_{\max}(x, y), F_{\min}(x, y) - F_l(x - d(x, y), y)\}. \end{aligned}$$

Обоснование для выбора именно такой меры несовпадения см. в содержании статьи [184].

В общем виде задача (5.14) является *NP*-полной, и в настоящее время для таких задач не известны решения полиномиальной сложности, т.е. задачу (5.14) приходится решать полным перебором допустимых значений $d(x, y)$ во всех точках изображения, что является неприемлемым для систем реального времени, поскольку вычислительная сложность такого решения экспоненциальна:

$$O(e^{WH \ln d_{\max}}),$$

где $W \times H$ — ширина и высота кадров F_l и F_r , а d_{\max} — максимальная диспаратность при поиске. Однако можно использовать приближённое решение задачи (5.14), при котором мы считаем, что приближённо минимум штрафной функции можно рассчитать вдоль одного или нескольких прямых путей на изображении, тем самым сужая окрестность $U(x, y)$ в исходной целевой функции до некоторого интервала. В результате, если мы минимизируем функцию $E(x, y)$ только вдоль прямой с направляющим вектором $\mathbf{r} = (r_x, r_y)$, компоненты которого целые и не превосходят по модулю единицу, функция $E(x, y)$ трансформируется в функцию $E_{\mathbf{r}}(x, y)$, и для неё справедливы следующие рекуррентные соотношения, которые позволяют применить схему динамического программирования:

$$\begin{aligned} E_{\mathbf{r}}(x, y) = & C(x, y, d(x, y)) + \\ & + \min \{ E_{\mathbf{r}}(x - r_x, y - r_y, d(x, y)), \\ & E_{\mathbf{r}}(x - r_x, y - r_y, d(x, y) - 1) + P_1, \\ & E_{\mathbf{r}}(x - r_x, y - r_y, d(x, y) + 1) + P_1, \\ & \min_{i \in [0, d_{\max}]} E_{\mathbf{r}}(x - r_x, y - r_y, i) + P_2 \}. \end{aligned}$$

Чтобы устранить постоянный рост штрафов вдоль вектора \mathbf{r} , вместо функции $E_{\mathbf{r}}(x, y)$ используют функцию

$$E'_{\mathbf{r}}(x, y) = E_{\mathbf{r}}(x, y) - \min_{k \in [0, d_{\max}]} E_{\mathbf{r}}(x - r_x, y - r_y, k),$$

которая формирует такое же решение $d(x, y)$, при этом позволяет избежать использования арифметики с высокой разрядностью. Для устранения шума на краях объектов конечная функция штрафов рассчитывается по формуле

$$\tilde{E}(x, y) = \sum_{\mathbf{r} \in \mathcal{R}} E'_{\mathbf{r}}(x, y), \quad (5.15)$$

где \mathcal{R} — набор лучей, вдоль которых рассчитывается штраф в каждой точке. Сложность расчёта диспаратности при минимизации функции $\tilde{E}(x, y)$ равна

$$O(WHd_{\max}|\mathcal{R}|),$$

где $|\mathcal{R}|$ — количество направлений, участвующих в расчёте штрафов. Эта сложность полиномиальна относительно точек изображения, но коэффициент $d_{\max}|\mathcal{R}|$ может быть достаточно велик, и современные вычислительные устройства, как правило, не могут вести расчёт диспаратности в реальном времени с помощью данного алгоритма. Однако возможно применить принцип семантического сжатия для ускорения подсчёта диспаратностей.

Для этого и на правом, и на левом изображении мы можем выделить подвижные точки движения и фоновые изображения $B_r(x, y)$ и $B_l(x, y)$ для правой и левой камеры с помощью алгоритмов выделения движения, описанных в п. 1.3.

Для фоновых изображений с некоторой периодичностью, например, раз в 5 секунд, необходимо расчитывать диспаратность $d_B(x, y)$. Для расчёта диспаратности текущего кадра используется следующий принцип: если точка неподвижна, то $d(x, y) = d_B(x, y)$. Если в точке есть движение, то расчёт диспаратности ведётся по основному алгоритму с вычислением функции стоимости (5.15), но при этом в процессе динамического программирования используются промежуточные таблицы, задействованные при расчёте диспаратности на фоновых изображениях. Таким образом, вычисления будут выполняться только в точках с движением, и поскольку, как правило, доля точек движения в кадре невелика, то и вычислительная нагрузка при расчёте диспаратности $d(x, y)$ существенно снизится, и сможет быть выполнена в реальном времени.

Выводы

1. Проведённое исследование видеосистем показывает, что на данный момент возможности восстановления глубины с помощью стерео видеокамер практически не используются, поскольку большинству вычислителей не хватает производительности для расчёта диспаратности с приемлемым качеством.
2. Разработаны алгоритмы сопровождения движущихся объектов \mathcal{A}_{mono} , \mathcal{A}_{stereo} и \mathcal{A}_{comb} на основе семантического анализа изображения одной камеры, семантической обработки изображений стереокамер и комбинированный алгоритм, соответственно.
3. Проведён анализ эффективности разработанных алгоритмов, который показал полезность применения карты глубины для решения задач семантической обработки изображений: достоверность детектирования объектов возрастает с 75% до 90% в условиях плотного движения в кадре.
4. Разработан алгоритм выбора синхронных изображений для стереосистем с камерами с асинхронным режимом выдачи изображений. Данный алгоритм существенно улучшает достоверность детектирования объектов с помощью стереоизображений полученных от видеокамер без синхронизации.

5. Разработан алгоритм достоверного детектирования объектов, переходящих допустимые границы, основанный на автоматической оценке расстояния до объекта с помощью стереосистем. Исследованы пределы применимости данного алгоритма, связанные с оптическими характеристиками приборов видеонаблюдения.
6. Разработан способ ускорения вычислений карты глубины при анализе потока изображения видеокамер с высоким разрешением на основе принципа семантического сжатия.

6. Исследование алгоритмов обработки панорамных изображений, получаемых автоматически управляемыми поворотными видеокамерами в режиме постоянного сканирования зоны обзора

Обычно управляемые поворотные видеокамеры используются в системах видеонаблюдения как неподвижные камеры, несмотря на то, что многие из них обладают режимом сканирования по определённой, задаваемой программно, траектории. Обусловлено это сложностью восприятия информации постоянно движущегося изображения: как правило, лица, эксплуатирующие видеосистемы с поворотными камерами в составе, делают выбор в пользу какой-либо фиксированной сцены в поле зрения камеры, и лишь во время специальных мероприятий управляют видеокамерами на поворотных устройствах, сопровождая какой-нибудь объект наблюдения в ручном режиме.

Идея использовать непрерывное сканирование сцены с помощью горизонтального вращения поворотной видеокамеры, отображая при этом панорамное изображение, лежит на поверхности. Панорамное изображение позволяет избежать неудобств, связанных с наблюдением за происходящими на сцене событиями, поскольку сама панорама в процессе движения камеры отображает неподвижную сцену, изменения наблюдаются только в области, которые соответствуют текущему положению камеры, причём изменения связаны только с изменениями сцены, а не с движением камеры.

Однако построение панорамы из последовательности изображений — вычислительно сложная задача. Известная библиотека обработки изображений OpenCV содержит модуль построения панорамы [185], но для построения панорамы из 10 кадров программе из этой библиотеки требуется порядка 10 секунд, что неприемлемо для систем реального времени. В рамках данной главы приведено описание исследований и разработок, которые позволили на два порядка ускорить построение панорамного изображения для видеокамер в режиме сканирования. Это позволяет применять алгоритм построения панорамы в реальном времени для видеопотоков с частотой 25 кадров в секунду, т.е. обработка одного кадра выполняется менее, чем за $1/25\text{с} = 40\text{мс}$.

6.1. Современные алгоритмы построения панорамных изображений из набора цифровых снимков, сделанных из одной точки в различных направлениях

Построение панорамных изображений необходимо тогда, когда из разрозненного набора изображений, содержащих одну и ту же сцену в разных ракурсах, требуется построить единственное изображение, которое агрегирует информацию с исходного набора изображений.

На данный момент существуют программные библиотеки, которые способны строить панорамные изображения [185]. Обычно программный код построения панорамных изображений строится на идеях и методах, изложенных в работе [187]. Алгоритм построения панорамы, описанный в данной работе, принимает на вход изображения F_1, F_2, \dots, F_n , а на выходе у него панорамное изображение F_P . При построении панорамы выполняются следующие шаги.

1. Для каждого изображения F_i выделяются характерные точки $\{S_{ik}\}$ и векторы Δ_{ij} описания изображения в точке, инвариантного к масштабированию и повороту. Такой вектор называется дескриптором или информационным вектором, вычисление координат особых точек и их инвариантов проводится согласно способу, описанному в работе [186].
2. Для каждой пары изображений F_i и F_j производится сопоставление характерных точек $\{S_{ik}\}$ и $\{S_{jl}\}$ на основе инвариантного описания Δ_{ik} и Δ_{jl} по методу, описанному в работе [188].
3. Для каждой пары изображений, у которых на предыдущем шаге выделено достаточно большое количество соответствующих друг другу характерных точек, оценивается матрица поворота оптической оси камеры от положения, при котором снималось одно из изображений в паре, в положение, при котором снималось другое изображение. Также оценивается матрица камеры. Метод оценки внутренних и внешних параметров камер повторяют методы оценки параметров камеры при калибровке стереопары (см. предыдущую главу), отличие в том, что выбор пар на предыдущем шаге осуществляется приближённым способом, поэтому требуется обеспечить фильтрацию шумовых пар. Для этого используется метод RANSAC [189].
4. В наборе изображений $\{F_i\}$ выделяются связанные компоненты в виде остовного дерева из тех пар, для которых удалось достаточно точно оценить внутренние и внешние параметры камер.
5. Каждое изображение отображается в координатах сферической проекции. При этом ценой небольших искажений исходных изображений достигается одинаковое разрешение и в центре панорамного изображения, и на краях. Для устранения искривлений панорамного изображения в сферических координатах среди всех оптических осей выбирается главное направление, которое будет соответствовать средней широте сферической проекции.
6. Каждое изображение $\{F_i\}$ приводится к одной яркости и контрастности, чтобы переходы между границами исходных изображений на панораме были менее заметны.
7. Производится «швивка» изображений методом наложения пирамид исходных изображений друг на друга для того, чтобы погрешность совмещения отдельных пар

изображений не влияло на качество визуализации панорамы.

Практика показывает, что программный модуль [185] достаточно надёжно обрабатывает изображения панорамной съёмки, однако он не пригоден для использования в системах реального времени, потому что, во-первых, десяток кадров обрабатывается десятки секунд, а во-вторых, сложность работы данного алгоритма равна $O(n^2)$, где n — количество кадров, поданных на вход алгоритма построения панорамы. Это значит, что чем больше кадров в режиме обработки видеопотока будет поступать на вход данного алгоритма, тем линейно меньше будет скорость работы устройства, реализующего данный алгоритм.

6.2. Исследование влияния выбора информации, описывающей характерные точки изображений, на точность и вычислительную сложность алгоритмов построения панорамы

Процесс построения панорамного изображения в конечном итоге сводится к задаче сопоставления двух кадров $F_1(x, y)$ и $F_2(x, y)$, полученных видеокамерой из одной точки, но с разным направлением оптической оси, и, возможно, разными настройками увеличения объектива, при этом часть видимой сцены на кадрах F_1 и F_2 совпадает. Под сопоставлением изображения имеется ввиду поиск преобразования сопряжения (см. п. 2.3), что, фактически, совпадает с задачей поиска внутренних параметров камеры при съёмке изображений F_1 и F_2 и внешних параметров поворота камеры из положения, в котором производилась съёмка кадра F_2 в положение, при котором производилась съёмка кадра F_1 : из данных матриц формируется проективное преобразование, переводящее координаты изображения общей сцены с одного кадра, на другой. Для поиска данных матриц и преобразований, как уже отмечалось ранее, необходимо найти как минимум 4 точки соответствия на общей части сцены кадра, причём это нужно сделать в полностью автоматическом режиме.

В библиотеке построения панорамных изображений [185] используется метод поиска характерных точек SIFT (scale-invariant feature transform) [186], который основан на анализе локальных экстремумов функции

$$L(x, y, \sigma) = D(x, y, k\sigma) - D(x, y, \sigma), \quad (6.1)$$

где k — некоторый параметр, близкий по значению к единице, но не равный ей, а $D(x, y, \sigma) = g(x, y, \sigma) * F_\bullet(x, y)$ — свёртка с ядром Гаусса $g(x, y, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2+y^2}{2\sigma^2}}$.

Выбор семейства функций с параметром σ (6.1) обусловлен рядом свойств, рассмотренных в работах [190, 191, 192, 193].

Во-первых, показано, что семейство функций (6.1) являются результатом решения дифференциального уравнения тепловодности, то есть $\frac{\partial L}{\partial \sigma}(x, y, \sigma) = \sigma \Delta L(x, y, \sigma)$ при условии, что сделана замена временного параметра $t = \sigma^2/2$ (в классической

форме уравнение тепловпроводности записывается как $\frac{\partial L}{\partial t} = \Delta L$). Это позволяет при анализе экстремумов функции (6.1) перейти к приближённой функции при аппроксимации производной отношением приращений функции $L(x, y, \sigma)$ по параметру σ :

$$\frac{D(x, y, k\sigma) - D(x, y, \sigma)}{k\sigma - \sigma} \approx \frac{\partial L}{\partial \sigma} = \sigma \Delta L(x, y, \sigma) \Rightarrow L(x, y, \sigma) = (k - 1)\sigma^2 \Delta L, \quad (6.2)$$

то есть с точностью до константы $k - 1$ и нормировочного коэффициента σ^2 разность (6.1) представляет собой результат действия оператора Лапласа $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$. А в работе [193] показано, что экстремумы Лапласиана изображения эффективнее выделяют характерные точки, чем, например, детектор Хариса [173], в том смысле, что результат работы детектора характерных точек более устойчив к шуму и аффинным преобразованиям.

Во-вторых, в указанных выше работах показано, что точки локальных экстремумов (x_i, y_i, σ_i) функции (6.1) соответствуют некоторым характерным областям изображения, причём масштаб областей соответствует значению σ_i в точке локального экстремума.

Для поиска локальных экстремумов функции (6.1) в работе [187] предложено использовать следующий алгоритм.

1. Значение функций $L(x, y, \sigma)$ для изображения $F(x, y)$ рассчитывается на сетке $\{(x_l, y_m, \sigma_n)\}$, где $\{(x_l, y_m)\}$ — равномерная сетка с целочисленными координатами на изображении F , а $\{\sigma_n\} = \{2^{\frac{n}{s}}\}$ — сетка значений с экспоненциальным шагом при условии, что коэффициент $k = 2^{\frac{1}{s}}$ для функции (6.1), s — некоторое натуральное число не менее 2. Эксперименты показывают, что значение $s = 3$ формирует достаточную мелкость разбиения сетки для работы алгоритма. Следует отметить, что при переходе через узлы сетки $n = j \cdot s$ можно для сокращения числа вычислительных операций перейти к уменьшенному вдвое разрешению при расчёте значений функции $L(x, y, \sigma)$, пользуясь соотношениями $L(x, y, \sigma) \approx \tilde{L}(x/2, y/2, \sigma/2)$ и $L(x/2, y/2, \sigma/2) = = \frac{1}{4}(L(x, y, \sigma) + L(x + 1, y, \sigma) + L(x, y + 1, \sigma) + L(x + 1, y + 1, \sigma))$, где \tilde{L} — функция, рассчитанная на сетке уменьшенного разрешения. Подобный подход принято называть работой с пирамидой изображений, см. п. 1.3.

2. В первом приближении экстремум функции $L(x, y, \sigma)$ — это такая точка сетки $\{(x_l, y_m, \sigma_n)\}$, для которой все соседние значения функции в окрестности $3 \times 3 \times 3$ либо не больше, либо не меньше значения в центре окрестности. Список характерных точек $\{(\hat{x}_j, \hat{y}_j, \hat{\sigma}_j)\}_{j=1}^Q$ формируется после прохода всех внутренних точек сетки, введённой на предыдущем шаге, обозначим количество найденных экстремумов как Q . Подобный подход вполне устойчив к шуму изображения в данном случае, поскольку исходное изображение слажено Гауссовой фильтрацией.

3. Положение локальных экстремумов, найденных приближённо на первом шаге, можно уточнить, используя аппроксимацию второго порядка с помощью разложения в ряд Тейлора:

$$L(\mathbf{r}) \approx L(\hat{\mathbf{r}}) + \frac{\partial L^T}{\partial \mathbf{r}} \Delta \mathbf{r} + \frac{1}{2} \Delta \mathbf{r}^T \frac{\partial^2 L}{\partial \mathbf{r}^2} \Delta \mathbf{r}, \quad (6.3)$$

где \mathbf{r} и $\hat{\mathbf{r}}$ — трёхмерные вектор-столбцы, компоненты которых соответствуют трём аргументам функции $L(x, y, \sigma)$, $\Delta \mathbf{r} = \mathbf{r} - \hat{\mathbf{r}}$, $\frac{\partial L}{\partial \mathbf{r}}$ — столбец, соответствующий вектору градиента функции L в точке $\hat{\mathbf{r}}$, $\frac{\partial^2 L}{\partial \mathbf{r}^2}$ — матрица Гессе размера 3×3 вторых производных в точке . Дифференцируя разложение Тейлора по $\Delta \mathbf{r}$, и приравнивая производную к 0, получим

$$\Delta \mathbf{r} = -\frac{\partial^2 L}{\partial \mathbf{r}^2}^{-1} \cdot \frac{\partial L}{\partial \mathbf{r}},$$

что позволяет уточнить положение экстремума $\mathbf{r} = \hat{\mathbf{r}} + \Delta \mathbf{r}$, а также уточнить после подстановки в (6.3) значение самого экстремума

$$L(\hat{\mathbf{r}}) = L(\mathbf{r}) + \frac{1}{2} \frac{\partial L^T}{\partial \mathbf{r}} \cdot \frac{\partial^2 L}{\partial \mathbf{r}^2}^{-1} \cdot \frac{\partial L}{\partial \mathbf{r}}. \quad (6.4)$$

Таким образом, положение экстремумов $\{\hat{x}_j, \hat{y}_j, \hat{\sigma}_j\}_{i=1}^Q$ уточняется, и в конце данного шага получается список характерных точек и их радиусов $\{(x_j, y_j, \sigma_j)\}_{i=1}^Q$.

4. Для каждой точки проводится дополнительная фильтрация: если в процессе расчётов по формуле (6.4) выясняется, что значение $|L(\mathbf{r}) - L(\hat{\mathbf{r}})|/L(\mathbf{r})$ слишком мало (например, менее 0,01), то окрестность точек \mathbf{r} и $\hat{\mathbf{r}}$ признаётся окрестностью с изображением слишком низкого контраста для того, чтобы выделить устойчивый локальный экстремум. Затем в каждом экстремуме производится оценка устойчивости по критерию Хариса [173]

$$\frac{(\text{tr } H)^2}{\det H} < \frac{(\xi_{max} + 1)^2}{\xi_{max}},$$

где H — элементы матрицы Гессе $\frac{\partial^2 L}{\partial \mathbf{r}^2}$, соответствующие производным по x и по y , а ξ_{max} — максимально допустимое отношение $\xi = \lambda_{max}/\lambda_{min}$ максимального собственного значения матрицы H к минимальному собственному значению. Данное неравенство следует из того, что $\text{tr } H = \lambda_{min} + \lambda_{max}$, $\det H = \lambda_{min} \cdot \lambda_{max}$ и функция $\frac{(\xi+1)^2}{\xi}$ при $\xi \geq 1$ монотонно возрастает. ξ_{max} является параметром алгоритма, который, как правило, выбирают из диапазона от 10 до 20. В результате данного шага из Q экстремумов отбирается Q' характерных точек $\{(x_j, y_j, \sigma_j)\}_{i=1}^{Q'}$.

5. В каждой характерной области $\{(x_j, y_j, \sigma_j)\}_{i=1}^{Q'}$ необходимо вычислить информационный вектор Δ_j , который бы не зависел от масштаба и поворота характерного элемента изображения, соответствующего выделенным на предыдущих этапах алгоритма характерным точкам. Для этого в работе [186] предложено посчитать

в два этапа гистограмму распределения норм градиентов по их направлениям в окрестности $U_\varepsilon(x, y)$ точки (x, y) . При первом проходе радиус окрестности считается равным $\varepsilon = 4,5\sigma_j$ по норме в пространстве L_1 , в каждой точке для градиента $\nabla = (L(x_j + 1, y_j, \sigma_j) - L(x_j - 1, y_j, \sigma_j), L(x_j, y_j + 1, \sigma_j) - L(x_j, y_j - 1, \sigma_j))$ формируется гистограмма распределения нормы $|\nabla| = \sqrt{\nabla_x^2 + \nabla_y^2}$ от угла $\theta = \arctg \frac{\nabla_x}{\nabla_y}$, углы от 0° до 360° разбиваются при этом на 36 отрезков, 10° каждый. На полученной гистограмме $|\nabla|(\theta)$ находится максимум, а также все значения, которые не меньше, чем 80% от максимума. Из указанных значений формируются уже чётверки чисел $\{(x_j, y_j, \sigma_j, \theta_j)\}_{i=1}^{Q''}$ в количестве Q'' , т.е. у каждой характерной области помимо радиуса появляется главное направление, причём для одной точки исходного набора может быть добавлено несколько точек с одним и тем же положением и радиусом, но разными направлениями в том случае, если максимум на гистограмме $|\nabla|(\theta)$ не сильно выделяется среди остальных значений. На втором этапе производится расчёт компонентов вектора Δ_j . Для этого используется окрестность $U_\varepsilon(x_j, y_j)$ при $\varepsilon = 8$ точек изображения $L'(x, y, \sigma_j)$, которое получается после поворота изображения $L(x, y, \sigma_j)$ вокруг точки (x_j, y_j) на угол θ_j , причём радиус окрестности в данном случае учитывается по манхэттенской норме L_1 . Получившийся квадрат 16×16 разбивается на квадраты меньшего размера 4×4 , т.е. на $4 \times 4 = 16$ клеток. В каждой клетке строится гистограмма распределения нормы градиентов функции $L'(x, y, \sigma_j)$ в 16 точках по 8 направлениям угла (то есть 360° разбивается на интервалы направлений по 45° , четыре направления вправо, вверх, влево, вниз, и четыре промежуточных направления). При этом вклад в ячейку гистограммы домножается на вес $e^{-d^2/32}$, где d — расстояние по норме L_2 до центра окрестности $U_\varepsilon(x_j, y_j)$. Для сглаживания эффекта искажения информации в гистограмме из-за случайного шума, который может приводить к попаданию значения градиента в соседние ячейки, гистограмма сглаживается за счёт добавки в соседние ячейки некоторой доли от нормы градиента, причём добавка линейно зависит от циклического расстояния между ячейками (т.е. расстояние между 0-й и 7-й ячейкой гистограммы — единица вследствие поворота на соответствующих этим ячейкам градиентов на углы порядка 360°). Элементы гистограммы нормируются на единицу, а затем, чтобы устраниТЬ влияние переполнения динамического диапазона чувствительности сенсоров камеры при засветках, все значения выше определённого значения устанавливаются равными этому значению, для сглаживания выбросов при резкой смене освещения. В конце концов производится новая нормировка на единицу, и из кортежей-гистограмм с восемью значениями для каждого из 16 квадратов 4×4 составляется посредством конкатенации информационный вектор Δ_j , значения которого обладают некоторой устойчивостью по отношению к поворотам изображений, к случайному шуму, к изменению освещённости в определённых пределах и к неболь-

шим перспективным искажениям, т.е. является инвариантным описанием небольшой области (x_j, y_j, σ_j) по отношению к небольшому изменению ракурса и условий съёмки.

Изложение данного известного алгоритма сделано для того, чтобы провести подробный анализ вычислительной сложности операций, задействованных в нём. В доступных автору источниках информация о сложности алгоритма не приводится, тем не менее, она является важной для оценки применимости алгоритма в системах реального времени.

Сложность шага 1 соответствует сложности вычисления sN свёрток изображения $F(x, y)$ с ядром $g(x, y, \sigma_n)$ при условии, что N — это количество уровней пирамиды изображений $L(x, y, \sigma_n)$ (на каждом уровне в рассматриваемом алгоритме было введено s изображений, отличающихся по масштабу сглаживания не более, чем в 2 раза). Свёртка может быть вычислена с помощью преобразования Фурье, асимптотика алгоритмов быстрого преобразования Фурье равна $O(WH \log_2 WH)$, где $W \times H$ — разрешение изображения $F(x, y)$. Однако существует алгоритм приближённого вычисления свёртки с гауссовым ядром, асимптотика которого $O(WH)$, см. [194, 195, 196]. Таким образом, сложность выполнения одной свёртки примерно равна $C_1 HW$, где C_1 — константа, характеризующая количество сложений и умножений при расчёте отклика конечной импульсной характеристики, приближающей фильтр сглаживания (примерно по 20 сложений и по 20 умножений, т.е. 40 операций с вещественными числами на одну точку). С учётом того, что на каждом уровне пирамиды разрешение обрабатываемых изображений снижается вдвое по ширине и высоте, то в целом сложность вычисления семейства сглаженных функций $L(x, y, \sigma_n)$ равна

$$\sum_{t=1}^N sC_1 \frac{W}{2^{t-1}} \frac{H}{2^{t-1}} = sC_1 HW \sum_{t=1}^N \frac{1}{4^{t-1}} \leq \frac{4}{3} sC_1 HW,$$

поскольку частичная сумма геометрической прогрессии может быть оценена как

$$\sum_{t=1}^N \frac{1}{4^{t-1}} \leq \sum_{t=1}^{\infty} \frac{1}{4^{t-1}} = \frac{1}{1 - 1/4} = \frac{4}{3}.$$

Таким образом, асимптотика построения сглаженных разностей изображений равна $O(sWH)$ и с точностью до константы-параметра алгоритма s соответствует асимптотике линейной фильтрации, приближающей Гауссово размытие. При $s = 3$ на каждую точку изображения $F(x, y)$ приходится около 160 вещественных операций сложения и умножения.

На шаге 2 циклически в каждой точке пирамиды изображений $\{L(x, y, \sigma_n)\}$ требуется сделать $3 \times 3 \times 3 - 1 = 26$ сравнений значения функций в текущей точке со значениями в соседних точках сетки. Следовательно, повторяя рассуждения при

оценке сложности предыдущего шага, сложность данной операции равна

$$26C_{comp} \frac{4}{3} WH,$$

где C_{comp} — сложность одной операции сравнения интенсивности в двух разных точках. То есть сложность операции поиска экстремумов $O(HW)$, на каждую точку изображения приходится около 35 операций сравнения вещественных чисел.

Алгоритмическая сложность шагов 3 и 4 определяется количеством характерных точек Q , для которых необходимо рассчитать матрицу Гёссе (на каждый из 9 элементов матрицы одно умножение и два сложения), градиент (6 сложений), вычислить обратную матрицу (примерно 45 операций сложения и умножения) и получить уточнённое значение экстремума (12 умножений, 9 сложений) и уточнённое положение (3 сложения). После фильтрации по контрастности в окрестности экстремума (одна операция сравнения) необходимо в каждой точке провести дополнительную фильтрацию по критерию Хариса (примерно 20 операций с вещественными числами). В итоге получается порядка 120 вещественных операций на каждую из Q точек, в целом сложность данной процедуры $O(Q)$.

Шаг 5 по вычислению 128-компонентного вектора Δ_j состоит из двух этапов по оценке направления (сложность $O(Q')$) и расчёта компонентов вектора (сложность $O(Q'')$). Для оценки направления необходимо произвести расчёт градиентов в квадратной окрестности не более, чем $18 \times 18 = 324$ точек, потому что радиус $\varepsilon = 4,5\sigma_j$ на уровне пирамиды $[j/N]$ охватывает от 4,5 до 9 точек изображения по принципу построения пирамиды, а сама окрестность в два раза шире. Сложность построения гистограммы градиентов из 36 ячеек и поиска максимума определяется, таким образом, сложностью следующих операций:

- операция вычисления нормы и ориентации градиентов: 3 сложения, два умножения, одно вычисление квадратного корня, одно вычисление функции arctg) — всего $7 \cdot 324$ вещественных операций;
- операция взвешивания: 2 умножения и одно сложение для расчёта нормы L_2 до центра окрестности, одно взятие экспоненты, одно умножение на полученный вес — всего $5 \cdot 324$ вещественных операций;
- операция по формированию гистограммы: сложение каждого нового элемента с текущим значением в существующей ячейке гистограммы и одно сравнение для вычисления максимума — всего $2 \cdot 324$ вещественных операций;
- 36 операций сравнения по выбору всех элементов гистограммы, значение которых превышает 80% от максимального значения.

В целом, для расчёта направления в каждой характерной точке требуется $324 \cdot (7 + 5 + 2) + 36 \approx 4600$ вещественных операций. Аналогичную оценку числа вещественных операций можно сделать для вычисления информационного вектора Δ_j , имея ввиду размер окрестности $16 \times 16 = 256$, для которой считается информационный вектор:

- для расчёта нормы градиента и его ориентации требуется 7 операций в каждой точке окрестности (см. выше), для учёта поворота на угол θ_j требуется 4 умножения и 3 сложения в каждой точке окрестности, то есть всего для расчёта градиентов требуется $14 \cdot 256$ операций;
- для взвешивания элементов гистограмм требуется $5 \cdot 256$ операций (см. выше);
- добавление значения в гистограмму по сравнению с предыдущим этапом дополняется сглаживанием за счёт внесение в соседние ячейки уменьшенных домножением на определённый вес значений, что требует дополнительно для каждого элемента 8 действий сложения и умножения для внесения поправок в каждую из 8 ячеек гистограммы блоков 4×4 , то есть сложность данного шага $(2 + 8 + 8) \cdot 256 = 18 \cdot 256$ вещественных операций;
- нормировка, фильтрация больших значений, и вновь нормировка для каждой из 16 гистограмм с восемью ячейками обладает сложностью $8 \cdot 2$ операций сложения и $8 \cdot 2$ операций умножения для нормировки и 8 операций сравнения для фильтрации, всего $(8 \cdot 2 \cdot 2 + 8) \cdot 16 = 5 \cdot 256$ вещественных операций.

Таким образом, расчёт информационного вектора соответствует сложности $(14 + 5 + 18 + 5) \cdot 256 \approx 10800$ элементарных вещественных операций.

Как правило, числа Q , Q' и Q'' являются числами одного порядка, поэтому сложность последних трёх шагов можно обозначить как $O(Q'')$, причём константа пропорциональности ассимптотики примерно соответствует сложности $120 + 4600 + 10800 \approx \approx 15500$ вещественных операций.

Конечная характеристика вычислительной сложности приведённого выше алгоритма выглядит следующим образом: 160 вещественных операций на каждую точку изображения, и 15,5 тысяч вещественных операций на каждую характерную точку. Например, для изображения с разрешением 640×480 , на котором найдено 500 характерных точек¹, обрабатывается с помощью $640 \cdot 480 \cdot 160 + 500 \cdot 15500 \approx 57 \cdot 10^6$ вещественных операций, что при частоте обработки 25 кадров в секунду потребует более одного гигафлопа в секунду. Современные вычислительные приборы обладают

¹На типичном изображении средней текстурированности находится примерно столько характерных точек.

производительностью, которая по порядку величины соответствует данному показателю, поэтому при использовании данного алгоритма практически все ресурсы будут отданы на его исполнение, что не приемлемо, поскольку, например, для алгоритма построения панорамы требуется выполнять и другие операции. Поэтому необходимо использовать более быстрые способы поиска характерных точек. Таким способом является алгоритм, описанный в работе [197]. Этот алгоритм называется ORB (Oriented BRIEF [199]), рассмотрим шаги данного алгоритма и их вычислительную сложность.

1. На основе кадра $F(x, y)$ строится пирамида изображений $\{F_l(x, y)\}$ с мультиплексивным шагом увеличения 1,2. Операция изменения изображения с разрешением $W \times H$ записывается как

$$\begin{aligned} F_0(x, y) &= F(x, y), \\ F_{l+1}(x, y) &= (1 - \alpha)(1 - \beta)F_l(\tilde{x}, \tilde{y}) + \alpha(1 - \beta)F_l(\tilde{x} + 1, \tilde{y}) + \\ &\quad + (1 - \alpha)\beta F_l(\tilde{x}, \tilde{y} + 1) + \alpha\beta F_l(\tilde{x} + 1, \tilde{y} + 1), \\ \text{где } \hat{x} &= \frac{x}{1,2}, \hat{y} = \frac{y}{1,2}, \tilde{x} = \lfloor \hat{x} \rfloor, \tilde{y} = \lfloor \hat{y} \rfloor, \alpha = \hat{x} - \tilde{x}, \beta = \hat{y} - \tilde{y}. \end{aligned}$$

Как следует из данных формул, операция изменения разрешения обладает сложностью $C'_1 \cdot W' H'$, где C'_1 соответствует выполнению в каждой точке изображения с новым разрешением $W' \times H'$ 14 умножений и 5 сложений для билинейной интерполяции. Уммарно для всех N уровней пирамиды количество операций

$$\sum_{t=1}^N C'_1 \frac{WH}{1,2^t \cdot 1,2^t} \leq C'_1 WH \sum_{t=1}^{\infty} \left(\frac{1}{1,44} \right)^t = C'_1 WH \left(\frac{1}{1 - 1/1,44} - 1 \right) \approx 2,3C'_1 WH,$$

поэтому для построения пирамиды на каждую точку приходится $2,3 \cdot (14 + 5) \approx 39$ вещественных операций.

2. В каждой точке изображений пирамиды $\{F_l\}$ производится проверка на то, является ли она характерной, согласно алгоритму, описанному в работе [198]: точка считается характерной, если существует последовательность из 9 соседних пикселей в окрестности-кольце из 16 точек, пересекаемых окружностью радиуса 3 с центром в проверяемой точке (см. рис. 6.1), в которых интенсивность изображения либо всегда больше некоторого порога δ , либо всегда меньше этого порога. В худшем случае для осуществления этой операции необходимо сделать 16 последовательных сравнений, однако в указанной работе предложен способ сократить это число до 2–4 за счёт того, что сразу отбрасываются точки, в которых заведомо не выполняется указанное условие. Таким образом, при поиске характерных точек в каждой точке выполняется примерно 3 вещественные операции, остальные операции, а именно: сравнение с порогом δ , рассчёт смещения точек на окружности радиуса 3 — выполняются заранее, их результат доступен в форме таблиц во время выполнения алгоритма. Таким

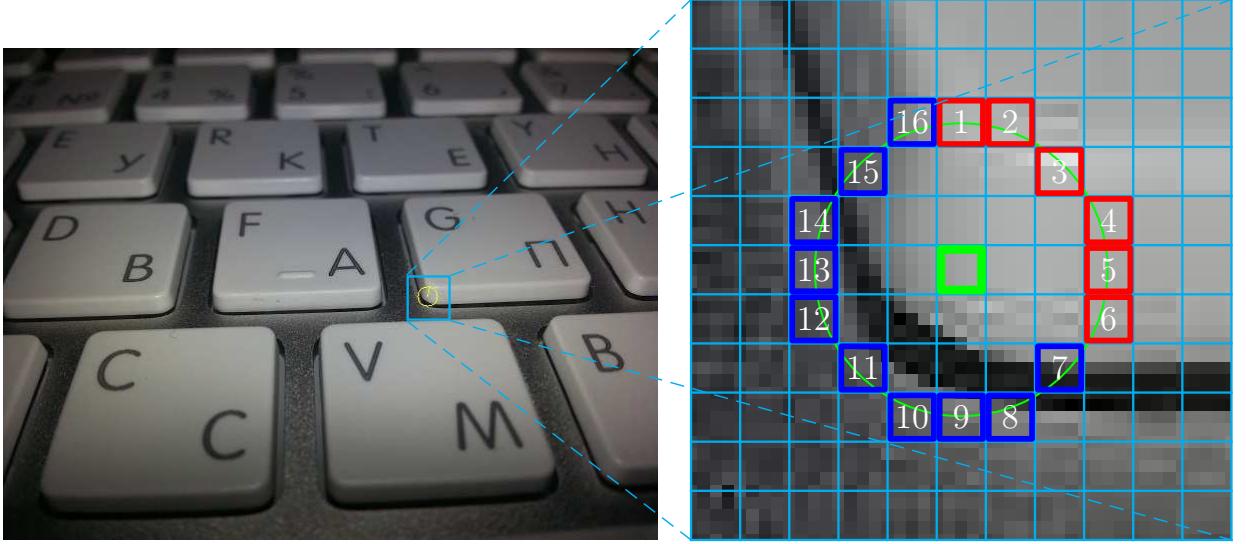


Рисунок 6.1. Схема проверки характерной точки FAST: точки с 1 по 16 находятся примерно на расстоянии 3 от тестируемой точки, точка является характерной, если на окружности есть непрерывная дуга из как минимум 9 точек, интенсивность которых либо больше интенсивности центральной точки на порог δ , либо меньше интенсивности центральной точки на этот порога (на рисунке точки с 7 по 16 меньше центральной точки).

образом, в целом в каждой точке изображений $\{F_l\}$ выполняется 4 вещественных операции, и на каждую точку исходного изображения F приходится $2,3 \cdot 4 \approx 9$ операций, в результате формируется список из Q точек $\{(x_j, y_j, r_j)\}_{j=1}^Q$, в которых радиус r_j определяется уровнем пирамиды изображений, на котором выполнилось приведённое выше условие для характерной точки.

3. Для повышения устойчивости каждая точка из числа $\{(x_j, y_j, r_j)\}_{j=1}^Q$ оценивается с помощью критерия Харриса, что, как показано для предыдущего алгоритма SIFT, требует около 20 вещественных операций в каждой точке. Список точек сортируется по отклику Харриса с помощью $Q \log_2 Q$ операций сравнений и перестановок быстрой сортировки для того, чтобы отобрать фиксированное количество точек Q' с наилучшим показателем устойчивости.

4. В каждой из точек $\{(x_j, y_j, r_j)\}_{j=1}^Q$ производится оценка ориентации характерной области. Для этого в окрестности $U_{15}(x_j, y_j)$ (радиус указан по норме L_2) рассчитывается значение первых моментов F_{mx}, F_{my} интенсивности и сумму интенсивностей F_m на соответствующем точке уровне пирамиды $F_l(x, y)$. Направление вектора $(F_{mx}/F_m - x_j, F_{my}/F_m - y_j)$ считается равным

$$\theta_j = \operatorname{arctg} \frac{F_{mx} - F_m x_j}{F_{my} - F_m y_j}.$$

Неопределённость вида $0/0$ в данном соотношении возникает редко ввиду заранее заданного перепада интенсивностей по условию поиска характерной точки. Окрестность U_{15} состоит из $\pi 15^2 \approx 700$ точек, и для расчёта координат центроида необходимо

сделать в каждой точки окрестности 2 умножения и 4 сложения, т.е. 6 вещественных операций. Общая сложность вычисления θ_j в таком случае $700 \cdot 6 = 4200$ вещественных операций на каждую из Q' точек.

5. Информационный вектор Δ_j для точки (x_j, y_j) в данном алгоритме состоит из 256 значений, которые равны либо 0, либо 1. Для расчёта Δ_j вводится квадратная окрестность 31×31 , повернутая относительно точки (x_j, y_j) на угол θ_j . Каждая s -ая компонента вектора Δ_j соответствует паре точек (x'_{sj}, y'_{sj}) и (x''_{sj}, y''_{sj}) , принадлежащей указанной окрестности и обладающей псевдослучайными координатами, которые указываются в повернутой на θ_j системе координат с началом в точке (x_j, y_j) , расчёт значения Δ_{sj} производится следующим образом:

$$\begin{aligned}\hat{x}'_{js} &= x_j + \cos \theta_j x'_{js} - \sin \theta_j y'_{js}, \\ \hat{y}'_{js} &= y_j + \sin \theta_j x'_{js} + \cos \theta_j y'_{js}, \\ \hat{x}''_{js} &= x_j + \cos \theta_j x''_{js} - \sin \theta_j y''_{js}, \\ \hat{y}''_{js} &= y_j + \sin \theta_j x''_{js} + \cos \theta_j y''_{js}, \\ \Delta_{sj} &= \begin{cases} 1, & \text{если } F_l(\hat{x}'_{js}, \hat{y}'_{js}) < F_l(\hat{x}''_{js}, \hat{y}''_{js}), \\ 0 & \text{в противном случае.} \end{cases}\end{aligned}$$

Набор из 256 пар точек (x'_{sj}, y'_{sj}) и (x''_{sj}, y''_{sj}) известен заранее до выполнения алгоритма, координаты подбираются методом машинного обучения на большом числе (десятка тысяч) изображений размера 31×31 , таким образом, чтобы с одной стороны все значения Δ_{sj} принимали с равной вероятностью как значение 1, так и значение 0, а с другой стороны, взаимная корреляция различных компонент вектора была минимальной. Эти критерии позволяют по возможности исключить информационную избыточность в компонентах векторов Δ_j . На каждое сравнение по приведённым выше формулам приходится по 8 умножений, 8 сложений и одному сравнению, т.е. для всего дескриптора это составляет $(8 + 8 + 1) \cdot 256 = 4350$ вещественных операций. Чтобы шум не влиял на результат сравнения каждое изображение пирамиды $\{F_l\}$ предварительно сглаживается фильтром Гаусса, для этого нужно выполнить 40 вещественных операций для точек пирамиды изображений (см. выше оценку сложности фильтра сглаживания).

Итак, для работы быстрого алгоритма выделения характерных точек с их описанием требуется 80 вещественных операций в каждой точке $W \times H$, и примерно 8600 операций для каждой найденной точки. Это примерно в два раза меньше, чем нужно в алгоритме SIFT, приведённом ранее. На практике выигрыш расчёта характерных точек ORB получается больше, за счёт того, что во время выполнения операций совершается значительно меньше обращений к внешней памяти при исполнении программы на процессорах современных архитектур, в каждой из которых предусмотрена

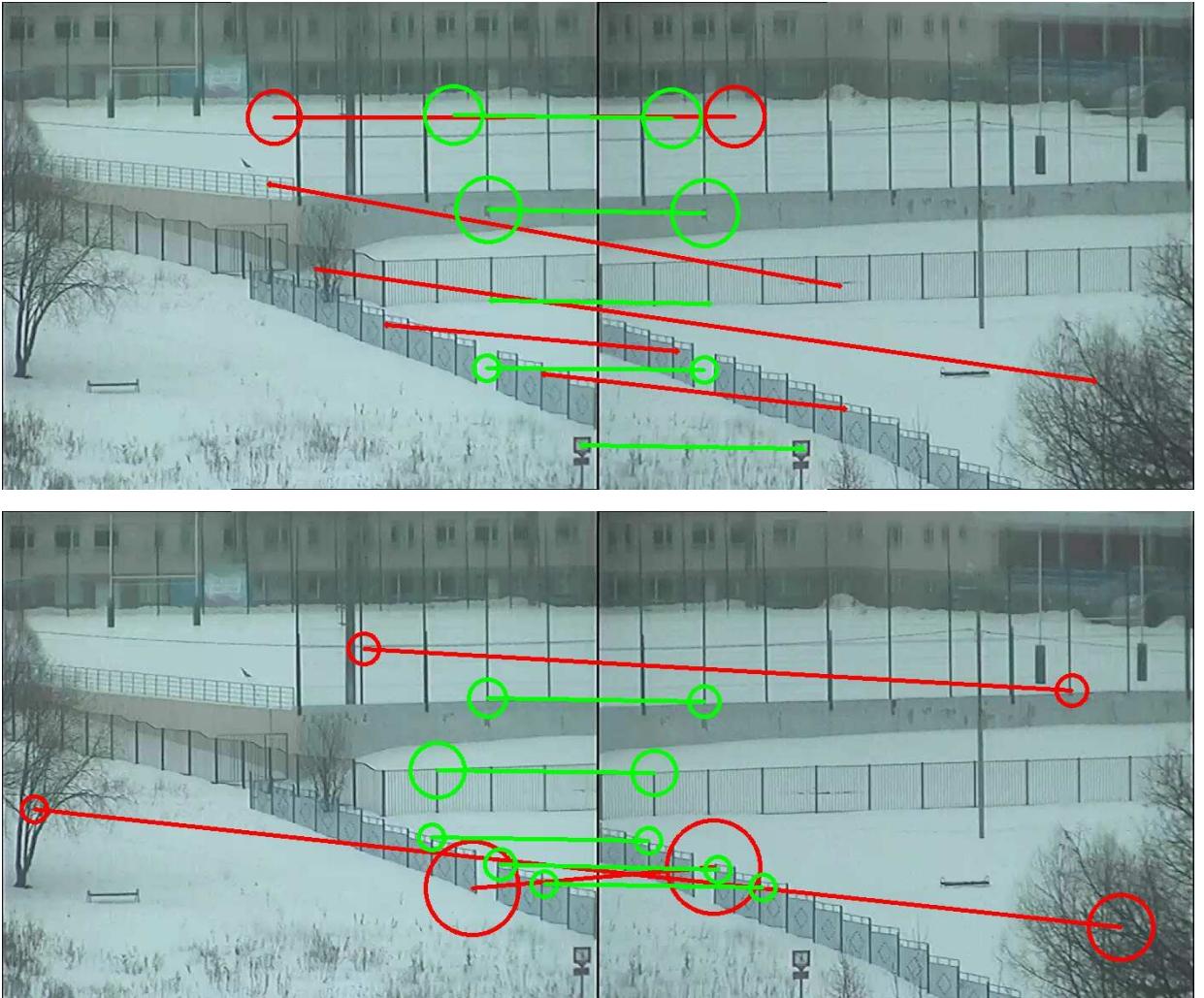


Рисунок 6.2. Примеры поиска характерных точек и их пар для SIFT (сверху) и ORB (снизу): светлым соединены правильно определённые пары характерных точек, тёмным соединены ошибочно определённые пары точек.

быстрая внутренняя память достаточно большого объёма, так называемый кэш.

Примеры ошибочных и правильных сопоставлений с помощью сравнения информационных векторов по методу жадного алгоритма паросочетаний для SIFT и ORB см. на рис. 6.2.

Согласно тестам на нескольких десятках тысяч изображений в работе [197], с помощью информационного вектора SIFT можно точнее, чем с помощью ORB, идентифицировать фрагменты изображений, которые получены при съёмке с другого ракурса. Данные тесты подразумевают либо наличие размеченных вручную пар изображений, на которых пользователь указывает положение одних и тех же точек сцены, либо автоматическую генерацию парных изображений: на исходное изображение добавляется гауссов шум, изображение подвергается поворотам, изменениям масштаба и проективным искажениям, после этого выясняется, какая часть характерных точек исходного изображения может быть найдена по методу поиска ближайших соседей в пространстве информационных векторов на изменённом изображении. Тесты повтор-

ряются для большого набора случайных изображений, например, из сети Интернет. Согласно этим тестам метод ORB более устойчив, чем SIFT, по отношению к шуму, но менее устойчив к геометрическим изменениям области изображения вокруг характерных точек. Тем не менее, проигрыш не столь существенный, и для решения конечной задачи подбора преобразования, которое переводит одно изображение в другое, как показывает практика, дополнительные ошибки ORB не имеют значения, поскольку они компенсируются на этапе фильтрации ошибочного поиска пар точек, например, методом RANSAC.

Несмотря на меньшую точность идентификации с помощью ORB, гораздо большее значение для построения панорамных изображений в реальном времени имеет кратное превышение скорости работы этого алгоритма над скоростью работы более часто применяемого алгоритма SIFT.

6.3. Анализ влияния предварительной юстировки поворотного устройства на эффективность алгоритмов обработки панорамных изображений

Выделение характерных точек на паре изображений в автоматическом режиме, рассмотренное в предыдущем параграфе, необходимо для того, чтобы вычислить преобразование координат с одного изображения на другое, чтобы впоследствии произвести операцию наложения изображений для построения панорамы. Операция наложения изображений с общими элементами сцены уже рассматривалась в параграфе 2.3 для случая, когда точки соответствия изображений устанавливаются при настройке операторами. При построении панорамы подобная операция должна производится полностью автоматически, причём алгоритмы построения проективного преобразования должны учитывать то, что существенная часть пар точек, найденных автоматически, сопоставлена ошибочно, что иллюстрирует рис. 6.2. Один из способов фильтрации неверных пар точек заключается в применении метода RANSAC. Второй подход основан на использовании калибровочных параметров видеокамеры и поворотного устройства, и будет рассмотрен в данном параграфе.

Используя модель видеокамеры, введенную в параграфе 5.1, проективное преобразование мировых координат в обобщенные координаты изображения можно представить в виде суперпозиции двух преобразований: проективного преобразования и преобразования поворота, то есть если x, y — координаты на кадре $F(x, y)$, а X, Y, Z — трёхмерные координаты в системе отсчёта, связанной с наблюдаемой сценой, то координаты объектов сцены и однородные координаты их изображений связаны как

$$\alpha \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = M \cdot [R|P] \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}, \quad (6.5)$$

где M — матрица параметров камеры, R — матрица поворота, которая переводит координаты из системы, связанной со сценой в систему, связанную с камерой, а P — вектор-столбец, равный смещению между началами отсчёта двух систем координат.

Рассмотрим видеокамеру, установленную на поворотном устройстве. Как правило, поворотные устройства для видеокамер обладают двумя степенями свободы при движении: вращение вокруг горизонтальной оси и вращение по отношению к горизонту, см. п. 3.3. Будем считать, что две эти оси пересекаются в одной точке, назовём её центром вращения. Пусть начало отсчёта системы координат, связанной с наблюдаемой сценой, находится в центре вращения камеры, ось Y совпадает с вертикальной осью вращения, а ось Z совпадает с направлением оптической оси видеокамеры при нулевом угле наклона в одном из положений поворотной платформы, см. рис. 6.3. Горизонтальный угол поворота платформы в этом положении также примем равным нулевым. При таком выборе точки отсчёта координат X, Y, Z вектор $P = 0$ при любых значениях углов поворота и наклона, которые далее обозначаются как φ и θ , соответственно. Если на изображении выбрать систему координат для x, y , начало которой $(0, 0)$ расположено в центре изображения, то для видеокамер с объективом без астигматизма $M = \text{diag}(f, f, 1)$, где f — это фокусное расстояние оптической системы, измеренное в единицах, равных по размеру одному элементу чувствительного сенсора камеры, а матрица поворота R является произведением двух матриц R_θ, R_φ , соответствующих преобразованиям поворота на угол наклона θ (вокруг оси OX) и угол вращения φ (вокруг оси OZ), то есть $R = R_\theta R_\varphi$, где

$$R_\theta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{bmatrix}, \quad R_\varphi = \begin{bmatrix} -\sin \varphi & 0 & \cos \varphi \\ 0 & 1 & 0 \\ \cos \varphi & 0 & \sin \varphi \end{bmatrix}.$$

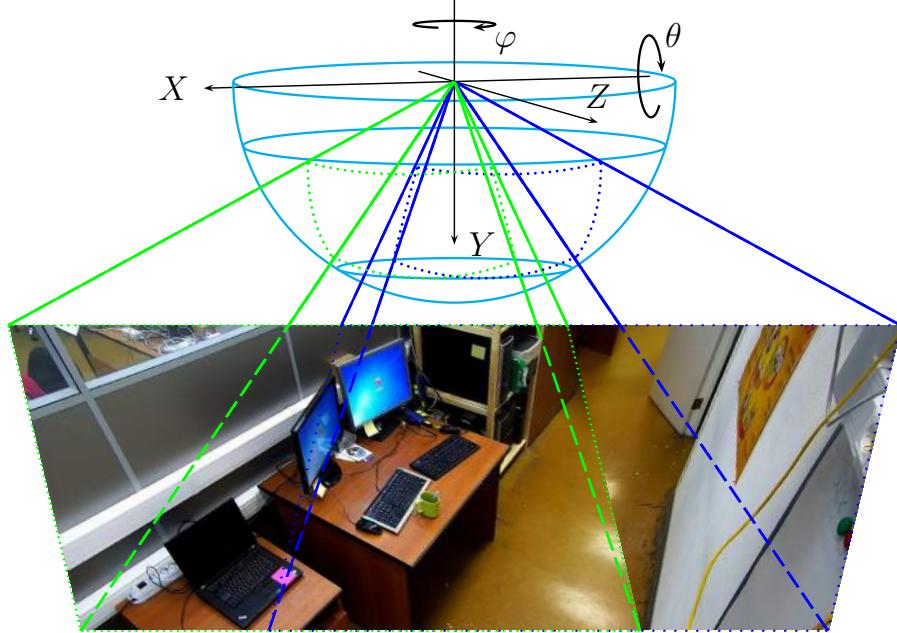
Как правило, в процессе сканирования угол наклона θ не меняется, а угол поворота φ изменяется с некоторой постоянной по модулю скоростью (направление сканирования может меняться с заданным периодом).

Метод, предложенный в работе [187], предполагает оценку параметров f, θ, φ посредством сопоставления характерных точек. Для этого на двух изображениях $F_1(x, y)$ и $F_2(x, y)$ производится поиск характерных пар точек с помощью алгоритмов, описанных в предыдущем параграфе. Обозначим пары точек как $\{u_i, v_i\}_{i=1}^Q$, где $u_i = [x_1^i \ y_1^i \ 1]^T, v_i = [x_2^i \ y_2^i \ 1]^T$ — вектор-столбцы обобщённых координат, нижние индексы при x и y соответствуют индексам изображений F_1 и F_2 . Выберем начальное положение координат таким, что ось Z совпадает с оптической осью камеры в положении, соответствующем изображению F_1 , и обозначим $H_1 = M_1$ и $H_2 = M_2 R$, где нижние индексы матриц означают их принадлежность к описанию положения камер, в которых получены изображения F_1 и F_2 , соответственно, а матрица R является



Исходное изображение
 $(\theta = 39^\circ, \varphi = 196^\circ)$

Изображение после смещения
 $(\theta = 39^\circ, \varphi = 225^\circ)$



Изометрическая проекция двух плоскостей проектирования изображений

Рисунок 6.3. Схема сканирования наблюдаемой сцены видеокамерой с углом зрения $54,1^\circ$ на поворотном устройстве. Тёмные прямые линии соответствуют исходному положению камеры, светлые прямые линии соответствуют положению камеры после сдвига. На сфере точками, соответственно, тёмного и светлого пунктира показаны области определения преобразования сферического проектирования прямоугольных изображений, а также показана область-полоса на сфере, соответствующая панорамному изображению при непрерывном изменении угла φ .

матрицей поворота, переводящей систему координат второй камеры в систему координат первой камеры. Поскольку $H_1^{-1}u_i \cong H_2^{-1}v_i$, то мы можем определить матрицу $H = H_1H_2^{-1}$ проективного преобразования из координат изображения F_2 к координатам изображения F_1 , сначала минимизируя невязку в системе линейных уравнений относительно элементов матрицы H

$$\alpha_i u_i = H v_i, i \in [1, Q],$$

а затем с помощью методов оптимизации минимизировать уже саму невязку $\sum_{i=1}^Q \|u_i - H v_i\|$. Одно из решений данной задачи предложено автором работы в п. 2.3, а также в [201]. Как показывает рис. 6.2, многие пары сопоставляются ошибочно. Для

устранения влияния ошибочных паросочетаний используют метод RANSAC [189] последовательного случайного отбора подмножества пар из исходного множества, и для разных случайных подвыборок многократно осуществляется поиск матрицы H , из всех решений выбирается матрица H с наименьшей итоговой невязкой.

После того, как найдена матрица H , можно найти фокусное расстояние и матрицу поворота, действуя так, как это описано в работе [200]: используя свойство ортогональности введённой ранее матрицы поворота, а также результат умножения матричных равенств

$$\alpha H M_2 = H_1 H_2^{-1} M_2 = M_1^{-1} R^{-1}, \alpha M_2^T H^T = R(M_1^{-1})^T,$$

получаем в силу симметрии диагональных матриц M_1 и M_2

$$\alpha^2 H M_2 M_2^T H^T = \alpha^2 H M_2^2 H^T = M_1^{-1} R R^T (M_1^{-1})^T = (M_1^{-1})^2. \quad (6.6)$$

Обозначая $M_1 = \text{diag}(f_1, f_1, 1)$, $M_2 = \text{diag}(f_2, f_2, 1)$ и элементы матрицы H как h_{ij} , получим, что выражение (6.6) представляет собой серию следующих 6 уравнений (с учётом симметрии матриц 3 из 9 уравнений полностью дублируются):

$$\alpha^2 (f_2^2(h_{11}h_{21} + h_{12}h_{22}) + h_{13}h_{23}) = 0, \quad (6.7)$$

$$\alpha^2 (f_2^2(h_{11}h_{31} + h_{12}h_{32}) + h_{13}h_{33}) = 0,$$

$$\alpha^2 (f_2^2(h_{11}h_{31} + h_{12}h_{32}) + h_{13}h_{33}) = 0,$$

$$\alpha^2 (f_2^2(h_{11}^2 + h_{12}^2) + h_{13}^2) = \frac{1}{f_1^2}, \quad (6.8)$$

$$\alpha^2 (f_2^2(h_{12}^2 + h_{22}^2) + h_{23}^2) = \frac{1}{f_1^2}, \quad (6.9)$$

$$\alpha^2 (f_2^2(h_{31}^2 + h_{32}^2) + h_{33}^2) = 1,$$

где первые три уравнения соответствуют недиагональным элементам симметричного матричного равенства, а последние три уравнения соответствуют диагональным элементам. Данная система уравнений накладывает избыточное количество ограничений относительно переменных α , f_1 и f_2 , и из-за приближённого вычисления элементов матрицы H она может быть не совместной. Поэтому используют несколько решений, например, следствием из уравнения (6.7) является

$$f_1 = \sqrt{-\frac{h_{13}h_{23}}{h_{11}h_{21} + h_{12}h_{22}}},$$

а из разности уравнений (6.8) и (6.9) следует

$$f_1 = \sqrt{\frac{h_{23}^2 - h_{13}^2}{h_{11}^2 + h_{12}^2 - h_{31}^2 - h_{32}^2}}.$$



Рисунок 6.4. Пример панорамного изображения в сферической проекции (сверху) и в плоской проекции(снизу).

Из нескольких решений выбирают наилучшее с точки зрения вычислительной устойчивости к погрешности, обусловленной погрешностью оценки элементов матрицы H .

Аналогичным способом можно вычислить коэффициент f_2 матрицы M_2 . Матрицу поворота R после этого можно вычислить как $R = M_2^{-1}H^{-1}M_1$. Предложенный способ применим для любых пар изображений, однако в случае сканирования при постоянном угле θ и неизменном угле зрения (см. рис. 6.3) $f_1 = f_2 = \text{const}$, при этом удобно представлять матрицы H_1 и H_2 в виде $H_1 = MR_\theta$, $H_2 = MR_\theta R_\varphi$. В этом случае можно представить панорамное изображение в сферической проекции в виде ограниченной полосы с откладыванием широты по горизонтали и долготы по вертикали, см. рис. 6.4. Формулы перехода от точки сферы с широтой η и долготой ξ , начало отсчёта которых совпадает с началом отчётом углов наклона θ и поворота φ , к координатам

изображения x, y записываются как

$$\begin{aligned} \begin{bmatrix} \alpha x \\ \alpha y \\ \alpha \end{bmatrix} &= M R_\theta^{-1} R_\varphi^{-1} \cdot \begin{bmatrix} \cos \eta \sin \xi \\ \sin \eta \\ \cos \eta \cos \xi \end{bmatrix} = \\ &= \begin{bmatrix} f \cos(\eta + \theta) \sin(\xi + \varphi) \\ f \sin(\eta + \theta) \\ \cos(\eta + \theta) \cos(\xi + \varphi) \end{bmatrix} \Rightarrow \begin{aligned} x &= f \operatorname{tg}(\xi + \varphi), \\ y &= f \frac{\operatorname{tg}(\eta + \theta)}{\cos(\xi + \varphi)}. \end{aligned} \end{aligned} \quad (6.10)$$

Для того, чтобы высота изображения панорамы примерно соответствовала высоте изображения камеры, необходимо установить коэффициент $\tau = f/2$ для координат изображения панорамы ξ', η' , то есть $\xi' = f\xi, \eta' = f\eta$ при изображении панорамы в виде сферической проекции $\Phi(\xi', \eta')$. Как видно из рис. 6.4, сферическая проекция по сравнению с плоской обладает двумя преимуществами: во-первых, степень искажения изображения сферической проекции в целом постоянно, и не зависит от угла φ , во-вторых, область значений преобразования сферической проекции для изображения с шириной W пикселей ограничена полосой сферы $[-\pi f/2, \pi f/2] \times [f\theta - f \arctg \frac{W}{2f}, f\theta + f \arctg \frac{W}{2f}]$, область значений проективного преобразования на плоскость уходит в бесконечность при близких к числу π значениях модуля угла φ .

Для перехода от матрицы поворота R между системами координат, связанных с двумя изображениями F_1 и F_2 , к матрицам R_θ и R_φ , которые задают поворот в системе координат, связанной с поворотной платформой, заметим, что при постоянном угле $\theta = \operatorname{const}$ ось X , связанная с горизонтальным направлением изображения камеры, вращается в горизонтальной плоскости вращения поворотной платформы. Если \mathbf{e}_x и \mathbf{e}_y — единичные орты, соответственно, осей X и Y , тогда $R\mathbf{e}_x$ — вектор единичного орта, соответствующий новому положению оси X для системы координат, связанной с новым положением камеры. Следовательно, направляющий вектор оси вращения можно определить как векторное произведение $\mathbf{r} = [\mathbf{e}_x \times R\mathbf{e}_x]$, угол поворота выражается через скалярное произведение единичных векторов как

$$\varphi = (\widehat{R\mathbf{e}_x}, \mathbf{e}_x) = \arccos(R\mathbf{e}_x, \mathbf{e}_x),$$

а угол наклона выражается через смешанное произведение векторов как

$$\theta = (\widehat{\mathbf{r}}, \mathbf{e}_z) = \arccos \frac{(\mathbf{r}, \mathbf{e}_z)}{|\mathbf{r}|} = \arccos \frac{(\mathbf{e}_x, R\mathbf{e}_x, \mathbf{e}_z)}{|\sin \varphi|}.$$

Данная формула является вычислительно неустойчивой при небольших углах поворота φ , поэтому при построении панорамы во время движения следует некоторое время

перед стартом построения собирать статистику, которая позволяет определить угол θ более точно как за счёт увеличения числа отсчётов измеряемых параметров видеокамеры, так и за счёт увеличения разности углов φ между соседними положениями видеокамеры.

Таким образом, и внешние параметры φ , θ , и внутренний параметр f могут быть автоматически вычислены в процессе сканирования сцены за счёт выделения характерных точек изображения. Однако панорамное изображение может быть построено также и в том случае, когда информация приборных датчиков углов поворота платформы и текущей настройки фокусного расстояния видеокамеры может быть получена внешними системами, причём данная информация должна быть синхронизирована с видеопотоком изображений камеры. В этом случае для построения панорамного изображения достаточно показания датчиков подставить в формулы сферического проектирования (6.10). При этом достаточно высоки требования к точности показаний датчиков: для того, чтобы наложение соседних изображений друг на друга не превышало смещение в одну точку, необходимо, чтобы выполнялись следующие соотношения:

$$\begin{aligned} |\Delta x| &\approx \left| \frac{\partial x}{\partial \theta} \right| |\Delta \theta| = 0 \text{ при варьировании } \theta, \\ |\Delta x| &\approx \left| \frac{\partial x}{\partial \varphi} \right| |\Delta \varphi| < 1 \Rightarrow |\Delta \varphi| < \frac{\cos^2(\xi + \varphi)}{f} \text{ при варьировании } \varphi, \\ |\Delta x| &\approx \left| \frac{\partial x}{\partial f} \right| |\Delta f| < 1 \Rightarrow |\Delta f| < |\operatorname{ctg}(\xi + \varphi)| \text{ при варьировании } f, \\ |\Delta y| &\approx \left| \frac{\partial y}{\partial \theta} \right| |\Delta \theta| < 1 \Rightarrow |\Delta \theta| < \frac{\cos^2(\eta + \theta) |\cos(\xi + \varphi)|}{f} \text{ при варьировании } \theta, \\ |\Delta y| &\approx \left| \frac{\partial y}{\partial \varphi} \right| |\Delta \varphi| < 1 \Rightarrow |\Delta \varphi| < \frac{\cos^2(\xi + \varphi) |\operatorname{ctg}(\eta + \theta)|}{f \sin(\xi + \varphi)} \text{ при варьировании } \varphi, \\ |\Delta y| &\approx \left| \frac{\partial y}{\partial f} \right| |\Delta f| < 1 \Rightarrow |\Delta f| < |\cos(\xi + \varphi)| |\operatorname{ctg}(\eta + \theta)| \text{ при варьировании } f. \end{aligned}$$

Далее мы будем рассматривать класс оптических приборов, у которых и горизонтальный, и вертикальный углы зрения объективов менее 90° . Если у камер больший угол зрения, то такие камеры сами по себе являются панорамными, задача построения панорамы для них не столь актуальна, и должна решаться отдельно. Ограничение угла зрения величиной 90° фактически означает, что

$$-\frac{\pi}{4} \leq \xi + \varphi \leq \frac{\pi}{4}, \quad -\frac{\pi}{4} \leq \eta + \theta \leq \frac{\pi}{4},$$

поэтому обозначив $D_\xi^\phi = \{\xi | -\frac{\pi}{4} - \varphi \leq \xi \leq \frac{\pi}{4} - \varphi\}$ и $D_\eta^\theta = \{\eta | -\frac{\pi}{4} - \theta \leq \eta \leq \frac{\pi}{4} - \theta\}$,

можно вычислить, что

$$\begin{aligned} \inf_{D_\xi^\varphi} |\cos(\xi + \varphi)| &= \frac{1}{\sqrt{2}}, & \inf_{D_\xi^\varphi} |\operatorname{ctg}(\xi + \varphi)| &= 1, & \inf_{D_\xi^\varphi} \cos^2(\xi + \varphi) &= \frac{1}{2}, \\ \inf_{D_\xi^\varphi} \frac{1}{\sin(\xi + \varphi)} &= \sqrt{2}, & \inf_{D_\eta^\theta} \cos^2(\eta + \theta) &= \frac{1}{2}, & \inf_{D_\eta^\theta} |\operatorname{ctg}(\eta + \theta)| &= 1. \end{aligned}$$

Выбирая данные максимально сильные ограничения в областях D_ξ^φ и D_η^θ для полученных ранее неравенств, определяющих требуемую точность датчиков, путём замены функций на их нижние грани области значений на введённых областях определения, получим

$$|\Delta\varphi| < \frac{1}{2f}, \quad |\Delta\theta| < \frac{1}{2\sqrt{2}f}, \quad |\Delta f| < \frac{1}{\sqrt{2}}.$$

Как правило, углы зрения меньше, чем 90° , поэтому с точностью до порядка можно считать, что погрешность датчика углов поворота не должна быть больше, чем $1/f$, а погрешность датчика, позволяющего определить параметр f , не может быть более одной градации, то есть одного пикселя, так как именно в точках изображения изменяется данный параметр. Значение f привязано к разрешению изображения $W \times H$, фокусное расстояние оптической системы видеокамеры определяется по формуле

$$f' = \frac{f\omega}{W},$$

где ω — горизонтальный размер чувствительной матрицы, см. таблицу 5.2. Самые распространённые на данный момент форматы чувствительных матриц обладают шириной $\omega = 5$ мм. Это значит, что при разрешении 640×480 одна градация параметра f соответствует изменению f' на $\omega/480 \approx 10^{-2}$ мм, и поскольку минимальное фокусное расстояние большинства объективов превышает 3 мм, то для обеспечения точности в одну градацию для параметра f датчик фокусного расстояния должен обеспечивать, чтобы относительная погрешность измерения была не более 0,003 для короткофокусных объективов, причём чем больше фокусное расстояние у оптической системы, тем обратно пропорционально ниже значение допустимой относительной погрешности. Допустимую погрешность измерения углов можно оценить по порядку величины, если принять, что наиболее частый сценарий использования видеокамер на поворотной платформе предполагает, что f и ширина изображения W — это величины одного порядка, поэтому $f \approx 640$ для изображения 640×480 и допустимая абсолютная погрешность датчика угла должна быть менее $1/f \approx 0,002$ рад $\approx 0,1^\circ$.

Погрешность датчиков поворота $0,1^\circ$ и относительная погрешность 0,003 для измерения фокусного расстояния является предельными для поворотных видеокамер, известных автору данной работы. При увеличении разрешения обрабатываемого изображения, например, до распространённого на данный момент формата 1920×1080 , а также при увеличении фокусного расстояния требуется, чтобы погрешность была

обратно пропорционально меньше, и такие требования уже не могут быть выполнены современными поворотными видеокамерами. Более того, показания датчиков углов и фокусного расстояния должны быть синхронизированы с видеопотоком, однако современное состояние таково, что для большинства распространённых поворотных устройств с датчиками поворота расхождение в синхронизации информации, поступающей от датчиков, и видеопотока достигает 0,3 сек. При скорости сканирования в $30^{\circ}\text{сек}^{-1}$ в течении 0,3 сек будет пройдено примерно 10° , что будет носить существенный вклад в погрешность определения текущего угла φ на кадре. Поэтому время синхронизации не должна превышать 0,01 сек, только в этом случае погрешность определения угла, обусловленная сопоставлением показаний датчика угла поворота с кадрами видеопотока по времени получения показаний и времени захвата изображения на чувствительной матрицы, будет сопоставима по порядку величины с 0,1°.

Таким образом, современные датчики в поворотных видеокамерах не настолько хорошо откалиброваны и синхронизированы с видеопотоком, чтобы можно было отказаться от анализа характерных точек изображения в процессе автоматического построения панорамы при сканировании наблюдаемой сцены. Тем не менее, показания датчиков поворота могут быть использованы для повышения эффективности работы алгоритма поиска преобразования сопряжения двух изображений с общей зоной обзора, рассмотренного ранее в данном параграфе. Представим, что в каждый момент известны значения φ , θ и f с погрешностью, соответственно, $\Delta\varphi$, $\Delta\theta$ и Δf .

Во-первых, это позволит на начальном этапе построения панорамы, на котором требуется определить текущие угол θ и параметр фокусировки f , отбросить заведомо неверные оценки данных параметров, которые вполне могут получиться в результате псевдослучайного алгоритма RANSAC, используемого при оценке внутренних и внешних параметров видеокамеры.

Во-вторых, знание текущего положения φ и θ с точностью до $\Delta\varphi$ и $\Delta\theta$ позволяет отбрасывать заведомо неправильные пары по пространственному положению. Чем меньше погрешность определения углов, обусловленная погрешностью синхронизации и погрешностью датчиков поворота, тем пропорционально меньше точек приходится сопоставлять друг с другом на паре изображений, что влияет и на скорость работы алгоритма поиска преобразования сопряжения, и на его точность, поскольку при дополнительных ограничениях вероятность возникновения ошибочных паросочетаний характерных точек уменьшается.

В-третьих, погрешность показания датчиков углов поворота может быть уменьшена с помощью программной калибровки. Будем считать, что сканирование местности при помощи видеокамеры на поворотной платформе осуществляется при постоянном увеличении $f = \text{const}$ и при постоянном угле наклона $\theta = \text{const}$, а угол φ изменяет-

ся с постоянной по модулю скоростью вращения $|\dot{\varphi}| = \text{const}$ между углами φ_n и φ_k , соответствующие начальному и конечному положению видеокамеры в секторе сканирования. Если измерить скорость $\dot{\varphi}$, и время t_n начала смещения из точки φ_n , то при известном текущем времени t можно достаточно точно оценить угол по формуле

$$\varphi(t) = t_n + \dot{\varphi}t. \quad (6.11)$$

Чтобы стало возможным использовать эту формулу, в рамках данной работы создана специальная процедура калибровки, позволяющая автоматически определять скорость движения камеры $|\dot{\varphi}|$ и состоящая из следующих шагов.

1. С помощью команд поворотной видеокамеры, рассмотренных в параграфе 2.6, а именно: ПОВЕРНУТЬ(), УВЕЛИЧЕНИЕ(), ОСТАНОВИТЬ() — оператор системы выбирает начальное положение камеры θ , φ_n , f . Углы и фокусное расстояниечитываются с датчиков видеокамеры с помощью команды ЧТЕНИЕ_КООРДИНАТ($\varphi_n, \theta, \gamma, f$), где γ — параметр фокусировки.
2. С помощью выбранной для сканирования скорости оператор смещает камеру с помощью команды ПОВЕРНУТЬ(влево, v_φ , 0) (направление и индекс скорости v_φ выбраны для определённости) до положения φ_k , которое считывается с помощью команды ЧТЕНИЕ_КООРДИНАТ($\varphi_k, \theta, \gamma, f$). Во время движения автоматически строится панорамное изображение путём наложения в сферической проекции последовательных кадров видеопотока. Это позволяет предварительно определить, достаточно ли на сканируемой сцене текстурных элементов для определения преобразования сферической проекции в автоматическом режиме с помощью характерных точек, а также зафиксировать изображения в окрестности точек φ_n и φ_k для дальнейшего анализа.
3. Дальнейшая процедура выполняется без участия оператора, при этом многократно выполняются следующие действия в цикле с индексом i :

- выполняется команда ВЫЗОВ_КООРДИНАТ($\varphi_n, \theta, \gamma, f$);
- запускается автоматическое сканирование с помощью команды ПОВЕРНУТЬ(влево, v_φ , 0);
- после отсылки команды при помощи анализа изменений изображения фиксируется время фактического начала движения изображения t_n^i , которое не совпадает с временем подачи команды из-за задержек синхронизации;
- во время сканирования постоянно выполняется построение панорамного изображения совместно с проверкой, не является ли текущее изображение похожим на те, которые были получены на шаге 2 в окрестности, соответствующем углу поворота φ_k ;

- если проверка достижения конца сектора прошла отрицательно, движение продолжается;
- в случае, если предыдущая проверка прошла положительно, вызывается команда `ОСТАНОВИТЬ()`, затем изображение анализируется до тех пор, пока в последовательности кадров не прекращаются изменения, связанные с перемещением видеокамеры, после этого фиксируется время остановки t_κ^i и фактический угол φ_κ^i , который будет отличаться от φ_κ из-за задержек синхронизации видеопотока и процесса управления поворотным устройством, но который при этом можно вычислить, вычисляя описанным ранее способом преобразование сопряжения с кадрами, полученными во время ручного сканирования в окрестности φ_κ .

4. Если цикл на предыдущем шаге выполнялся N раз, то мы получим ровно столько оценок для скорости

$$\dot{\varphi}_i = \frac{\varphi_\kappa^i - \varphi_\kappa}{t_\kappa^i - t_\kappa^i},$$

поэтому можно уменьшить вклад случайной погрешности, обусловленной погрешностью синхронизации, в \sqrt{N} раз, вычислив выборочное среднее

$$\dot{\varphi} = \frac{1}{N} \sum_{i=1}^N \dot{\varphi}_i. \quad (6.12)$$

Таким образом, калибровка, проведённая по описанному выше алгоритму, позволяет в целом устранить погрешность оценки текущего угла поворота φ за счёт достаточно точного вычисления скорости движения. За счёт многократности измерений в том числе возможно несколько повысить чувствительность процесса измерения при помощи датчиков угла поворота.

Благодаря более точной оценке угла примерно в 3–10 раз сокращается время работы алгоритма по поиску пар точек на соседних изображениях по методу поиска паросочетаний с помощью жадного алгоритма с применением хэширования при поиске ближайшего соседа, так как сложность алгоритма $O(Q \log Q)$ почти линейно зависит от числа возможных пар Q . Кроме того, в несколько раз снижается вероятность ошибочного поиска преобразования с применением алгоритма RANSAC, поскольку из возможных пар исключаются те пары, которые не лежат в пределах погрешности вычисления угла поворота $\Delta\varphi$ в сферической проекции. Ускорение процесса построения панорамы с помощью данных, полученных при калибровке скорости, возможно также за счёт того, что уточнение текущего положения φ можно делать не для всех кадров видеопотока, а только для каждого второго, или каждого третьего, а для промежуточных кадров оценивать угол поворота по формуле (6.11).

Изображение видеокамер может зависеть от времени суток и от погодных условий. Если по каким-то причинам снижается контрастность изображения в силу внешних условий, то поиск преобразования с помощью характерных точек становится невозможен, и единственным способом оценить текущее положения камеры является оценка по скорости, полученной по формуле (6.12). Поэтому предварительная калибровка поворотных устройств, разработанная в рамках данной работы, имеет большое значение для эффективности, быстродействия и качества алгоритма построения панорамного изображения в процессе сканирования видеокамерой, установленной на поворотной платформе.

6.4. Разработка и исследование алгоритмов семантического выделения подвижных объектов на панорамных изображениях

Предыдущие параграфы описывали процесс построения панорамного изображения, при этом показан ряд приёмов, повышающих быстродействие и качество наложения изображений на панораму во время сканирования наблюдаемой сцены с помощью поворотной видеокамеры. Применяя совместно алгоритмы семантического выделения подвижных объектов, рассмотренные в п. 1.3 и в п. 2.1, и алгоритмы построения панорамных изображений, можно выделить подвижные объекты на наблюдаемой сцене. Поскольку изображение сцены на панораме не смещается по мере движения поворотной платформы видеокамеры, вполне применим подход, заключающийся в накоплении панорамного фона с последующим вычитанием из фона текущего изображения и сегментацией областей с высокой нормой разности текущего изображения и фона. При таком подходе возникают две основные проблемы.

Во-первых, алгоритм построения панорамы является вычислительно ёмким, а для корректной обработки изображения с помощью алгоритма вычитания фона требуется высокая частота обработки кадров, как правило 25 к/с. То есть построение панорамы необходимо выполнять не более, чем за $1/25$ сек = 40 мсек, причём следует учесть, что вычислительные ресурсы нужны не только для алгоритма построения панорамы, но и для алгоритма выделения подвижных объектов. Для изображения 640×480 описанные ранее алгоритмы выделения движения выполняются примерно 5–10 мсек для одного кадра видеоследовательности, поэтому построение панорамы должно укладываться в 20–30 мсек. Устранив за счёт известной траектории движения камеры квадратичную зависимость вычислительной сложности алгоритма [187] от числа изображений, участвующих при построении панорамы, заменив алгоритм поиска характерных точек на более быстрый (см. п. 6.2), и воспользовавшись при формировании пар характерных точек уточнённой оценкой углов поворота за счёт предварительной калибровки (см. п. 6.3), возможно получить алгоритм построения панорамы, который обрабатывает одно изображение за 90–120 мсек вместо исходных

10 сек. Тем не менее, для реализации сопровождения подвижных объектов на панорамных изображениях, необходим способ, который примерно в 3–4 раза сократит время обработки одного кадра панорамного изображения с последующим выделением движущихся объектов.

Во-вторых, при наложении изображения на текущее панорамное изображение не всегда удается добиться полного совмещения изображения с точностью до одного пикселя. Как показывает практика, погрешность наложения может составлять 1–2 пикселя. При формировании визуального отображения панорамы в работе [187] предлагаются использовать алгоритмы формирования плавного перехода от одного изображения к другому, однако отсутствие визуальных недостатков на панораме еще не означает, что алгоритм вычитания фона корректно обработает изображения при смещении фона и текущего изображения на 1–2 пикселя. Практика показывает, что из-за ошибок наложения кадров значительно возрастает число ошибок первого и второго рода при идентификации движущихся на сцене наблюдения объектов, поэтому требуется доработка подхода, основанного на сопоставлении текущего кадра и накопленного фонового изображения.

Для решения проблемы быстродействия алгоритмов семантической обработки панорамных изображений в рамках данной работы используется параллелизм современных вычислительных устройств и приборов. Рассмотрим граф-программу, или граф управления с элементами информационного графа в терминах труда [202] для алгоритма семантического анализа изображений сцены во время сканирования, см. рис. 6.5. На рисунке граф состоит из вершин с сокращенными названиями процедур: ПХТ — поиск характерных точек, ПП — поиск пар точек, ПС — поиск преобразования сопряжения, ВИП — вложение изображения в панораму, ВФ — выделение и вычитание фона, ПТР — поиск траекторий движущихся объектов. В качестве входных параметров перечисленных выше процедур на рисунке указаны последовательные кадры видео $F_{i-1}(x, y)$, $F_i(x, y)$, характерные точки $\{u_m\}$ и $\{v_n\}$ на изображениях $F_{i-1}(x, y)$ и $F_i(x, y)$, соответственно, отобранные пары точек $\{(u_k, v_k)\}$, параметры камеры: фокусное расстояние f , угол наклона θ и угол поворота φ_i для текущего кадра $F_i(x, y)$, панорамное изображение для текущего кадра $\Phi_i(\xi, \eta')$, множество областей движения $R_i = \{c_i^r\}$ и описание траекторий объектов $T_i = \{t_i^o\}$ для i -го кадра. Обозначения для процедуры построения панорамы заимствованы из параграфа 6.3, а обозначения для множеств областей движения и траекторий уже использовались ранее в параграфе 5.2 (следует отметить, что в данном случае области c_i^r состоят из множества точек изображений, а не только из точки центра масс).

Заметим, что при обработке кадров F_{i-1} и F_i процедуры ПХТ, ПП и ПС не требуют информации, полученной в ходе выполнения остальных процедур, в том числе им не

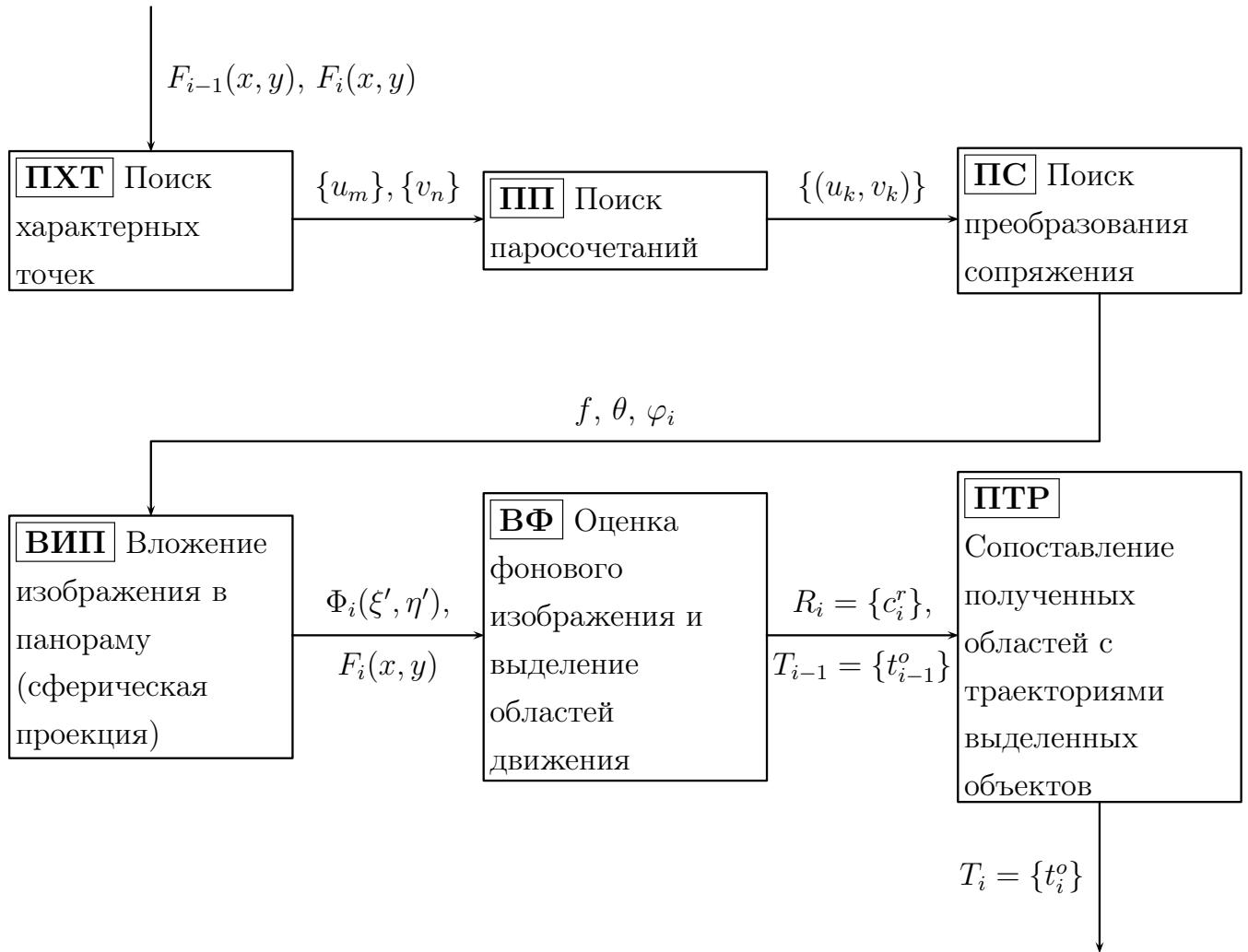


Рисунок 6.5. Граф-программа построения панорамного изображения и выделения на нём движущихся объектов (ПХТ — поиск характерных точек, ПП — поиск пар точек, ПС — поиск преобразования сопряжения, ВИП — вложение изображения в панораму, ВФ — выделение и вычитание фона, ПТР — поиск траекторий движущихся объектов).

нужна информация о характерных точках и преобразовании поворота, полученная для предшествовавших кадров видеопотока. Это позволяет применить параллельно-конвейерный подход к ускорению обработки панорамных изображений.

Измерение времени выполнения программы построения панорамы на одном ядре современного компьютера архитектуры Intel Core i7 показывает результаты, отображённые в таблице 6.1. Таблица показывает, что суммарно процедуры ПХТ, ПП и ПС, для которых отмечена независимость результата работы от любой информации, кроме изображений F_i и F_{i-1} при обработке i -го кадра, выполняются на одном ядре 80–110мс, а остальные процедуры ВИП, ВФ и ПТР, результаты работы которых зависят как от результата работы процедур выделения и обработки характерных точек, так и от последовательности поступающих кадров, суммарно выполняются за 15–20мс, в 3–5 раз быстрее, чем независимые процедуры.

Исходя из соотношения времени выполнения процедур, является целесообразным

Название процедуры	Аббревиатура	Время, мсек
Поиск характерных точек	ПХТ	20
Поиск пар точек	ПП	20–30
Поиск сопряжения	ПС	40–60
Вложение в панораму	ВИП	5
Вычитание фона	ВФ	5
Построение траекторий	ПТР	5–10

Таблица 6.1. Времена обработки данных отдельными процедурами анализа панорамных изображений при выполнении с использованием одного ядра процессора Intel Core i7.

построить вычислительный процесс обработки панорамных изображений на основе четырёх конвейеров, три из которых должны принимать на вход кадры $F_i(x, y)$, выполнять последовательно процедуры ПХТ, ПП, ПС и выдавать оценку угла φ_i , а один конвейер должен принимать на вход изображение $\Phi_i(\xi', \eta')$, кадр F_i и соответствующий ему угол φ_i и выполнять для этих данных процедуры ВИП, ВФ, ПТР. На выходе данного конвейера формируется конечный результат в виде траекторий объектов T_i .

Обозначим алгоритм выполняемых последовательно процедур ПХТ, ПП и ПС символом \mathcal{P} , а алгоритм выполняемых последовательно процедур ВИП, ВФ, ПТР символом \mathcal{T} . Если трактовать алгоритм как функцию, которую можно вычислить над входными данными за конечное число шагов, то имеет место

$$(\varphi_i, \{u_i\}, \{v_i\}) = \mathcal{P}(F_{i-1}, F_i, \{v_{i-1}\}), \quad \{u_i\} = \{v_{i-1}\},$$

$$(\Phi_i, T_i, S_i) = \mathcal{T}(\varphi_i, F_i, \Phi_{i-1}, T_{i-1}, S_{i-1}),$$

где S_i — некоторое внутреннее состояние алгоритма \mathcal{P} , связанное, в частности, с данными о накопленном фоновом изображении. Стадией конвейера будем считать трёхкратное вычисление приведённых выше функций для данных с индексами $i = 3s + p$, где s — номер стадии, а p — целочисленный индекс операции внутри стадии от 0 до 2. Таким образом, за время одной стадии три конвейера параллельно выполняют алгоритм \mathcal{P} , и один конвейер последовательно выполнит три операции \mathcal{T} .

Для того, чтобы все конвейеры, выполняющие алгоритмы \mathcal{P} и \mathcal{T} , были постоянно загружены, необходимо ввести отставание обработки данных конвейером, выполняющим \mathcal{T} , на одну стадию по сравнению с тремя конвейерами, выполняющими \mathcal{P} . Тогда данные $\varphi_{3(s-1)+p}$ с предыдущей стадии конвейеров, выполняющих \mathcal{P} , поступают на вход текущей стадии конвейера, выполняющего операцию \mathcal{T} , и одновременно с этим три конвейера, выполняющие \mathcal{P} , начнут готовить данные φ_{3s+p} для работы следующей стадии. Графическая схема описанной работы конвейера представлена на рис. 6.6. Отметим, что результат с траекториями выделенных движущихся объек-

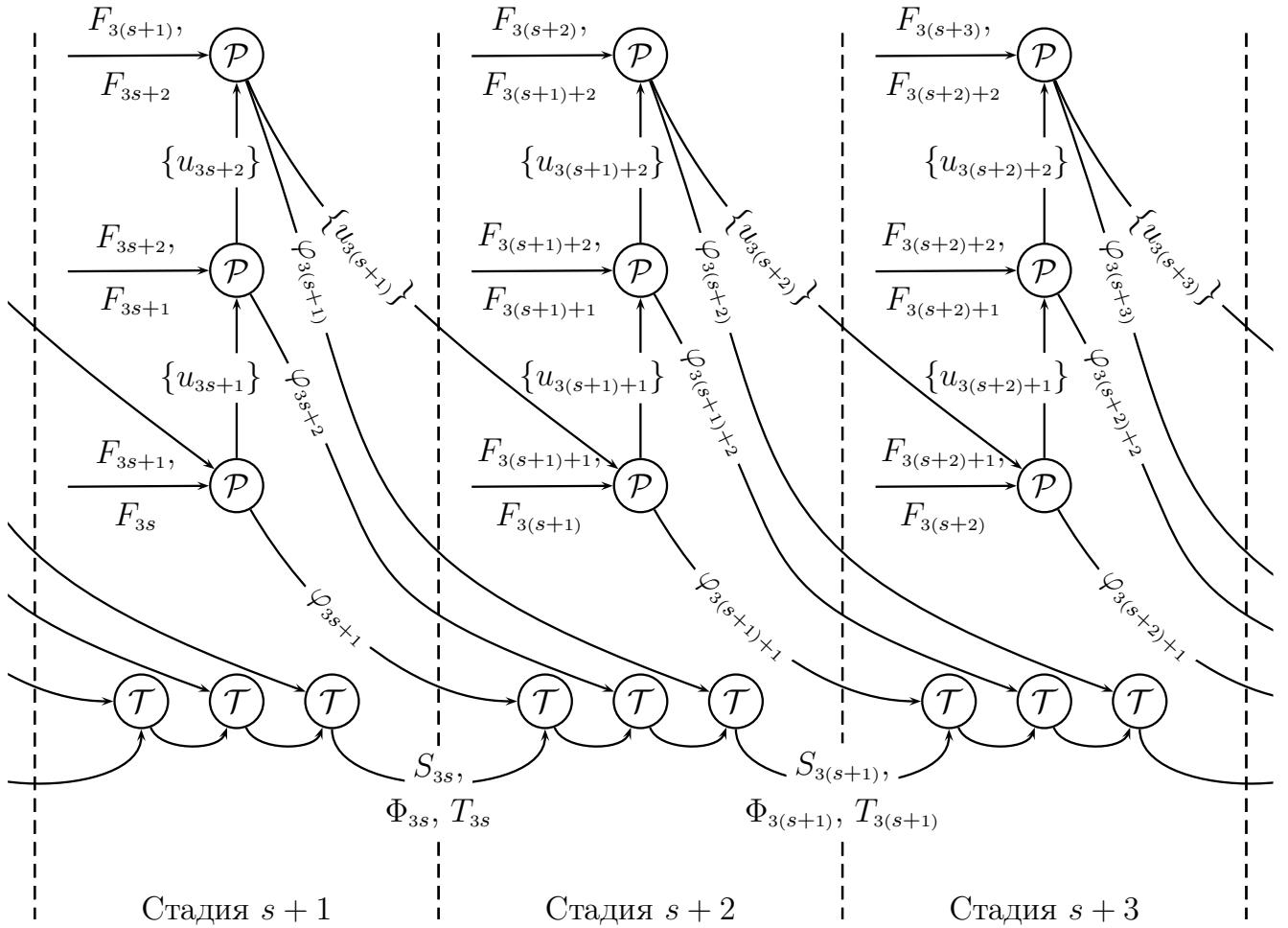


Рисунок 6.6. Схема параллельно-конвейерного построения и семантической обработки панорамных изображений.

тов T_i будет выдан конвейером с отставанием по сравнению с поступлением кадров с изображением объектов, так как для загрузки конвейера следует ввести очередь ожидания обработки кадров с длиной как минимум 3. В тот момент, когда новый кадр поступает в данную очередь, результат обработки будет готов только для кадра с меньшим на 3 индексом. При частоте поступления кадров 25 к/сек задержка составит около 160 мсек, что сопоставимо со средним временем самой быстрой моторной реакции у людей в ответ на какие-либо нервные раздражители, и, следовательно, такая задержка не должна создавать ощутимого дискомфорта для работы операторов систем видеонаблюдения.

Конвейер, схема которого представлена на рис. 6.6, за одну стадию обрабатывает 3 кадра одновременно. При этом конвейер, последовательно выполняющий 3 операции \mathcal{T} , будет действовать 45–80 мсек в течение одной стадии, а каждый из трёх конвейеров, выполняющих операцию \mathcal{P} , будет действовать 80–110 мсек в течение одной стадии, то есть конвейер \mathcal{T} будет, как правило, недозагружен. При этом каждая стадия обрабатывает 3 кадра подряд, поэтому среднее время обработки одного кадра пред-

ставленным конвейером будет 30–40 мсек, что почти соответствует заявленной цели достигнуть 20–30 мсек по времени обработки одного кадра. По крайней мере, с таким временем работы системы сможет обрабатывать все кадры видеопоследовательности с частотой 25 кадров в секунду.

Отметим, что большинство современных операционных систем для самых распространённых вычислительных устройств в целом и комплексы видеосистем в частности не являются системами реального времени в том смысле, что существует довольно большая неопределённость во времени исполнения вычислительных процедур и процедур ввода-вывода. Например, современные сетевые видеокамеры могут для оптимизации скорости передачи видеопотока по локальной вычислительной сети формировать пакеты из нескольких кадров подряд, задерживая таким образом передачу кадров на величину объёма буфера. В таких условиях более целесообразно рассматривать конвейерный процесс обработки как набор асинхронных процессов. Синхронизация между ними будет осуществляться на уровне очереди из входных данных: кадров F_i для процессов \mathcal{P} и углов φ_i для процессов \mathcal{T} : если очередь из входных данных на одном из конвейеров сильно выросла, значит, пора пропустить кадры в очереди на обработку, чтобы сократить разность скорости выдачи выходных данных по сравнению со скоростью входных данных. Если конвейеры, выполняющие операцию \mathcal{P} , являются узким местом, то вместо тяжёлой обработки изображения для оценки угла φ_i можно использовать элементарную формулу (6.11), при условии, что для камеры проведена калибровка и время кадра t_i известно достаточно точно. Это позволит исключить кадр F_i из обработки довольно длительной операцией \mathcal{P} , при этом, тем не менее, получить оценку угла поворота на основе интерполяции. Переход от полной синхронизации стадий к асинхронной схеме обработки позволяет гарантировано обработать изображения камер, установленных на поворотной платформе, во время сканирования с частотой обработки кадров 25 кадров в секунду, что соответствует типичной частоте генерации кадров для большинства современных видеокамер.

Проблема погрешности наложения изображений камеры на панорамное изображение, отмеченная в начале данного параграфа, приводит к тому, что появляется множество шумовых областей движения, особенно на границах неподвижных объектов сцены, поскольку смещение на 1–2 пикселя при наложении приводит к наибольшим различиям от фона в областях с высоким градиентом функций $F_i(x, y)$, так как границы смещаются друг относительно друга. Для решения этой проблемы предлагается два программных фильтра.

Один фильтр заключается в том, что кроме отслеживания движущихся областей можно производить отслеживание движущихся особых точек. И если в окрестности области движения c_i^r нет подвижных характерных точек, значит эту область не сле-

дует рассматривать как кандидата на формирование позиции одного из объектов во множестве T_i . Для отслеживания движения характерных точек можно использовать результаты работы процедур \mathcal{P} .

Второй фильтр основан на семантическом анализе формы областей $\{c_i^r\}$ и на том факте, что как правило, области движения, образованные из-за смещения границ, представляют собой тонкие, длинные, и почти прямые линии. Поэтому, если взять произвольную не очень большую окрестность $U_\varepsilon(x, y)$ (например, $\varepsilon = 16$) для любой точки $(x, y) \in c_i^r$, то для тонких областей собственные значения матрицы вторых моментов, посчитанных для точек $c_i^r \cap U_\varepsilon(x, y)$, будут значительно отличаться друг от друга, причём наименьшее значение будет достаточно маленьким и соответствовать квадрату ширины тонкой линии. Если почти все точки области c_i^r движения удовлетворяют таким соотношениям для собственных чисел вторых моментов, рассчитанных в некоторой окрестности, значит эту область также не следует рассматривать как кандидата для продолжения объектов во множестве T_i .

Предложенные фильтры позволили на два порядка сократить количество ошибок первого рода $F_N^{\alpha\beta}$ (см. определение в п. 4.2) на два порядка. Однако частота ложных срабатываний для приведённого метода пока достаточно велика и составляет единицы в несколько минут. Доля ошибок второго рода $F_P^{\alpha\beta}$ по отношению ко всем объектам составляет примерно 0,9. Таким образом, разработанный параллельно-конвейерный алгоритм позволяет проводить семантическую обработку панорамных изображений в реальном времени. Системы, построенные с использованием данного алгоритма, могут использоваться как вспомогательные детекторы движущихся объектов. Однако для использования сканирующих видеокамер в качестве основного детектора на данный момент требуется дальнейшая проработка теории обработки панорамных изображений.

6.5. Анализ алгоритмов сопряжения панорамного изображения с планом местности

Эффективность современных систем видеонаблюдения определяется тем, насколько просто и удобно они позволяют операторам контролировать события, происходящие в поле зрения видеокамер. Многие видеосистемы упрощают ситуационный контроль в поле зрения видеокамер за счёт автоматической обработки изображений и распознавания в видеосигнале событий интереса. В подобных системах любое событие, определённое с помощью автоматического анализа видеосигнала, отображается в виде сообщения оператору. Для повышения информативности такого сообщения автор данной статьи предложил отображать место события на плане местности, проведя предварительно процедуру калибровки, устанавливающую связь между координатами изображения камеры с координатами плана местности, см. [129] и п. 2.4. В дан-

ных работах предполагалось, что изображение, которое требуется привязать к плану местности, получено с помощью обычных видеокамер с прямоугольными матрицами чувствительных элементов и объективами, которые можно с хорошим приближением считать тонкими линзами. Для систем, в которых для контроля области наблюдения используются поворотные видеокамеры в режиме непрерывного сканирования, не подходит модель видеокамеры с тонкой линзой, поскольку с их помощью формируется панорамное изображение наблюданной территории, которое для удобства представления отображается с помощью сферической проекции. Поэтому механизмы, применяемые для построения преобразования координат изображения в координаты плана местности, необходимо модифицировать с учётом отклонения от модели плоской линзы и плоской матрицы ПЗС.

Для построения панорамы используется исходная декартова система координат $OXYZ$, центр которой совпадает с центром сферической проекции, ось Z направлена вдоль оптической оси камеры в положении, когда угол наклона камеры к плоскости вращения равен 0, и находится в стартовом положении панорамы, т.е. нулевом угле поворота относительно оси вращения. Если матрица камеры

$$K = \begin{bmatrix} f & 0 & W/2 \\ 0 & f & H/2 \\ 0 & 0 & 1 \end{bmatrix},$$

где $W \times H$ – разрешение камеры и f – характеристика угла обзора, если угол наклона оптической оси камеры к плоскости, перпендикулярной оси вращения, равен θ , а угол поворота камеры вокруг оси вращения равен φ , то для координаты изображения (x, y) (при условии проектирования в плоскость на расстоянии f от центра вращения) переводятся в исходную систему координат $OXYZ$ по формуле

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = RK^{-1} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \text{ где } R = R_{-\varphi}R_{-\theta} \text{ —матрица поворота,}$$

$$R_{-\theta} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & \sin \theta \\ 0 & -\sin \theta & \cos \theta \end{bmatrix}, \quad R_{-\varphi} = \begin{bmatrix} \cos \varphi & 0 & -\sin \varphi \\ 0 & 1 & 0 \\ \sin \varphi & 0 & \cos \varphi \end{bmatrix}.$$

Панорамное изображение мы получаем в пространстве сферических координат $(u, v) = S(X, Y, Z)$:

$$u = f \cdot \arctg \frac{X}{Z},$$

$$v = f \cdot \left(\pi - \arccos \frac{Y}{\sqrt{X^2 + Y^2 + Z^2}} \right).$$

Обратное преобразование $(X, Y, Z) = S^{-1}(u, v)$ описывается формулами

$$\begin{aligned} X &= \sin\left(\pi - \frac{v}{f}\right) \sin \frac{u}{f}, \\ Y &= \cos\left(\pi - \frac{v}{f}\right), \\ Z &= \sin\left(\pi - \frac{v}{f}\right) \cos \frac{u}{f}. \end{aligned}$$

Способ связи изображения с планом местности, описанный в работах [129, 130] и в параграфе 2.4, можно применить с модификацией: опорные точки, которые пользователь задаёт на панорамном изображении, необходимо с помощью преобразования $K \cdot S^{-1}(u, v)$ привести к исходным декартовым координатам изображения, и уже для данных координат найти проективные соотношения с координатами плана местности. Таким образом, для преобразования между изображением панорамы и плана местности, заданного на сетке $(u^i, u^i) \leftrightarrow (x_m^i, y_m^i)$, мы найдём сетку $(x^i, y^i) \leftrightarrow (x_m^i, y_m^i)$, имея ввиду $(x^i, y^i) \cong K^{-1} \cdot S(x^i, y^i)$, и пользуясь описанными ранее методами, найдём преобразование координат $(x, y) \leftrightarrow (x_m, y_m)$. Если $H(x, y)$ – найденное преобразование из координат изображения в систему координат карты (x_m, y_m) , то искомое преобразование из координат изображения панорамы на карту будет иметь вид

$$(x_m, y_m) \cong H(K \cdot S^{-1}(u, v)),$$

а преобразование из точки (x_m, y_m) карты на изображение панорамы будет

$$(u, v) = S(K^{-1} \cdot [\alpha H^{-1}(x_m, y_m) | \alpha]^T),$$

где столбец $[\alpha H^{-1}(x_m, y_m) | \alpha]^T = [\alpha x \quad \alpha y \quad \alpha]^T$ составлен из двумерных однородных координат, соответствующих результату преобразования

$$(x, y) = H^{-1}(x_m, y_m).$$

Поскольку функция $S(X, Y, Z)$ является однородной, то есть $S(\alpha X, \alpha Y, \alpha Z)$ не зависит от α , то при вычислениях по приведённой выше формуле можно считать $\alpha = 1$, так как результат не зависит от конкретного значения этого параметра.

Из полученных соотношений следует, что для связи панорамы и плана местности без существенных неровностей рельефа достаточно задать четыре точки, связывающие изображение панорамы с изображением плана. Для учёта неровностей рельефа целесообразно использовать кусочно-линейную интерполяцию.

Таким образом, для построения преобразования с панорамного изображения на план местности необходимы данные о матрице внутренних параметров камеры и внешних параметров в форме углов поворота. Эти данные позволяют устранить нелинейность при поиске проективного преобразования.

Выводы

1. Анализ открытых реализаций программных библиотек построения панорамных изображений показал, что данные библиотеки не могут быть использованы в системах реального времени, поскольку они в среднем обрабатывают одно изображение более 1 сек.
2. Предложен новый способ для сопряжения панорамных изображений с изображениями топографических карт для эффективной визуализации сопровождаемых объектов в сканирующих видеосистемах.
3. В процессе теоретического исследования вычислительной сложности и эффективности алгоритмов сопоставления характерных областей изображений показано, что быстрые алгоритмы типа ORB можно использовать в системах обработки панорамных изображений в реальном времени, так как незначительный проигрыш в достоверности сопоставления по сравнению с более общеупотребительными алгоритмами SIFT компенсируется методами дополнительной обработки информации о характерных точках, а выигрыш в производительности у быстрых алгоритмов более чем двукратный.
4. Предложен новый алгоритм автоматической калибровки поворотных видеокамер, проведён анализ влияния точности предложенного метода оценки скорости вращения поворотного устройства на эффективность и быстродействие алгоритмов построения панорамного изображения.
5. Создан новый параллельно-конвейерный алгоритм семантической обработки изображений, полученных от установленных на поворотных платформах видеокамер в режиме сканирования. Алгоритм в реальном времени проводит идентификацию подвижных объектов на сцене, используя современные методы построения панорам и семантический анализ изображений и обрабатывая один кадр изображения с разрешением 640×480 за 50 мс, что в 20 раз меньше времени обработки для аналогичных алгоритмов в открытых источниках.

7. Экспериментальная оценка эффективности разработанных алгоритмов сопряжения, синхронизации, управления и семантического сжатия изображений. Результаты внедрения

7.1. Внедрение результатов и апробация

К основным результатам диссертационной работы относятся способы синхронизации изображений неподвижных и поворотных видеокамер, алгоритмы сопряжения изображений с планом местности, алгоритмы обработки подвижных стереоизображений и панорамных изображений, способы семантической обработки изображений. Эти результаты легли в основу таких программно-аппаратных комплексов, как система контроля безопасности объектов на основе приборов технического зрения «Orwell2k», системы контроля соблюдения правил дорожного движения «Travio» и система подсчёта плотности потока движущихся объектов «Statistics-3d». Перечисленные комплексы созданы на предприятии ГУП НПЦ «ЭЛВИС» и его правоприемнике АО «ЭЛВИС-Неотек» при непосредственном участии автора в качестве руководителя и архитектора по разработке математического программного обеспечения. Проведём анализ основных достоинств данных программно-аппаратных систем, непосредственно связанных с вкладом автора, достигнутом по результатами диссертационной работы.

1. Обработка стереоизображений позволила существенно повысить точность результата в устройствах подсчёта посетителей «Statistics-3d» с 90% до 98% (протокол испытаний данных устройств в аэропорту «Шереметьево» представлен в приложении 2).

2. Использование быстродействующих алгоритмов построения панорамных изображений позволяет полноценно использовать видеокамеры на поворотных платформах в режиме непрерывного сканирования с автоматическим выделением событий в области контроля видеокамер, обрабатывая поток кадров с частотой выше 15 кадров в секунду на ЭВМ стандартной конфигурации; данный результат для систем реального времени достигнут впервые.

3. Система автоматизированной оценки достоверности алгоритмов семантического сжатия позволила эффективно выработать дополнительные способы и алгоритмы для снижения на 1–2 порядка частоты ошибок первого и второго рода при анализе событий с помощью обработки изображений от видеокамер. Более высокая достоверность фиксации событий позволяет уменьшить количество операторов в 5–10 раз при контроле обстановки в поле зрения видеокамер системы, что показано в работах автора [204, 205].

4. Сопряжение видеокамер с планом местности позволяет в несколько раз умень-

шить необходимое количество операторов системы для контроля больших периметров наблюдения за счёт повышения эргономических характеристик терминального модуля.

5. Сопряжение неподвижных видеокамер с общей зоной обзора между собой позволяет с достоверностью более 99% сопоставлять изображения одного и того же объекта на разных камерах, что снижает общее количество сигналов об объектах на наблюданной территории и позволяет уменьшить количество операторов системы, контролирующих обстановку на территории. Кроме того, настроенные преобразования сопряжения применяются при работе аналитических видеосистем в различных модулях семантической обработки изображений в реальном времени.

6. Алгоритмы сопряжения поворотных и неподвижных видеокамер, а также алгоритмы синхронизации и наведения с упреждением на движущиеся объекты позволяют с вероятностью более 99% получать изображения повышенного разрешения для всех объектов в зоне наблюдения неподвижных видеокамер, которые сопряжены с поворотными. Разрешение изображения повышается в 5–10 раз, а применение поворотных видеокамер вместо неподвижных позволяет сократить общее количество видеокамер системы.

7. Алгоритмы сопряжения и синхронизации, разработанные в данной диссертации, применимы также для сопоставления отметок транспортных средств, выделенных специальными многоцелевыми радарами, и изображений транспортных средств на видеокамере, что позволяет использовать скорость, полученную радаром с помощью эффекта Доплера, для контроля скоростного режима. Используя сопряжение и синхронизацию показаний радара совместно с алгоритмом автоматического выделения государственных номерных знаков, система «Travio» формирует без участия оператора штрафные квитанции за превышение скорости для владельцев транспортных средств с распознанным номерным знаком.

8. Использование алгоритмов семантической обработки изображений позволяет расширить область применения аналитических видеосистем. Например, с помощью алгоритмов классификации могут быть решены следующие задачи: наблюдение за автостоянками с целью предотвращения угона, оценка соответствия количества проданных билетов в кинозалы и реального количества зрителей, выделение оставленных вещей в местах скопления людей.

Программное обеспечение для семейства программно-аппаратных комплексов «Orwell2k», «Travio» зарегистрировано в реестре программ для ЭВМ, см. приложение 1. В ходе разработки этих программных комплексов автором созданы спецификации для программ юстировки сопряжения видеокамер между собой и планом мест-

ности, спецификации для программ сопряжения неподвижных и поворотных видеокамер, создана программа обучения и тестирования систем распознавания и семантической обработки изображений.

Апробация комплексов «Orwell2k» (см. [209]) и «Travio» проведена не только на конференциях, но и на различных выставках современных технологий в области систем охранного наблюдения, см. приложение 5. На многих из этих выставок система получила высокую оценку специалистов, в том числе благодаря результатам данной работы, внедрённым в систему. Например, предприятие ГУП НПЦ «ЭЛВИС» в 2005 году за разработку «Orwell2k» было удостоено национальной отраслевой премии по безопасности «За укрепление безопасности России» («ЗУБР-2005»). Кроме того, информация о системах «Orwell2k» и «Travio» представлена, например, на интернет-странице <http://www.elvees.ru>.

Благодаря перечисленным достоинствам система нашла своё применение у многих потребителей, что подтверждается документами приложений 3 и 2. Среди этих документов представлены акты внедрения, которые подтверждают применение систем, в которых внедрены результаты данной диссертационной работы, для контроля безопасности стратегически важных инфраструктурных объектов.

Технологии, реализованные в программно-аппаратном комплексе «Orwell2k», защищены патентами [212, 213, 214, 215] и другими (см. перечень в приложении 4).

7.2. Описание схемы видеосистем с приборами и блоками семантической обработки изображений

Результаты диссертационной работы внедрены при создании семейства программно-аппаратных комплексов «Orwell2k», разработанных при непосредственном участии автора. Благодаря техническим характеристикам комплексов «Orwell2k», полученным в том числе благодаря применению сопряжения, синхронизации, алгоритмов управления поворотными камерами и алгоритмам семантического сжатия (см. п. 1.1), данные программно-аппаратные комплексы применяются для обеспечения охранных мероприятий на значимых инфраструктурных объектах, среди которых можно выделить, например, аэропорты Домодедово и Шереметьево, железную дорогу между городом Сочи и посёлком Красная поляна (охранную систему «Orwell2k» начали использовать на ней во время проведения зимних олимпийских игр 2014 года), ряд гидроэлектростанций, нефтеперерабатывающий завод Оренбургской области (договор №09.08.07(1)/Д между ГУП НПЦ «ЭЛВИС» и организацией «Ритар-Юг»), а также обширные общественные территории городов (например, в Нижнем Новгороде, Геленджике, Саранске).

Типичным примером применения результатов диссертационной работы является система промышленного видеонаблюдения на объектах воздушного транспорта



Рисунок 7.1. Схема выполнения функций в системе «СПВ ОВТ».

РАЯЖ 46652.001-ОС.ПЗ, сокращённо «СПВ ОВТ» (МАД — Московский аэропорт Домодедово). Система была разработана на предприятии ГУП НПЦ «ЭЛВИС» для выполнения государственного контракта №2037735016845, зарегистрированного 24 июля 2003 года.

«СПВ ОВТ» является комплексом программно-аппаратных средств, предназначенных выполнять ряд задач, а именно:

- 1) контроль проезда транспорта через пропускные пункты;
- 2) наблюдение за обстановкой в залах ожидания;
- 3) видеоконтроль за зонами авиационной деятельности;
- 4) контроль ограждений аэропорта с целью предотвращения проникновения, осуществляемый средствами оптического наблюдения и вспомогательными индукционными датчиками, детектирующими прикосновение к ограждению;
- 5) видеонаблюдение в ангарных помещениях с целью автоматического выявления нештатных ситуаций (например, возгорание).

Особенности данной аналитической видеосистемы описаны в источниках [208, 207, 209, 210]. На рис. 7.1 отображена схема выполнения функций системой «СПВ ОВТ» при появлении объектов в поле зрения стационарных камер. Из этой схемы следует, что время реакции на появление объектов примерно равно 0,5с, что в 20 раз меньше, чем аналогичное время для обычных аналитических систем без автоматической



Рисунок 7.2. Мачта телевизионного наблюдения.

локализации на плане местности и без автоматического наведения поворотных камер (см. рис. 1.7).

Основными приборами и оборудованием в «СПВ ОВТ» являются:

- 1) неподвижные и поворотные видеокамеры, тип которых определяется типом внешнего освещения и требуемым углом наблюдения;
- 2) ЭВМ, именуемые серверами системы и предназначенные для анализа видеосигнала и управления поворотными видеокамерами;
- 3) коммутационное оборудование, предназначенное для передачи видеосигналов, сигналов управления и передачи электропитания приборам системы;
- 4) терминалы системы видеонаблюдения;
- 5) мачты для крепления телевизионных камер, устойчивые к вибрациям, обусловленным сильными ветрами (см. рис. 7.2).

Структура системы соответствует приведённой на рис. 1.1 (с. 22). Можно выделить четыре группы видеокамер в зонах наблюдения ангар (четыре неподвижные видеокамеры и две поворотные купольные видеокамеры), зона авиационной деятельности (двенадцать неподвижных видеокамер и две видеокамеры на поворотном устройстве), зал ожидания (две неподвижные видеокамеры) и контрольно-пропускной пункт (две неподвижные видеокамеры). Видеосигнал от данных видеокамер обрабатывают пять ЭВМ-серверов. Терминалы системы расположены в двух местах: на рабочем месте оператора службы охраны, и у начальника службы безопасности охраняемого участка аэропорта.

В результате применения разработанных в диссертации алгоритмов и способов, которые приведены в таблице 7.1, в системе «СПВ ОВТ» полностью автоматизировано управление поворотными видеокамерами с целью получения изображений высокого разрешения для объектов, присутствующих в зоне наблюдения неподвижных камер, а также прикоснувшихся к ограждению зоны авиационной деятельности, производится автоматическое сопоставление синхронных изображений объектов на неподвижных

Алгоритмы	Способы
<ul style="list-style-type: none"> – наведения поворотных камер с адаптивным упреждением (патент №36912 от 23.06.2003, №2265531 от 07.08.2003); – составного перемещения поворотных камер; – сопоставления изображений объектов в общей зоне видимости нескольких неподвижных камер; – синхронизации в цикле обработки информационных сообщений мастеров; – синхронизации в циклах обработки информационных сообщений мастеров и слейвов (свидетельство о регистрации программы №2003612604); – синхронизации в цикле обработки информационных сообщений слейвов; – идентификации и классификации изображений с двумя признаками на основе быстрой триангуляции. 	<ul style="list-style-type: none"> – сопряжения неподвижных камер между собой и планом местности с помощью проективных преобразований (патент №2268497 от 23.06.2003); – юстировки камер по отношению к плану местности с помощью измерения дальности до ориентиров; – моделирования неровностей поверхности земли; – сопряжения мастеров и слейвов с фиксированными позициями; – сопряжения мастеров и слейвов с координатным управлением углами поворота; – автоматического определения координат увеличения и фокусировки слейвов при наведении на объекты в зависимости от его размеров и дальности.

Таблица 7.1. Алгоритмы и способы, разработанные в диссертационной работе видеокамерах с частично общей зоной обзора, а также автоматическая классификация, идентификация на основе семантической обработки изображений и определение положения объектов на плане местности, что позволило значительно улучшить качество визуализации объектов и повысить за счёт автоматизации эффективность мероприятий, связанных с обеспечением безопасности аэропорта.

7.3. Сравнение эффективности систем технического зрения с различными принципами действия

Кроме систем автоматизированного контроля проникновения на территорию, подобных РАЯЖ 46652.001-ОС.ПЗ, существуют альтернативные способы выделения событий, требующих реакции оператора охранного комплекса приборов. Методика сравнения эффективности систем, построенных на датчиках различного действия, может быть построена следующим образом.

Во-первых, во время сравнительных испытаний все системы должны фиксировать события в одни и те же моменты времени, то есть область действия датчиков сравниваемых систем должна совпадать и в пространстве, и во времени.

Во-вторых, должны быть введены показатели, по которым будет сравниваться система: например, количество ошибок первого рода, и количество ошибок второго рода.

В-третьих, должен быть выработан критерий, по которому из разных показателей будет дана одна оценка, характеризующая качество системы в целом. Далее приведена методика сравнения систем по разным показателям.

Пусть N — количество сравниваемых систем технического зрения, возможно, от разных производителей. Для удобства в дальнейшем будем считать, что каждая сравниваемая система пронумерована, т.е. у каждой есть свой индекс от 1 до N .

Предположим, в процессе испытания мы получили для каждой из систем некие показатели качества её работы x_1, x_2, \dots, x_N . Рангом i -ой системы по показателю x мы будем обозначать величину $rg(i, x)$, равную порядковому номеру показателя системы i в последовательности, отсортированной по возрастанию показателя качества x . Например, если $x_1 = 1, x_2 = 3, x_3 = 1,5$ для трёх испытываемых систем, и увеличение показателя обозначает ухудшение качества, то $rg(1, x) = 3, rg(2, x) = 1, rg(3, x) = 2$. Аналогия ранга — это место на соревнованиях, присуждаемое «по очкам» x , только, в отличие от соревнований, чем больше номер места, тем лучше. Если после упорядочивания показателей несколько значений равны друг другу, то ранг присваивается всем соседям одинаковый, равный среднему значению индексов позиций с одинаковыми значениями показателя. Т.е., если $x_1 = 2, x_2 = 2, x_3 = 2$, то $rg(1, x) = rg(2, x) = rg(3, x) = 2$, а если $x_1 = 1, x_2 = 2, x_3 = 2$, то $rg(1, x) = 1, rg(2, x) = rg(3, x) = 2,5$.

Для каждого элемента функционала, заявленного производителем системы технического зрения, проводятся испытания, которые дают некоторую оценку качества данного функционала. В рамках каждого испытания мы получаем оценку Q_k^i (i — индекс системы, k — пункт программы испытаний) для каждой системы в виде ранга по целому ряду показателей, определённых в процессе испытания конкретного функционала систем. Если система не поддерживает функционал, то её оценка в данном испытании — 0. Оценка переносится в таблицу показателей эффективности систем технического зрения. Общий показатель эффективности для каждой из систем рассчитывается как среднее взвешенное показателей, перечисленных в этой таблице. Веса для среднего взвешенного выбираются таким образом, чтобы важные показатели давали больший вклад в конечную оценку.

В данном параграфе приводится результат сравнительных испытаний для четырёх разных систем, которые автоматически определяют движущиеся объекты, находящиеся в пределах от 50 до 200 метров от заданной точки в угловом секторе около 90° . Местоположение испытательного полигона схематически изображено на рис. 7.3, оно



Рисунок 7.3. Схематичное указание положения полигона испытаний

находится на территориях, прилегающих к предприятию АО «ЭЛВИС Неотек» в г. Зеленоград.

В проведённых автором сравнительных испытаниях участвовали четыре различные системы технического зрения.

Первая система аналогична системе РАЯЖ 46652.001-ОС.ПЗ и состоит из трёх стационарных видеокамер с углом зрения примерно 30° , которые «веером» закрывают показанный на рис. 7.3 сектор.

Вторая система ЛЦКБ.464412.002 состоит из радара ЛЦКБ.685631.001, купольной камеры Apix-36ZDome/D1 EXT и ЭВМ для семантической обработки сигналов. Сектор обзора радара настроен таким образом, чтобы соответствовать указанной ранее области, в которой проводятся сравнительные испытания.

Третья система ВАРШ.00003-01 состоит из тепловизионного локатора кругового обзора «Филин» ВАРШ.201219.003, установленного на поворотной платформе ВСС PTR-500 и ЭВМ для семантической обработки тепловизионных изображений. Угол обзора тепловизионной камеры примерно равен 30° , контроль показанного на рис. 7.3 производится с помощью шагового перехода положения камеры последовательно по трём положениям, которые примерно соответствуют положению видеокамер первой системы. Между переходами для фиксации событий тепловизионная камера остаётся неподвижной примерно на 7 секунд в каждой точке.



Рисунок 7.4. Поля зрения камер: сверху для неподвижных в первой системе, в центре для панорамной камеры (в виде панорамы), снизу для тепловизионной камеры в трёх положениях шагового режима.

Четвёртая система ВАРШ.00001-01 состоит из поворотной купольной камеры Bosch AutoDome 800HD и из сервера семантической обработки изображений, предназначенного для построения и анализа панорамы. Панорамное изображение настраивается таким образом, чтобы в него попадала область, соответствующая сектору, в котором испытываются сравниваемые системы.

Для иллюстрации области действия всех систем на рис. 7.4 показаны изображения, которые анализируются системами без радара. Пользовательский интерфейс радарной системы представлен на рис. 7.5.

Перечисленные четыре системы сравнивались по следующим показателям:

- 1) количество ложно идентифицированных объектов в светлое время суток за период проведения испытаний, соответствует ранговому показателю Q_{\bullet}^1 , где \bullet — индекс системы;
- 2) количество ложно идентифицированных объектов в тёмное время суток за период проведения испытаний, соответствует ранговому показателю Q_{\bullet}^2 ;
- 3) количество пропущенных системой объектов в светлое время суток за период

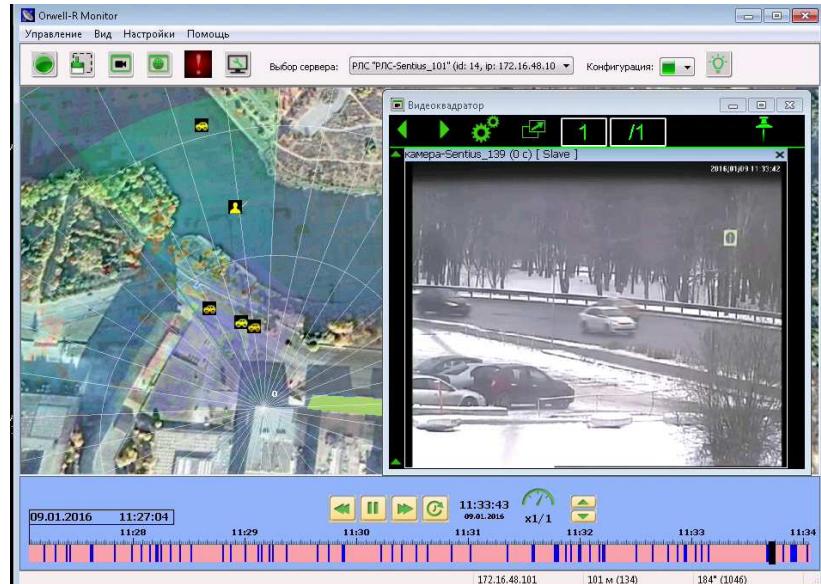


Рисунок 7.5. Внешний вид пользовательского интерфейса радарной системы технического зрения: поворотная камера наведена на один из автомобилей, идентифицированного радаром как движущаяся цель.

Показатель	Видеокамеры	Радар	Тепловизор	Панорама
Дневные ложные срабатывания	0	1	0	3
Ночные ложные срабатывания	1	0	0	0
Дневные пропуски	0	8	11	9
Ночные пропуски	1	1	11	30
Объекты без визуального подтверждения	0	1	16	0

Таблица 7.2. Таблица показателей работы систем технического зрения проведения испытаний, соответствует ранговому показателю Q_{\bullet}^3 ;

4) количество пропущенных системой объектов в тёмное время суток за период проведения испытаний, соответствует ранговому показателю Q_{\bullet}^4 ;

5) количество объектов, которые не попали в поле зрения видеокамер системы, соответствует ранговому показателю Q_{\bullet}^5 .

Период испытаний был выбран таким образом, чтобы в поле зрения всех систем побывало примерно 50 объектов в дневное время, и примерно 50 объектов в ночное время. Соответствующие показатели сведены в таблицу 7.2.

Соответствующие ранговые показатели систем Q_i^k , методика расчёта которых бы-

Показатель ранга	Видеокамеры	Радар	Тепловизор	Панорама
Q_{\bullet}^1	1,5	3	1,5	4
Q_{\bullet}^2	4	2	2	2
Q_{\bullet}^3	1	2	4	3
Q_{\bullet}^4	1,5	1,5	3	4
Q_{\bullet}^5	1,5	3	4	1,5

Таблица 7.3. Ранговая таблица показателей эффективности систем технического зрения

ла приведена ранее в этом параграфе, сведены в таблицу 7.3. Если при формировании итоговой оценки использовать среднее по показателям Q_i^k , а именно:

$$Q_i = \frac{1}{5} \sum_{k=1}^5 Q_i^k,$$

то получим следующие итоговые оценки: для системы на основе трёх видеокамер 1,9, для радара 2,3, для тепловизора и панорамной камеры одинаковое значение 2,9.

Более подробный анализ приведённых выше таблиц показывает, что панорамный детектор объектов хуже всего приспособлен для охраны объектов, поскольку вочных условиях он пропускает более половины событий, происходящих в поле зрения камеры. С другой стороны, плохая работа детектора обусловлена повышенными требованиями к чувствительности камеры: из-за того, что в ночное время сильно повышается экспозиция, изображения объектов становятся смазанными и не отличимыми от фона. Поэтому при соответствующем оборудовании панорамная система будет работать лучше тепловизионного шагового локатора, поскольку данный локатор пропускает очень много объектов из-за того, что они не попадают в область обзора или находятся в ней очень недолго.

Также более подробный анализ пропусков радарной системы показывает, что система из неподвижных видеокамер и радарная система сравнимы друг с другом по достоверности детектирования объектов, если видеосистема используется в приемлемых для неё условиях наблюдения (не далее 300м от видеокамеры). 8 пропусков, указанных в таблице 7.2 для радарной системы, обусловлены либо несовершенством алгоритма генерации нового события в условиях плотного движения, когда несколько объектов формируют только одну радарную отметку на одном проходе луча, либо узким вертикальным углом диаграммы направленности радара, что приводит к пропускам на склоне непосредственно возле радара, поэтому в рамках паспортных ограничений у системы с видеокамерами нет больших преимуществ перед радарными детекторами.

Тем не менее, проведённый эксперимент показывает, что на расстояниях, при ко-

торых атмосферные явления не вызывают помех в поле зрения видеокамер, системы детектирования объектов на основе семантической обработки видеоизображений являются надёжным средством автоматической фиксации событий в поле зрения видеокамер.

7.4. Методика оценки точности сопровождения движущихся объектов

Для оценки точности наведения поворотных видеокамер на движущиеся объекты, прежде всего необходимо развернуть аналитическую систему типа «СПВ ОВТ», описанную в п. 7.2. Минимальный набор оборудования, необходимый для оценки точности наведения — это пара видеокамер, неподвижная и поворотная, и ЭВМ, которая одновременно будет выполнять и программы сервера для обработки видеосигнала, и программы терминала для визуализации результатов анализа и наведения, и необходимую для измерения программу настройки сопряжения — см. рис. 2.1, с. 77.

После установки программного обеспечения для сервера и терминала настраивается сопряжение неподвижной и поворотной видеокамер так, как это описано в п. 2.6. Оценку точности сопряжения можно произвести двумя способами: с помощью программы настройки сопряжения и с помощью сбора статистики наведения на движущиеся объекты (п. 3.6).

Оценка точности наведения при помощи программы настройки сопряжения. Для оценки точности наведения в данном случае требуется провести ряд действий, направленных на измерение, суть которых описана ниже.

1. В программе настройки выбрать сопряжённую пару видеокамер мастер и слейв.
2. Выбрать на изображении неподвижной камеры точку, которая хорошо различима.
3. Посредством программы настройки навести в выбранную точку мастер-камеры слейв-камеру при помощи настроенного преобразования сопряжения. Записать значения координат углов азимута α^n и склонения β^n , отображаемых в программе настройки в таблицу вида 7.5 (см. с. 261). В данном случае номер измерения обозначается как n .
4. С помощью ручного управления программы настройки сопряжения произвести точное наведение в необходимую точку: выбранная точка должна находиться в центре изображения слейв-камеры. Считать угловые координаты α_0^n и β_0^n , соответствующие точному наведению на точку, и записать эти координаты в ту же строчку таблицы 7.5, в которую производилась запись в предыдущем пункте.
5. Вычислить отклонения действительных значений от расчётных,

$$\Delta\alpha^n = \alpha^n - \alpha_0^n, \quad \Delta\beta^n = \beta^n - \beta_0^n,$$

записать эти значения в таблицу 7.5 для измерения n .

6. Провести N различных измерений, где $N \geq 20$.
7. Итоговая погрешность наведения по азимуту Δ_α равна

$$\Delta_\alpha = 3\sigma_\alpha = 3\sqrt{\frac{1}{N-1} \sum_{n=1}^N (\Delta\alpha^n)^2},$$

погрешность по склонению Δ_β равна

$$\Delta_\beta = 3\sigma_\beta = \sqrt{\frac{1}{N-1} \sum_{n=1}^N (\Delta\beta^n)^2}.$$

Общую норму отклонения обоих углов при наведении можно оценить как $\|\Delta\vec{\tau}\| = 3\sqrt{(\sigma_\alpha^2 + \sigma_\beta^2)/2}$, где $\vec{\tau} = (\alpha, \beta)$ — вектор углов наведения. Конкретные числовые значения для данных погрешностей вычислены в п. 7.5.

Описанная методика позволяет оценить точность наведения на неподвижные объекты. Для оценки точности наведения на подвижные объекты целесообразно использовать статистический подход.

Статистическая оценка погрешности наведения на движущиеся объекты. Для того, чтобы оценить погрешность наведения статистическим способом так, как это описано в п. 3.6, необходимо выполнить следующие шаги.

1. После настройки сопряжения пары мастер и слейв, запустить сервер системы. Необходимо при этом установить такие параметры сервера, чтобы слейв-камера наводилась с одним и тем же углом зрения на все объекты, иначе формула (3.31) неприменима для оценки погрешности.
2. Оставить сервер в режиме автоматического наведения на любые движущиеся объекты, которые выделяют алгоритмы обработки видеопотока от мастер-камеры. Необходимо, чтобы слейв-камера совершила S попыток наведения на движущиеся объекты, причём для адекватного сбора статистики необходимо $S \geq 10^3$.
3. Запустить терминал системы и войти в режим вывода списков объектов, которые система выделила на мастер-камере, и у которых есть снимки слейв-камеры.
4. Среди S изображений объектов от поворотной камеры, полученных в результате наведения, выделить S_α изображений, в которые объект не попал из-за слишком большой погрешности наведения по азимуту, и S_β изображений, в которые объект не попал из-за слишком большой погрешности наведения по склонению.
5. Оценить горизонтальный угол зрения α слейв камеры, при котором она наводилась на объекты. Поскольку координата увеличения слейв-камеры в данном эксперименте зафиксирована, и известна, то для получения угла зрения необходимо либо

воспользоваться техническим паспортом камеры, в котором указана зависимость угла зрения от координаты увеличения объектива камеры, либо необходимо оценить эту зависимость с помощью лабораторных измерений.

6. Вычислить вертикальный угол зрения слейв-камеры β , который удовлетворяет соотношению

$$\beta = 2 \operatorname{arctg} \left(\frac{3}{4} \operatorname{tg} \frac{\alpha}{2} \right)$$

в силу геометрических свойств ПЗС-матриц современных видеокамер. При $\alpha < 30^\circ$, т.е. в приближении малых углов, $\beta \approx \frac{3}{4}\alpha$.

7. Пользуясь формулой (3.31), вычислить оценку погрешности по азимуту и склонению как, соответственно,

$$\Delta_\alpha = 3 \cdot \frac{\alpha/2}{F^{-1}(1 - S_\alpha/2S)} \text{ и } \Delta_\beta = 3 \cdot \frac{\beta/2}{F^{-1}(1 - S_\beta/2S)},$$

где F^{-1} функция ошибок, обратная к функции стандартного нормального одномерного распределения.

Данная методика, в отличие от предыдущей, позволяет оценить погрешность наведения, обусловленную всеми факторами сразу: погрешность сопряжения, задержки синхронизации данных о положении объектов, неточное выделение объектов алгоритмами выделения движения и другие.

7.5. Экспериментальная оценка точности управления поворотными видеокамерами

Программно-аппаратное обеспечение экспериментальных измерений. Оборудование, которое в данной работе задействовано в измерении точности наведения поворотных камер, состоит из ЭВМ, поворотной и неподвижной видеокамеры, устройств управления оператора (клавиатура и «мышь») и монитора в качестве устройства вывода.

В качестве ЭВМ используется компьютер универсального назначения с высокими показателями производительности: центральный процессор Intel Core-i7 с тактовой частотой 3ГГц, системная плата с эффективной частотой шины данных 1,6ГГц. ЭВМ, таким образом, обладает эталонной производительностью (см. определение 2.5).

В эксперименте используются неподвижная и поворотная видеокамеры, предназначенные для внешнего наблюдения. Неподвижная видеокамера представляет из себя прибор, включающий в себя объектив от производителя «COMPUTAR» с фокусным расстоянием 6-12мм и автоматическим управлением диафрагмой, прибор-камера с ПЗС-матрицей 1/2" производителя «WATEC» с паспортной чувствительностью 0.01Лк и гермокожух, предохраняющий от влаги и перепадов внешних температур

камеру и объектив, а также преобразователь питания $\sim 220\text{В}/-9\text{В}$ для камеры и блока управления диафрагмой объектива. В качестве поворотной видеокамеры используется купольная видеокамера «WV-CS860» производителя «Panasonic» с управляемым фокусным расстоянием, которое может изменяться от 8мм до 80мм.

В эксперименте неподвижная и поворотная камера располагаются на небольшом, порядка 50см, расстоянии друг от друга, см. фотографию на рис. 7.6. На этом же рисунке представлен мастер. Передача видеосигнала от видеокамер к ЭВМ осуществляется в формате «PAL» по коаксиальным проводам, и поступает в ЭВМ с помощью специальных аналого-цифровых плат-преобразователей.

Резюмируя описание приборов, задействованных в эксперименте, приведём таблицу 7.4, описывающую основные технические характеристики данных приборов.

Тактовая частота ЦП ЭВМ	3ГГц
Частота обмена с памятью	400МГц
Фокусное расстояние объектива мастера	12мм
Тип ПЗС-матрицы камеры	1/2"(1/2 дюйма)
Горизонтальный угол обзора мастер-камеры	30°
Дистанция до дальнего края зоны обзора мастера	100м
Высота подвеса мастера и слейва	5м
Максимальная скорость вращения слейва	300°с ⁻¹
Угол зрения слейва при эксперименте	5°
Фокусное расстояние объектива слейва	55мм
Увеличение слейва относительно мастера	6×

Таблица 7.4. Технические характеристики приборов измерения точности управления поворотной камерой.

Для подготовки экспериментальных измерений необходимо настроить сопряжение неподвижной и поворотной видеокамеры (см. п. 2.6). В ходе измерений используется как приложение настройки сопряжения, так и терминальное приложение. Для того, чтобы выполнить шаги измерений, необходимо ознакомиться с руководством эксплуатации данного программного обеспечения.

Результаты измерения точности наведения на неподвижный объект. Выберем набор точек на изображении видеокамеры, в которых будет производится оценка точности наведения. Точки измерения обозначены на рис. 7.7.

Рассмотрим, например, измерение точности наведения для точки 1. Выберем увеличение относительно мастера 1,5–2 и выполним наведение на основе преобразования сопряжения, нажимая правую кнопку «мыши» по изображению мастер-камеры и удерживая при этом клавишу клавиатуры «CTRL». В результате в программе со-

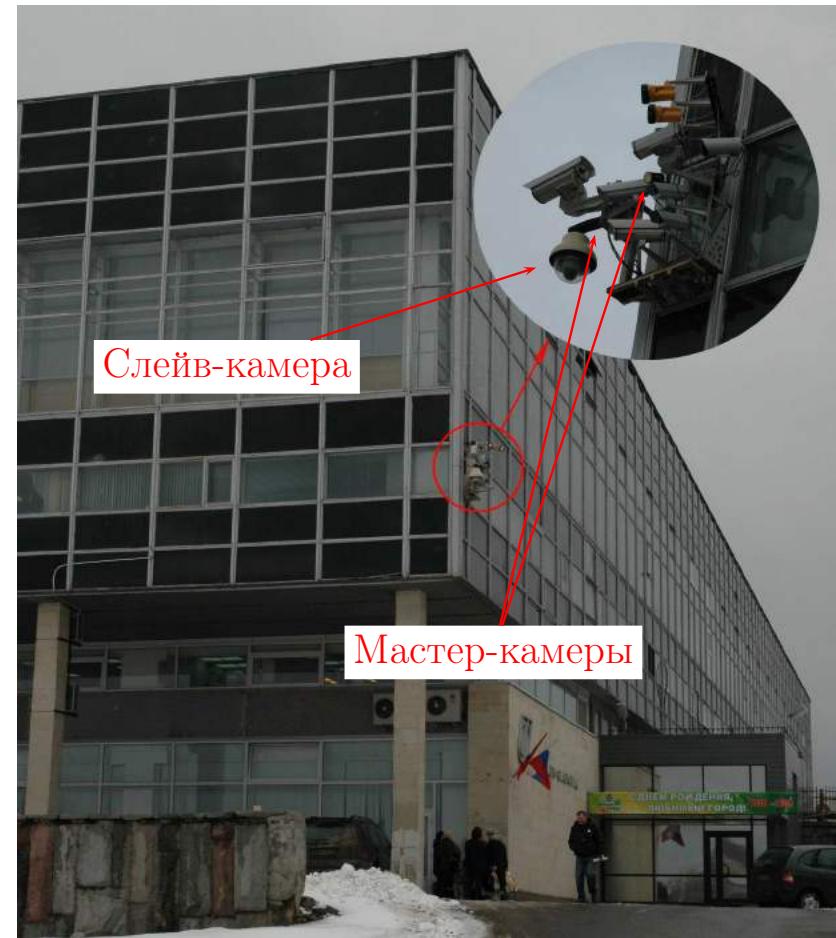


Рисунок 7.6. Справа — внешний вид установки оптических приборов для измерения точности управления поворотными видеокамерами, слева — изображение неподвижной камеры.

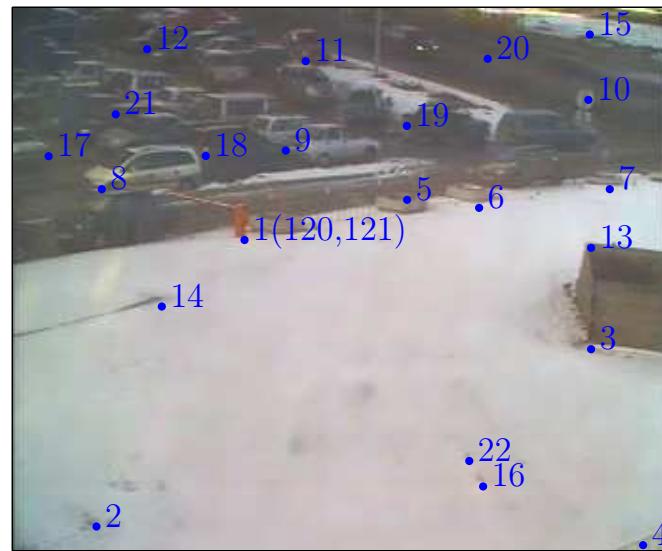


Рисунок 7.7. Точки измерения погрешности наведения. Номер точки соответствует номеру измерения. Для первой точки отображены координаты (120,121) на изображении мастера.



Рисунок 7.8. Смещение изображений слейва и мастера из-за погрешности наведения (изображение шлагбаума в центре левого изображения раздвоено).

пряжения увидим небольшое смещение изображений мастера и слейва, см. рис. 7.8. С помощью команды ЧТЕНИЕ_КООРДИНАТ(), отправленной из приложения настройки сопряжения по кнопке «Установка» интерфейса управления программой, мы можем считать координаты α и β , и записать их в первую строчку таблицы 7.5. Как видно из рисунка 7.8, цифровые угловые координаты азимута и склонения равны, соответственно, 2767 и 1695. Для купольной поворотной камеры, используемой в эксперименте, эти координаты соответствуют углам $\alpha = 276,7^\circ = 2767/10$ и $\beta = 84,75^\circ = 1695/20$.

С помощью кнопки «» интерфейса программы настройки или соответствующих клавиш клавиатуры необходимо вручную установить наиболее точное возможное сов-



Рисунок 7.9. Совмещённые вручную изображения слайва и мастера (в центре левого изображения совмещены клумба и шлагбаум).

мешение изображений мастера и слайва, как показано на рис. 7.9. Затем необходимо повторно считать угловые координаты, которые соответствуют координатам точного наведения α_0 и β_0 , и занести их в таблицу 7.5. Цифровым координатам 2764 и 1696 углов азимута и склонения, изображённых на рис. 7.9, соответствуют углы $\alpha_0 = 276,4^\circ$ и $\beta_0 = 84,8^\circ$.

После проведения 22 серий испытаний, заполнения таблицы 7.5 и выполнения расчёта по методике, описанной в предыдущем параграфе, получим следующий результат: погрешность наведения по азимуту равна

$$\Delta_\alpha = 3 \cdot \sqrt{\frac{1}{21} \sum_{n=1}^{22} (\Delta\alpha^n)^2} = 0,45^\circ,$$

погрешность наведения по склонению равна

$$\Delta_\beta = 3 \cdot \sqrt{\frac{1}{21} \sum_{n=1}^{22} (\Delta\beta^n)^2} = 0,55^\circ,$$

совокупная погрешность наведения равна

$$\|\Delta\vec{\tau}\| = \sqrt{\Delta_\alpha^2 + \Delta_\beta^2} = \sqrt{(0,45^2 + 0,55^2)/2} = 0,5^\circ.$$

Данный экспериментальный результат совпадает с теоретическим значением, полученным в п. 3.4.

Определение погрешности наведения на движущиеся объекты на основе статистических экспериментальных данных. Для сбора статистики программное обеспечение сервера запускается в специальном режиме, при котором горизонтальный угол зрения слайва постоянен, и соответствует координате увеличения 180

n	α	β	α_0	β_0	$\Delta\alpha$	$\Delta\beta$
1	276,8	84,8	276,9	84,75	-0,1	0,05
2	269,6	72,15	269,6	72,25	0,0	-0,1
3	293,0	80,1	293,0	80,2	0,0	-0,1
4	295,7	71,4	295,7	71,45	0,0	-0,05
5	284,5	86,75	284,2	86,55	0,3	0,2
6	287,8	86,4	287,5	86,35	0,3	0,05
7	293,6	87,4	293,6	87,15	0,0	0,25
8	270,2	87,25	270,4	87,25	-0,2	0,0
9	278,8	89,0	278,9	88,95	-0,1	0,05
10	292,6	91,4	292,5	91,15	0,1	0,25
11	279,7	93,15	279,6	92,95	0,1	0,2
12	272,5	93,65	272,6	93,55	-0,1	0,1
13	292,9	84,6	292,8	84,7	0,1	-0,1
14	272,9	81,8	273,1	81,9	-0,2	-0,1
15	292,6	94,25	292,4	94,0	0,2	0,25
16	288,0	73,6	288,1	74,05	-0,1	-0,45
17	267,8	88,85	268,0	88,75	-0,2	0,1
18	275,1	88,7	275,2	88,6	-0,1	0,1
19	284,4	90,2	284,3	89,85	0,1	0,35
20	288,0	93,25	287,9	92,95	0,1	0,3
21	270,9	90,7	271,1	90,55	-0,2	0,15
22	287,4	74,85	287,4	75,15	0,0	-0,3

Таблица 7.5. Значения измеренных углов при оценке точности наведения. n — порядковый номер измерения, α и β — экспериментальное значение азимута и склонения, α_0 и β_0 — точное значение азимута и склонения, $\Delta\alpha$ и $\Delta\beta$ — отклонение по азимуту и по склонению. Значения углов приведены в градусах.

для камеры «Panasonic WV-CS860». Кроме того, для оценки эффективности алгоритма наведения, введённого в п. 2.6, мы проведём три серии испытаний.

В первой серии сервер управляет поворотной камерой без учёта упреждения, т.е. поворотная камера наводится в угловые координаты α_1 , β_1 , в которых объект находится в начале движения поворотной камеры. Назовём такой способ наведения *алгоритмом наведения без упреждения*.

Вторая серия испытаний проводится с упреждением на фиксированный отрезок времени, без решения уравнения упреждения (3.13). Таким образом, наведение производится в угловые координаты $\alpha_1 + \omega_\alpha t$, $\beta_1 + \omega_\beta t$, где ω_α , ω_β — угловые скорости изменения координат азимута и склонения объекта, которые оценивает алгоритм вы-

деления движущихся объектов из потока видеосигнала неподвижной камеры, а t — фиксированное время наведения. Наиболее логичный выбор времени t — это среднее время перехода видеокамеры из одной позиции в другую. Для поворотной видеокамеры «Panasonic WV-CS860» в данном эксперименте время t равно 1,4с. Назовём такой способ наведения *алгоритмом наведения с линейным упреждением*.

Третья серия измерений проводится на основе результатов, полученных в п. 3.3, т.е. наведение производится с упреждением в координаты $\alpha_1 + \omega_\alpha t$, $\beta_1 + \omega_\beta t$, как и во второй серии испытаний, но время t вычисляется на основе уравнения (3.13). Поскольку время перехода камеры «Panasonic WV-CS860» из координат α_0 , β_0 в координаты α , β определяется функцией $T(\vec{\tau}, \vec{\tau}_0) = T_\alpha|\alpha - \alpha_0| + T_\beta|\beta - \beta_0| + C$, где $T_\alpha = 0,008\text{с}$, $T_\beta = 0,008\text{с}$, $C = 0,8\text{с}$, то значение t определяется соотношениями вида (3.18), а именно:

$$\begin{aligned} &\text{если } \underbrace{\alpha > \alpha_0, \beta > \beta_0}_{}, \text{ то } t = \frac{T_\alpha(\alpha_1 - \alpha_0) + T_\beta(\beta_1 - \beta_0) + C}{1 - \omega_\alpha T_\alpha - \omega_\beta T_\beta}, \\ &\text{если } \underbrace{\alpha \leq \alpha_0, \beta > \beta_0}_{}, \text{ то } t = \frac{T_\alpha(\alpha_0 - \alpha_1) + T_\beta(\beta_1 - \beta_0) + C}{1 + \omega_\alpha T_\alpha - \omega_\beta T_\beta}, \\ &\text{если } \underbrace{\alpha \leq \alpha_0, \beta \leq \beta_0}_{}, \text{ то } t = \frac{T_\alpha(\alpha_0 - \alpha_1) + T_\beta(\beta_0 - \beta_1) + C}{1 + \omega_\alpha T_\alpha + \omega_\beta T_\beta}, \\ &\text{если } \underbrace{\alpha > \alpha_0, \beta \leq \beta_0}_{}, \text{ то } t = \frac{T_\alpha(\alpha_1 - \alpha_0) + T_\beta(\beta_0 - \beta_1) + C}{1 - \omega_\alpha T_\alpha + \omega_\beta T_\beta}. \end{aligned}$$

Такой способ наведения будем называть *алгоритмом наведения с адаптивным упреждением*.

Для набора 1000 изображений слайва при наведении на объекты в каждой серии измерений требовалось порядка 1 часа в тех условиях наблюдения, при которых проводился эксперимент. Полученные изображения извлекаются из базы данных системы с помощью терминального приложения, и затем вручную ведётся подсчёт количества «промахов» мимо объекта. Для этого изображения со слайва и соответствующие им синхронные изображения с мастера сводятся в таблицу вида 7.6.

С помощью приведённой таблицы можно посчитать для каждой из трёх серий общее количество снимков, а также количество «промахов» вверх, вниз, влево и вправо. Общее количество изображений во всех трёх сериях для простоты расчётов равно 1000. Количество промахов отображено на гистограммах на рис. 7.10. На этих гистограммах считается известным угол зрения слайва. Координата увеличения 180 соответствует углу зрения 5° и шестикратному ($30^\circ/5^\circ = 6$) увеличению изображения слайва относительно изображения неподвижной камеры. Получить значение угла зрения слайва можно, непосредственно рассмотрев зону наблюдения слайва на мастере: размеры этой зоны по отношению к размерам изображения мастера задаёт соотношение углов, а угол зрения мастер-камеры известен из условий измерения (см.

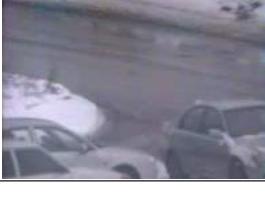
Изображение слайва	Изображение мастера	Комментарий
		изображение объекта в кадре
		изображение объекта в кадре
		промах: вверх
		промах: вправо
		изображение объекта в кадре
		изображение объекта в кадре
		изображение объекта в кадре
		промах: вправо

Таблица 7.6. Таблица изображений слайва при наведении на движущиеся объекты.
 «⊕» — объект наведения, [—] — зона слайва.

таблицу 7.4).

С помощью формулы

$$3\sigma = \frac{3\Delta}{F^{-1}(1 - p/2)},$$

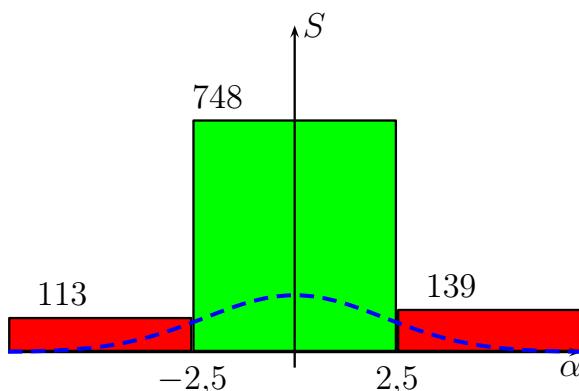
приведённой в методике измерения погрешности (см. п. 7.4, Δ — половина угла зрения слейва, горизонтального для азимута и вертикального для склонения, p — доля промахов, и F^{-1} — функция, обратная к функции стандартного нормального распределения) на основе собранной статистики получим результаты, которые сведены в таблицу 7.7, см. ниже.

Наведение	доля промахов по азимуту	погрешность по азимуту	доля промахов по склонению	погрешность по склонению
без упреждения	0,252	7°	0,096	3°
линейное	0,047	4°	0,053	3°
адаптивное	0,008	3°	0,002	2°

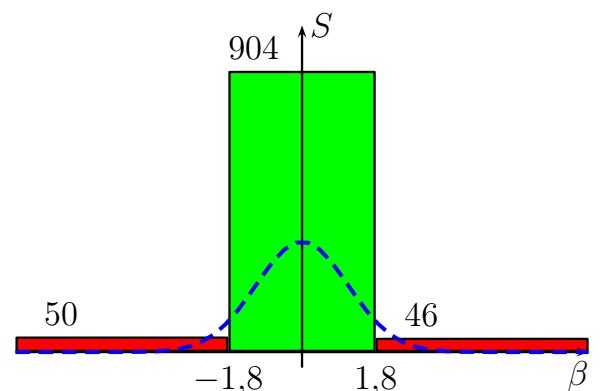
Таблица 7.7. Результаты измерения погрешности наведения. 1-я серия измерений производилась без упреждения, 2-я серия измерений производилась с фиксированным временем упреждения (алгоритм линейного упреждения), 3-я серия измерений производилась с наведением по алгоритму, описанному в п. 3.3 (адаптивное упреждение).

Из анализа таблицы результатов 7.7 следует три вывода вывода. Во-первых, экспериментальная оценка погрешности 3° в несколько раз меньше теоретической оценки погрешности 20°, полученной в п. 3.4, что обусловлено тем, что при её получении предполагались наихудшие условия наведения, которые редко встречаются на практике. Во-вторых, применение алгоритмов наведения, разработанных в диссертационной работе, позволяет уменьшить погрешность наведения в два раза по сравнению с погрешностью, которая получается при использовании самых простых алгоритмов управления поворотной видеокамерой. При этом доля промахов уменьшается на порядок в условиях проведения измерений: для первой серии при наведении без упреждения она примерно равна $3,0 \cdot 10^{-1}$, а в третьей серии при наведении по алгоритму, разработанному в данной диссертации, доля промахов $1,0 \cdot 10^{-2}$. В-третьих, погрешность наведения 3° при угле наблюдения сопряжённого мастера 30° позволяет получать изображения объекта на слейве с разрешением в 10 раз выше, чем изображение объекта на неподвижной камеры при высокой достоверности наведения на объект.

A) статистика наведения без упреждения.

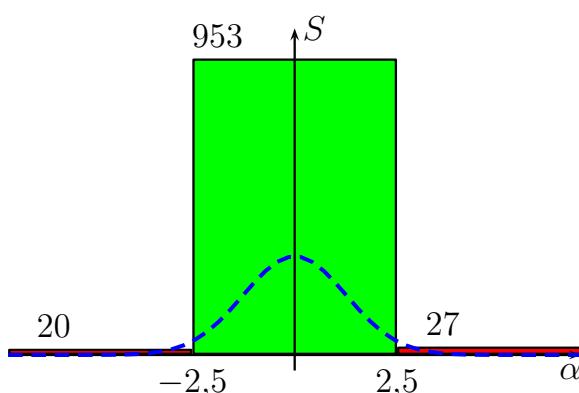


«Промахи» по азимуту,
252 из 1000

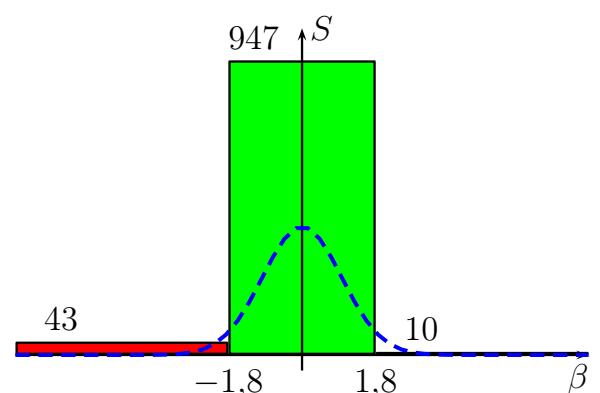


«Промахи» по склонению,
96 из 1000

B) статистика наведения с линейным упреждением.

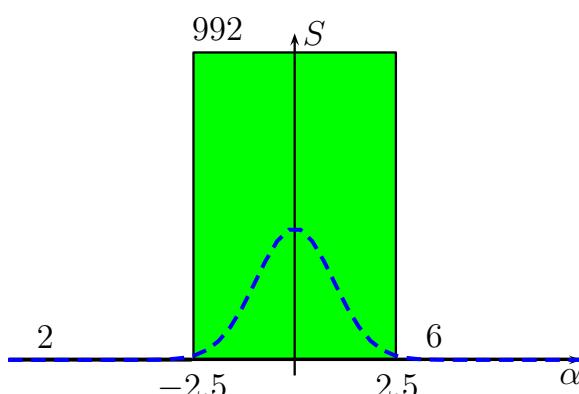


«Промахи» по азимуту,
47 из 1000

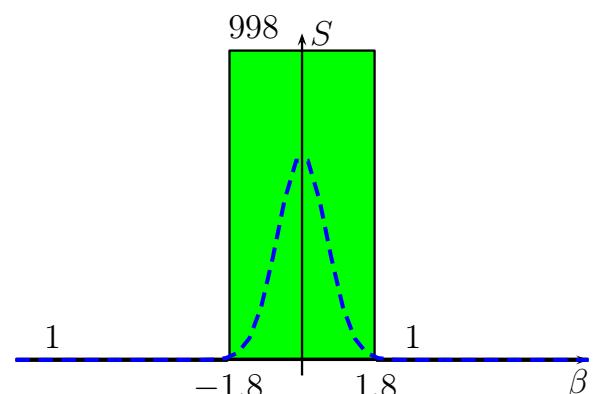


«Промахи» по склонению,
53 из 1000

B) статистика наведения с адаптивным упреждением.



«Промахи» по азимуту,
8 из 1000



«Промахи» по склонению,
2 из 1000

Рисунок 7.10. Статистика «промахов» при наведении на объекты (светлые столбики — «попадания», тёмные столбики — «промахи», пунктирная линия — график плотности распределения).

7.6. Методика оценки достоверности сопоставления изображений объектов на сопряжённых видеокамерах

Методика измерения точности сопряжения двух неподвижных видеокамер во многом совпадает с методикой оценки точности сопряжения поворотных и неподвижных видеокамер, описанной в п. 7.4. Для измерения погрешности сопряжения неподвижных камер также, как и для оценки погрешности наведения, необходимо развернуть систему, аналогичную системе «СПВ ОВТ», см. п. 7.2, которая должна включать в себя минимум две неподвижные видеокамеры с пересекающимися зонами наблюдения, которые будут сопряжены. Для оценки времени синхронизации данных с помощью сетевых взаимодействий вместо одной ЭВМ требуется минимум две ЭВМ для запуска программного обеспечения серверов на каждой из них. Для оценки влияния сетевой задержки синхронизации данных ЭВМ системы должны быть объединены ЛВС. Помимо программного обеспечения серверов на одной из ЭВМ должны выполняться программа терминала и программа настройки сопряжения неподвижных видеокамер, см. рис. 2.7, с. 91. Данные программы используются при измерениях как средство получения экспериментальных данных и статистики. Таким образом, по крайней мере одна из ЭВМ должна быть рабочим местом оператора, то есть быть подключена к монитору, клавиатуре и манипулятору «мышь».

После установки программного обеспечения для серверов и терминала настраивается сопряжение между неподвижными видеокамерами так, как это описано в п. 2.3 и в документации программного обеспечения. Оценку точности сопряжения можно произвести двумя способами: с помощью программы настройки сопряжения и с помощью сбора статистики сопоставления изображений объектов в общей зоне наблюдения сопряжённых мастеров (п. 3.6).

Оценка погрешности сопряжения при помощи программы настройки. Перечислим шаги измерения погрешности сопряжения неподвижных камер по порядку.

1. В программе настройки необходимо выбрать пару сопряжённых неподвижных камер и вывести окна с их изображением при включённом режиме отображения преобразования координат с одной камеры на другую.
2. Выбрать камеру, с которой производится преобразование. Для определённости будем называть эту камеру первой. Затем выбрать в зоне общего обзора камер на изображении первой камеры точку, которая хорошо различима.
3. Навести курсор «мыши» на выбранную точку первой камеры.
4. Визуально определить координаты точки преобразования (x^n, y^n) , которые отображаются в строке состояния программы настройки сопряжения. Записать полу-

ченные значения для измерения с порядковым номером n в таблицу вида 7.9.

5. Определить действительное положение точки преобразования (x_0^n, y_0^n) посредством наведения курсора в программе настройки сопряжения, в которой координаты курсора на изображении камеры также отображаются в строке состояния.

6. Вычислить отклонения экспериментально-расчётных значений от действительных,

$$\Delta x^n = x^n - x_0^n, \quad \Delta y = y^n - y_0^n,$$

записать эти значения в таблицу 7.9.

7. Провести N измерений, где $N \geq 20$.

8. Вычислить оценку дисперсии евклидовой нормы векторов $(\Delta x^n, \Delta y^n)$ как

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{n=1}^N ((\Delta x^n)^2 + (\Delta y^n)^2)}.$$

Итоговая погрешность сопряжения, обозначенная как Δr , равна 3σ :

$$\Delta r = 3 \cdot \sqrt{\frac{1}{N-1} \sum_{n=1}^N ((\Delta x^n)^2 + (\Delta y^n)^2)},$$

конкретное значение погрешности приведено далее в п. 7.7.

Данная методика оценивает статическую погрешность сопряжения изображений. Для оценки сопряжения движущихся изображений используется статистический подход.

Статистическая оценка погрешности сопряжения при сопоставлении подвижных изображений мастер-камер. Чтобы оценить погрешность сопряжения с помощью статистики по результатам работы алгоритма сопоставления объектов так, как это описано в п. 3.6, необходимо выполнить нижеперечисленные действия.

1. После настройки сопряжения двух мастеров с пересекающимися зонами обзора следует запустить серверы системы, выполняющие анализ и сопоставление изображений.

2. Для упрощённого поиска объектов в зоне пересечения областей зрения сопряжённых мастеров в терминальном модуле следует создать так называемые тревожные датчики, которые срабатывают при условии, что объект попал в выделенную зону.

3. Программное обеспечение серверов должно работать в режиме автоматического выделения движущихся объектов. На сопряжённых камерах автоматически производится сопоставление движущихся объектов в общей зоне обзора. Для набора статистики нужно, чтобы было сопоставлено между собой S пар изображений одних и

тех же объектов в зоне пересечения областей наблюдения сопряжённых мастеров. Необходимое число сопоставлений $S \geq 1000$.

4. Запустить терминальный модуль и войти в режим вывода списков объектов, для которых сработали тревожные датчики, введённые в пункте 2. Поскольку данные объекты находятся в зоне пересечения областей обзора сопряжённых мастеров, то для них обязательно хотя бы раз должна была производиться процедура сопоставления изображений.

5. Среди S пар изображений объектов выделить S_x пар, которые ошибочно не были сопоставлены между собой.

6. Рассчитать количество пикселей Δ , которые соответствуют горизонтальному «промаху» при ошибочном сопоставлении объектов.

7. Вычислить погрешность Δx по формуле

$$\Delta x = 3 \frac{\Delta}{F^{-1}(1 - S_x/2S)},$$

которая следует из соотношения (3.32)) для погрешности сопряжения подвижных объектов в общей зоне обзора.

Данная методика, в отличие от предыдущей, позволяет учесть вклад задержек синхронизации в погрешность сопряжения изображений объектов сопряжённых мастеров.

7.7. Экспериментальные результаты сопоставления синхронных изображений объектов

Описание аппаратных и программных средств эксперимента. Оборудование, задействованное в измерении погрешности сопряжения неподвижных видеокамер, в целом такое же по составу, как в экспериментах измерения погрешности наведения поворотных камер. Состав приборов включает в себя одну или две ЭВМ, две неподвижные видеокамеры, устройства ввода и вывода для взаимодействия с оператором при измерениях и приборы, соединяющие ЭВМ в ЛВС.

В качестве ЭВМ используется такой же компьютер универсального назначения, как и в п. 7.5. Данный компьютер обладает эталонной производительностью (см. определение 2.5).

Обе неподвижные видеокамеры предназначены для наружного видеонаблюдения. Неподвижная видеокамера, как и в п. 7.5 — это система приборов, состоящая из, во-первых, объектива от производителя «COMPUTAR» с фокусным расстоянием 6-12мм и автоматическим управлением диафрагмой, во-вторых, камеры с ПЗС-матрицей 1/2" производителя «WATEC» (чувствительность 0.01Лк) и, в-третьих, гермокожу-

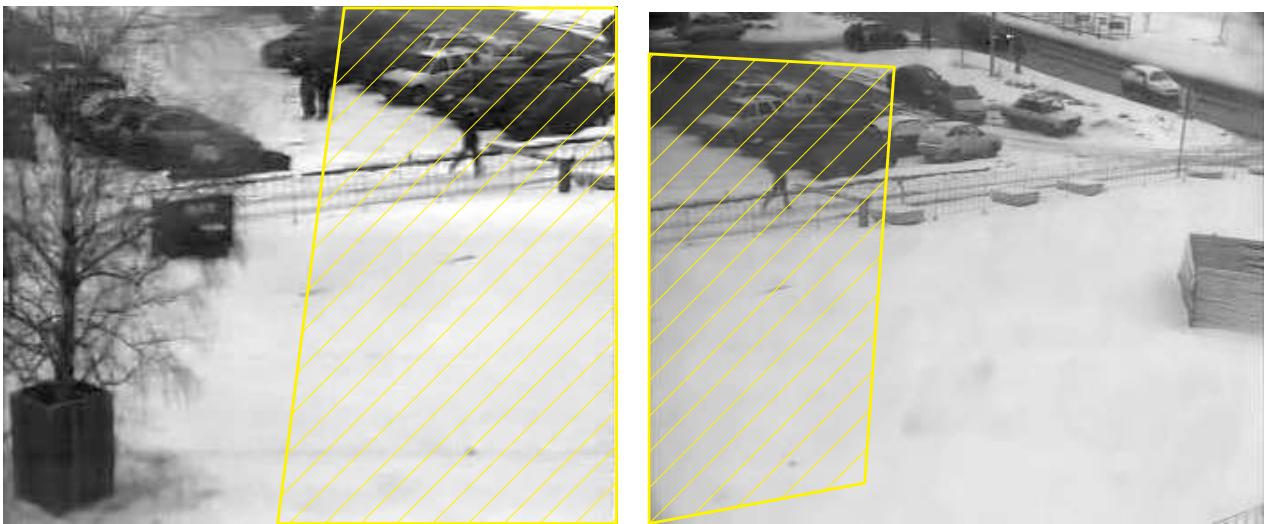


Рисунок 7.11. Изображение сопряжённых неподвижных видеокамер с общей зоной обзора. Многоугольниками обозначена зона пересечения.

ха, предохраняющего от влаги и перепадов внешних температур камеру, объектив, и преобразователь питания $\sim 220\text{В}/-9\text{В}$.

В эксперименте неподвижные камеры располагаются на небольшом расстоянии друг от друга, см. фотографию на рис. 7.6, с. 258. Изображение сопряжённых видеокамер представлены на рис. 7.11. Передача видеосигнала к ЭВМ осуществляется в формате «PAL» по коаксиальным проводам, и поступает в ЭВМ с помощью специальных аналого-цифровых плат-преобразователей.

Если в системе задействовано два сервера, то их необходимо соединить ЛВС. В качестве приборов, обеспечивающих данное соединение, в данном случае достаточно подключить к ЭВМ два сетевых адаптера типа «Ethernet» с пропускной способностью 100Мбит/с, широко распространённых в настоящее время, и соединить их кабелем витой пары с разведением контактов «крест на крест». Если ЭВМ в сети более двух, то необходима более сложная приборная конфигурация сети.

В таблице 7.8 приведены основные технические характеристики приборов эксперимента.

В ходе подготовки экспериментальных измерений настраивается сопряжение выбранной пары неподвижных видеокамер с пересечением зон обзора (см. п. 2.3). При измерениях используется как приложение настройки сопряжения, так и терминальное приложение.

Измерение погрешности сопряжения изображений объектов неподвижных камер без учёта движения. На рис. 7.12 обозначены точки, в которых измеряется погрешность сопряжения. Назовём условно камеры левой и правой, согласно рис. 7.12. Поскольку преобразование сопряжения определено как с левой камеры на правую, так и с правой камеры на левую, для измерения 22 отклонений достаточно выбрать 11 пар точек: для каждой пары возможно определить отклонение преобра-

Тактовая частота ЦП ЭВМ	3ГГц
Частота обмена с памятью	400МГц
Фокусное расстояние объектива мастера	12мм
Тип ПЗС-матрицы камеры	1/2"(1/2 дюйма)
Горизонтальный угол обзора мастер-камеры	30°
Дистанция до дальнего края зоны обзора мастера	100м
Высота подвеса мастеров	5м
Пропускная способность ЛВС, соединяющей ЭВМ	100Мбит/с

Таблица 7.8. Технические характеристики приборов измерения погрешности сопряжения мастер-камер.

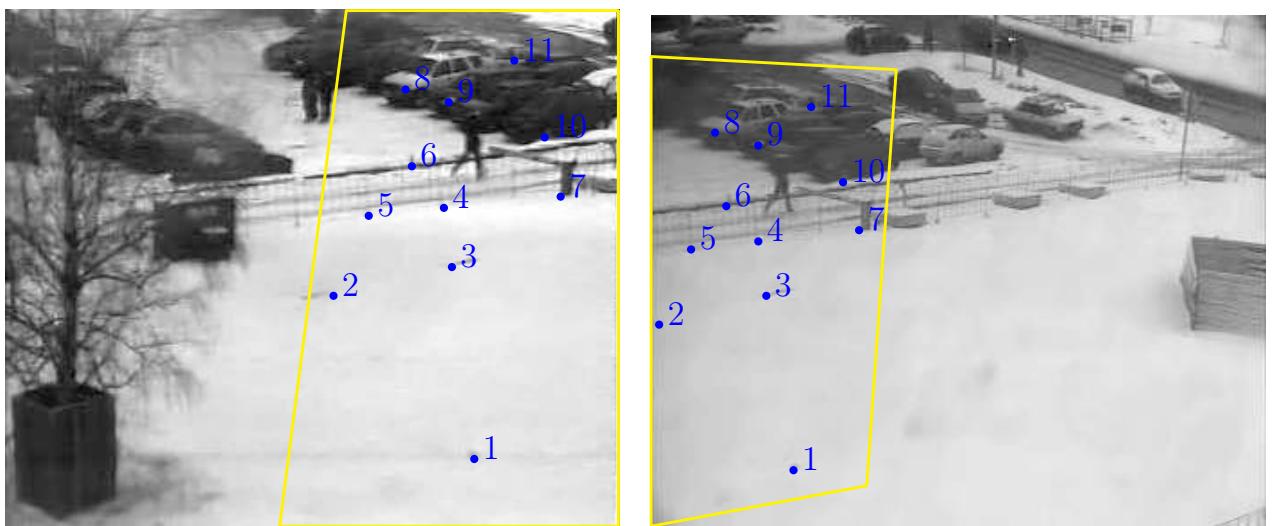


Рисунок 7.12. Точки измерения погрешности сопряжения неподвижных камер. Справа — изображения исходного мастера, слева — изображение, на которое производится преобразование.

зования с левой камеры на правую, и с правой на левую.

Пусть измерения пронумерованы таким образом, что точкам с номером 1 на левой и правой камерах соответствуют измерения 1 (преобразование с изображения левой камеры на изображение правой) и 2 (преобразование с изображения правой камеры на изображение левой), точкам с номером 2 — измерения 3 и 4, точкам с номером 3 — измерения 5 и 6, и далее аналогично. Рассмотрим, например, измерение номер 13 (точки с номером 7 на левой и правой камерах, преобразование слева направо).

Для оценки отклонения преобразования от точного значения необходимо в программе сопряжения подвести курсор «мыши» в точку 7 изображения левой камеры, при этом курсор будет формы «+», см. рис. 7.13. Программа настройки сопряжения неподвижных видеокамер автоматически покажет на изображении правой камеры точку преобразования, отметив её курсором вида «×», отобразив при этом координаты (116, 133) данной точки. Действительные значения координат точки преобра-

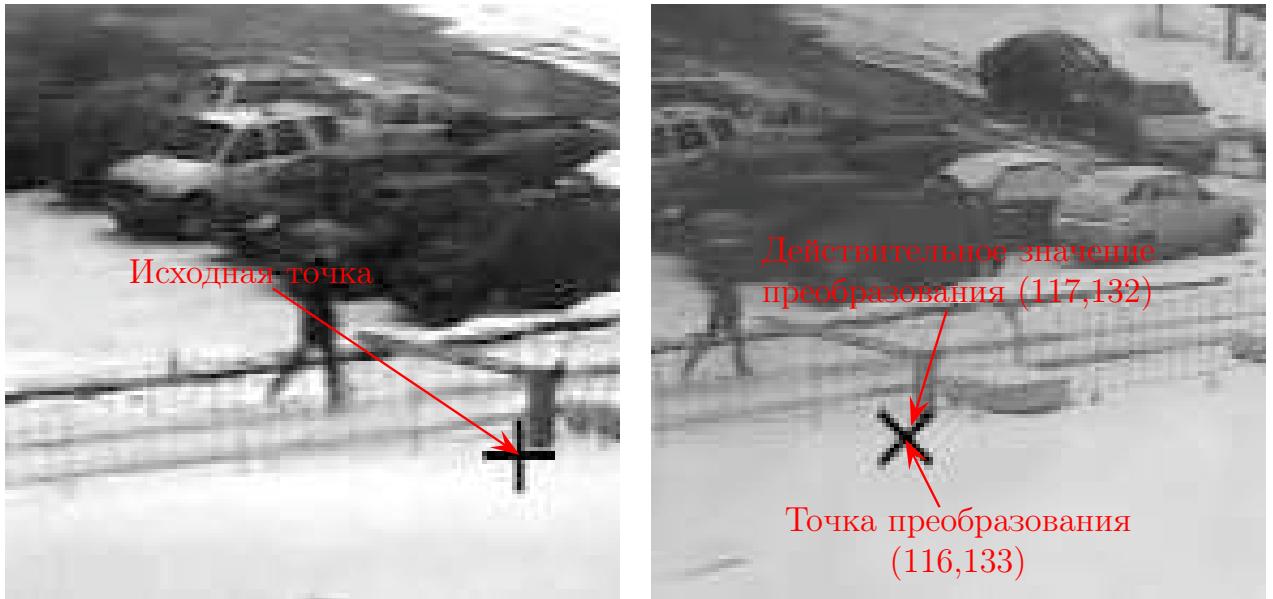


Рисунок 7.13. Измерение погрешности сопряжения в одной из точек (точка 7, рис. 7.12).

зования $(117, 132)$ можно узнать, подведя курсор «мыши» к данной точке уже на изображении правой камеры. Для того, чтобы настолько небольшие смещения были различимы, следует использовать возможность увеличения масштаба изображения, предусмотренную в программе настройки сопряжения.

Значения $(x^{13}, y^{13}) = (116, 133)$, $(x_0^{13}, y_0^{13}) = (117, 132)$, а также отклонения $(\Delta x^{13}, \Delta y^{13}) = (-1, 1)$ заносятся в таблицу 7.9. В эту же таблицу заносятся результаты остальных серий измерений.

В конечном итоге получим погрешность сопряжения Δr , равную

$$\Delta r = 3 \cdot \sqrt{\frac{1}{21} \sum_{n=1}^{22} ((\Delta x^n)^2 + (\Delta y^n)^2)} \approx 4 \text{ пикселя},$$

согласно формуле, указанной в методике, см. п. 7.6.

Определение погрешности сопряжения изображений движущихся объектов на основе статистических измерений. Проведём две серии испытаний для оценки влияния времени синхронизации данных на погрешность сопряжения движущихся изображений в зоне пересечения областей наблюдения двух неподвижных камер.

В первой серии обработчики сопряжённых мастеров запускаются на одной ЭВМ, и, соответственно, на время синхронизации не влияют задержки передачи данных по ЛВС.

Во второй серии измерений обработчики сопряжённых мастеров запускаются на различных ЭВМ, соединённых ЛВС, что вносит дополнительную погрешность при сопряжении движущихся изображений с помощью условия (3.29), см. с. 140, п. 3.5.

n	x	y	x_0	y_0	Δx	Δy
1	80	256	81	256	-1	0
2	260	249	260	248	0	1
3	4	182	4	180	0	2
4	182	158	181	157	1	1
5	64	167	65	166	-1	1
6	247	142	247	142	0	0
7	60	139	59	139	1	0
8	243	109	243	111	0	-2
9	22	143	22	143	0	0
10	201	114	200	115	1	-1
11	42	121	41	120	1	1
12	225	86	224	85	1	1
13	116	133	117	132	-1	1
14	307	103	307	103	0	0
15	36	83	35	84	1	-1
16	221	44	223	44	-2	0
17	60	90	59	89	1	1
18	245	51	245	52	0	-1
19	107	109	108	108	-1	1
20	299	70	299	70	0	0
21	89	70	88	71	1	-1
22	282	28	283	26	-1	2

Таблица 7.9. Таблицы измерения погрешности сопряжения, n — порядковый номер измерения, (x, y) — расчётные координаты точки преобразования, (x_0, y_0) — действительные координаты точки преобразования значение склонения, $(\Delta x, \Delta y)$ — отклонение экспериментальных значений от координат реперных точек.

Во обоих случаях необходимо в терминальном приложении настроить зоны тревоги, совпадающими с зонами пересечения областей зрения на изображениях мастер-камер (см. рис. 7.11). Это упростит поиск объектов, которые побывали в зоне пересечения областей зрения неподвижных видеокамер.

После запуска программного обеспечения серверов необходимо получить не менее 1000 изображений объектов из зоны пересечения областей зрения мастер-камер, причём это должны быть изображения людей, поскольку другие изображения в ходе эксперимента не рассматриваются. Изображения данных объектов возможно вывести в терминальном модуле, задав в условиях поиска то, что объект побывал в зоне тревоги, и классифицирован как человек. Информация о том, что данное изображение

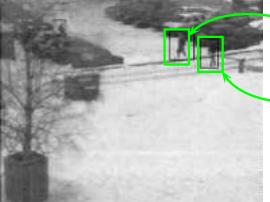
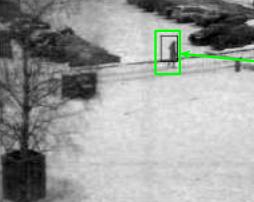
Левый мастер	Правый мастер	Левый мастер	Правый мастер
			
			
			
			
			

Таблица 7.10. Статистика сопоставления объектов на сопряжённых мастерах. Обозначение  указывает на успешное сопоставление, обозначение  применяется для объектов, которые не были сопоставлены с помощью алгоритма сопряжения.

было сопоставлено с изображением на сопряжённой мастер-камере, будет присутствовать в записях об объектах, которые выводит терминальный модуль.

Для подсчёта статистики представим изображения объектов вместе с информацией о их сопоставлении в таблице 7.10. В итоге для первой серии измерений получим $S_x = 10$ промахов из $S = 1000$, для второй серии — $S_x = 107$ промахов из $S = 1000$.

Чтобы воспользоваться формулой (3.32), которая оценивает погрешность Δx при помощи функции ошибок F^{-1} как

$$\Delta x = \frac{3\Delta}{F^{-1}(1 - S_x/2S)},$$

необходимо оценить значение Δ , которое соответствует промаху при сопоставлении изображений людей. Поскольку в условиях эксперимента сопоставляемые изображе-

ния людей находятся на расстоянии 20 метров от точки закрепления камер, и у обоих мастер-камер угол зрения 30° , то, считая видимую ширину человека примерно 0,4 метра, а ширину изображения мастеров 352 пикселя, получим $\Delta \approx \frac{0,4}{20} \cdot \frac{180}{3,14 \cdot 30^\circ} \cdot 352 \approx 10$ пикселей.

С помощью формулы, приведённой выше, становится возможным получить окончательный результат, который приводится в таблице 7.11, см. ниже.

	доля промахов	погрешность
1-ая серия	0,010	6 пикселей
2-ая серия	0,107	10 пикселей

Таблица 7.11. Результаты оценки погрешности сопряжения движущихся изображений в зоне пересечения областей зрения неподвижных камер.

Из анализа таблицы 7.11 следует, что в первой серии измерений задержки синхронизации не внесли существенного дополнительного вклада в погрешность сопряжения, поскольку погрешность 6 пикселей сопоставима с погрешностью 4 пикселя, полученной для неподвижных изображений. Во второй же серии измерений, при синхронизации данных посредством ЛВС, задержки синхронизации примерно в два раза увеличивают погрешность сопряжения, что существенно влияет на долю промахов, то есть на количество неверно сопоставленных объектов.

7.8. Экспериментальная оценка эффективности автоматизированной системы тестирования алгоритмов семантической обработки изображений

В рамках данной диссертационной работы автором был построена программная система для автоматического тестирования алгоритмов семантической обработки изображений на основе теоретических результатов, полученных в главе 4. Данная система для анализа эффективности алгоритмов обработки изображений получила название «Супервизор». Главное назначение данной системы заключается в возможности тестировать алгоритмы семантической обработки, обрабатывая потоки изображений, записанные в видеофайлы. Основные возможности системы тестирования «Супервизор» заключаются в:

- 1) предоставлении пользовательского интерфейса для сбора видеофайлов, объединения их в группы по различным категориям классификацию с последующим групповым анализом характеристик обработки алгоритмами семантического сжатия;
- 2) визуальном представлении математического описания $V^n = \{V_i^n\}, \{I_l^n\}, \{A_m^{an}\}$ автоматической и идеальной разметки видеороликов, которое введено в п. 4.2 (V^n — видеокадры, I_l^n — объекты ручной разметки, A_m^{an} — объекты идеальной разметки);

3) вычислении и демонстрации введённых в п. 4.2 показателей эффективности алгоритмов семантической обработки $F_P^{\alpha\beta}$ (число ошибок первого рода), $F_N^{\alpha\beta}$ (число ошибок второго рода), Q^α (доля времени автоматического сопровождения), E^α (частота ошибочной смены объекта сопровождения).

К важной особенности системы «Супервизор» следует отнести возможность распределённой обработки видеоматериала V^n при вычислении характеристик $F_P^{\alpha\beta}$ и $F_N^{\alpha\beta}$. В текущем исполнении системы «Супервизор», имеющемся в распоряжении автора, благодаря параллельной обработке видеороликов с использованием нескольких ЭВМ удаётся сократить суточный цикл тестовой обработки до 3–4 часов.

Рассмотрим конкретные практические примеры использования системы тестирования алгоритмов семантической обработки. В распоряжении автора имеются цифровые записи с видеокамер, установленных на периметре охраняемых объектов, общей длительностью около 2 часов, что составляет примерно 180 тысяч кадров. Данные видеопоследовательности подвергнуты ручной разметке, получено 150 объектов I_l^n . Видеоролики содержат проблемные фрагменты с точки зрения алгоритмов семантического сжатия, на них зафиксированы различные помехи, как правило, приводящие к ошибкам первого и второго рода: изменение освещённости сцены из-за переменной облачности днём и из-за световых искусственных помех ночью (например, свет фар автомобилей вне области наблюдения), движение травы и листвы деревьев от ветра, а также их теней на отражающих поверхностях сцены, и многие другие помехи.

В таблицу 7.12 сведены результаты оценки качества различных алгоритмов семантической обработки, которые формировались по мере разработки и совершенствования комплексов технического зрения «Orwell2k». Алгоритмы в данной таблицы обозначены годами, в которые они появлялись, далее приведены комментарии, описывающие нововведения более новых версий алгоритмов семантической обработки по сравнению с предыдущими.

Алгоритм, используемый для выделения движущихся объектов в 2002 году в системе «Orwell2k», основан на статье [15] и представляет собой вычитание фона с одномодальной моделью статистики фонового изображения, с последующей сегментацией движущихся областей и применением алгоритмов построения траектории движущихся объектов на основе поиска оптимальных паросочетаний в двудольном графе уже найденных объектов и вновь полученных областей движения. Данный подход является базовым для всех остальных алгоритмов, перечисленных далее.

В 2003 году базовый алгоритм был дополнен этапом билатеральной нормализации изображений, преобразующий изображение перед обработкой таким образом, что на него почти не оказывает влияние изменение интенсивности из-за переменной облачности и внешней засветки, если это изменение в пределах динамического диапазона

чувствительности камеры.

В 2008 году были внедрены фильтры выделенных объектов, основанные на движении особых точек (см. [17, 18, 173], а также п. 6.2) и, дополнительно, на применении схемы вычитания статистического фона в пространстве норм и углов градиентов функции интенсивности кадров изображения видеокамеры. Для того, чтобы движущиеся объекты были зафиксированы системой, дополнительно накладываются условия согласованного движения особых точек в окрестности объектов и наличие существенных изменений по направлению и норме градиентов по сравнению с фоновыми.

В 2012 фильтры особых точек и градиентный фильтр были доработаны таким образом, чтобы положение объекта определяло чувствительность фильтров: объекты на дальнем плане с маленьким размером изображения подвергаются фильтрации с более низким порогом, чем объекты на ближнем плане с большим размером изображения. Дополнительно для фильтрации объектов засветки был применён фильтр плотного оптического потока, т.е. расчёт поля смещений не только в особых точках, но и во всех точках окрестности движущегося объекта. Этот расчёт производится с целью установить согласованность смещений точек объекта с физической моделью движения.

Алгоритмы	$F_P^{\alpha\beta}$	$F_N^{\alpha\beta}$	Q^α	Время обработки кадра, мс
2002	>10000	20	60%	80
2003	≈ 500	15	60%	40
2008	50	10	55%	50
2012	20	10	50%	50
2012 с фильтром оптического потока	10	12	47%	55

Таблица 7.12. Улучшение показателей алгоритмов семантической обработки по мере совершенствования комплексов «Orwell2k».

Анализ таблицы 7.12 показывает, что количество ошибок первого и второго рода, которые допускают алгоритмы семантической обработки системы «Orwell2k», в целом, уменьшалось, правда, ценой некоторого уменьшения доли времени сопровождения объектов в течение его присутствия в поле зрения камер, а также ценой увеличения требований к производительности ЭВМ, выполняющих обработку.

Система «Супервизор» действует с 2005 года, и именно благодаря ей удалось выполнить столь существенную оптимизацию — на порядок сократить число ошибок первого рода, при этом несколько уменьшив количество ошибок второго рода, по-

скольку тестирование с помощью данной системы позволило выделить, классифицировать и затем исправить конкретные ошибки алгоритмов семантической обработки.

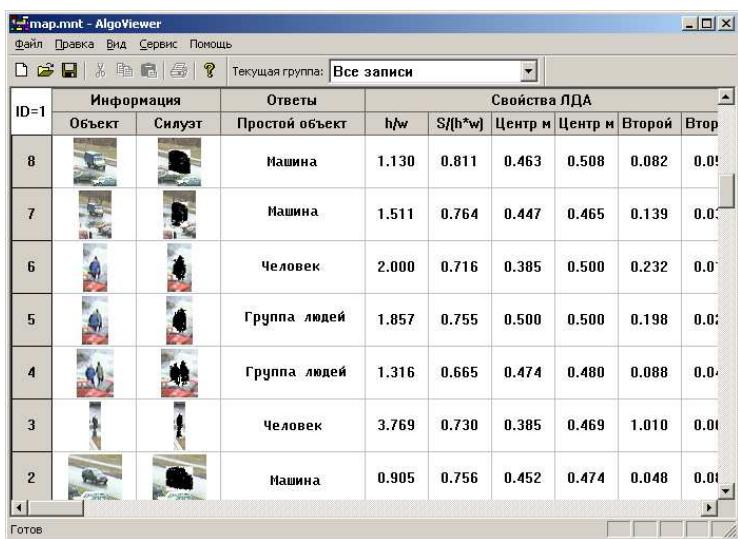
Несмотря на то, что ошибки устраниены не полностью, текущий уровень качества системы обнаружения подвижных объектов системы «Orwell2k» признаётся лучшим в мире. Подробный анализ ошибок показывает, что пропущенные объекты обладают малой контрастностью, находятся на дальнем плане, и не совершают достаточных (согласно паспортным характеристикам системы) угловых смещений по отношению к камерам, в поле зрения которых они попадают, а ложные объекты появляются вследствие помех, которые затрудняются классифицировать даже операторы системы. Ряд протоколов приложения 2 подтверждают данный тезис.

Другой пример эффективности предложенной системы тестирования алгоритмов семантической разработки — это процесс отладки алгоритмов, описанных в главе 5 и внедрённых в устройство подсчёта посетителей «Statistics-3d». Прежде, чем устройство достигло показателей точности подсчёта 97%, что зафиксировано в протоколе испытаний в аэропорту «Шереметьево» (см. приложение 2), последовательность применения фильтров алгоритма, описанного в 5.4, все пороговые параметры данного алгоритма подбирались с помощью тестовых запусков на больших выборках размещенного видеоматериала.

В заключении по данному параграфу хотелось бы отметить общую тенденцию в разработке систем технического зрения: арсенал методов семантической обработки в настоящее время довольно богат, но для решения конкретной задачи с его помощью необходимо подбирать последовательность применения конкретных алгоритмов семантической обработки, а также параметры работы этих алгоритмов. В чём-то этот процесс схож с процессом машинного обучения, например, с процессом поиска коэффициентов свёрточной нейронной сети. Порой наличие большого объёма видеоматериала с качественной разметкой имеет большее значение для создания алгоритма семантической обработки, чем сам процесс создания алгоритмов. В этих условиях средства тестирования, аналогичные предложенной системе «Супервизор», имеют критическое влияние на разработку систем технического зрения с элементами искусственного интеллекта.

7.9. Результаты применения алгоритмов семантического сжатия изображений. Оценка достоверности

В данном параграфе будут использоваться обозначения и определения, введённые в п. 1.4, такие как тестовое множество T и обучающее множество L , алгоритм распознавания A и другие. Также будет использоваться введённый в п. 3.7 ряд обозначений для параметров-признаков, рассчитываемых на основе силуэта, выделенного алгоритмами анализа движения на последовательностях изображений от мастеров.



Строчки — множество объектов обучения

Столбцы — свойства объектов: изображение, силуэт, класс, параметры-признаки

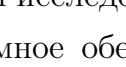
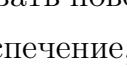
ID=1	Информация		Ответы		Свойства ЛДА					
	Объект	Силуэт	Простой объект		h/w	S/(h*w)	Центр м	Центр м	Второй	Втор
8			Машина		1.130	0.811	0.463	0.508	0.082	0.01
7			Машина		1.511	0.764	0.447	0.465	0.139	0.01
6			Человек		2.000	0.716	0.385	0.500	0.232	0.01
5			Группа людей		1.857	0.755	0.500	0.500	0.198	0.01
4			Группа людей		1.316	0.665	0.474	0.480	0.088	0.01
3			Человек		3.769	0.730	0.385	0.469	1.010	0.01
2			Машина		0.905	0.756	0.452	0.474	0.048	0.01

Рисунок 7.14. Внешний вид программы тестирования алгоритмов классификации.

Для того, чтобы исследовать поведение алгоритмов распознавания, требуется специальное программное обеспечение, в котором можно выполнять следующие действия:

- 1) отображать список объектов, выделенных алгоритмами анализа изображений, вместе с различными их свойствами: изображение объекта и его силуэт, значения параметров-признаков, предполагаемый класс объекта и действительный класс объекта;
- 2) для каждого объекта предоставлять возможность изменять вручную действительный класс объекта с целью формирования тестовых и обучающих множеств;
- 3) предоставлять возможность выделения отдельных множеств объектов;
- 4) запускать процедуры этапа обучения алгоритмов распознавания различных типов, подавая на вход подмножества параметров, и множество объектов обучения;
- 5) выводить результаты применения обученных алгоритмов к тестовым множествам объектов;
- 6) запускать процесс обучения с перебором типов алгоритмов и наборов параметров-признаков для фиксированной пары тестового и обучающего множеств с целью выбора наилучшего алгоритма по показателям достоверности распознавания, причём при сравнении двух алгоритмов распознавания можно пользоваться теоретическими результатами, описанными в [211].

В ходе диссертационной работы для выполнения данных задач была разработана специальная программа, представленная на рис. 7.14. С её помощью получены показатели достоверности для алгоритмов классификации, использующихся в системах, аналогичных «СПВ ОВТ». Все функции, необходимые для подбора оптимального алгоритма распознавания и для подбора оптимального набора типов признаков

при классификации, в данной программе заложены, однако прежде, чем запускать процедуру перебора, необходимо выяснить, какой объём тестовой выборки множества объектов необходим, чтобы полученная достоверность распознавания обладала приемлемой погрешностью. Если этого не сделать, то сравнение двух различных показателей достоверности для различных тестовых выборок не имеют смысла из-за неизвестного доверительного интервала обоих показателей.

Оценка объёма тестовой выборки[164]. Обозначим мощность тестового множества T как n . Воспользуемся известными фактами из математической статистики для получения нужной нам оценки.

Представим работу алгоритма A при распознавании объектов множества T в виде процесса Бернулли. Будем поочередно посыпать на вход алгоритма объекты, и сопоставим на i -ом опыте $x_i = 1$, если алгоритм верно классифицировал объект, и $x_i = 0$, если неверно. Алгоритм работает с некоторой точностью. Пусть p — вероятность верного распознавания текущего объекта алгоритмом, соответственно $(1 - p)$ — вероятность ошибки. В качестве несмешенной оценки вероятности p возьмем $\tilde{p} = \bar{x} = (\sum_{i=1}^n x_i)/n$, где n — количество опытов, совпадающее в данном случае с мощностью тестового множества. Дисперсия оценки

$$D\tilde{p} = (\sum_{i=1}^n Dx_i)/n^2 = n \cdot Dx_i/n^2 = p(1 - p)/n$$

в предположении независимости опытов (так можно считать, если тестовая выборка достаточно репрезентативна). Среднеквадратичное отклонение равно

$$\sigma = \sqrt{D\tilde{p}} = \frac{\sqrt{p(1 - p)}}{\sqrt{n}} \quad (7.1)$$

Долю правильных ответов мы можем рассматривать как несмешенную оценку в приведенном процессе Бернулли, тогда потребуется замена в формуле (7.1) n на $n - 1$, что при большом количестве примеров несущественно. По центральной предельной теореме мы можем считать оценку \tilde{p} распределенной нормально, поэтому значение 3σ характеризует отклонение оценки от истинного значения в 97,4% случаев. Для того, чтобы считать оценку достоверной, необходимо убедиться, что значение 3σ не превосходит допустимого для нас отклонения оценки.

Например, если мы получаем в результате работы алгоритма долю правильных ответов $\tilde{p} = 0,9$, то приемлемым интервалом можно считать $3\sigma = \pm 0,06$. Тогда из формулы (7.1) получаем

$$\sigma = \frac{\sqrt{0,09}}{\sqrt{n}} = \frac{0,3}{\sqrt{n}} < 0,02,$$

то есть для обеспечения необходимой точности требуется $n > (0,3/0,02)^2 = 225$ примеров.

Если мы хотим получить подтверждение, что наша оценка $\tilde{p} = 0,99$ достоверна, то, взяв $3\sigma = \pm 0,006$, получаем

$$\frac{\sqrt{0,0099}}{\sqrt{n}} \approx \frac{0,1}{\sqrt{n}} < 0,002,$$

откуда $n > (0,1/0,002)^2 = 2500$. Аналогичные расчеты легко произвести и для других значений оценки p и отклонения 3σ . Если $p > 0,99$, то принимая $p \approx 1$ и $\sigma = (1-p)/5$, получаем

$$n > \frac{25}{1-p}.$$

В нашем случае интересна оценка количества примеров для $p = 0,95$ при $3\sigma = 0,02$. Необходимо не менее 1000 объектов, чтобы обеспечить такую точность.

Распознавание объектов, полученных с мастер-камер. При распознавании движущихся объектов по классам «Человек», «Машина», «Группа людей» наилучший результат был достигнут после обучения нейросети по параметрам: Mx^2 , My^2 , $\frac{h}{w}$, $\frac{S}{wh}$, $\frac{Mx}{My}$, H . Нейросеть состояла из 10 нейронов на первом и втором уровне, и обязательных трёх нейронах на последнем, третьем уровне. Полученные результаты для тестового множества из 1231 объекта отображены в кросс-таблице 7.13.

	Человек	Машина	Группа людей	Отказы	Всего
Человек	445	2	4	1	452
Машина	5	438	15	12	470
Группа людей	3	12	291	3	309
Статистика	98%	97%	94%	2%	95%

Таблица 7.13. Результаты распознавания «Человек», «Машина», «Группа людей».

Показатели в реальной работе системы несколько ниже, чем на тестовом множестве, поскольку в реальной системе не всегда удаётся отфильтровать шумовые объекты, не входящие в тестовое множество.

Распознавание объектов по типам «Обычный», «Оставленный», «Остановившийся» построено на основе параметров t_s и f_s и эвристике, аналогичной описанной в пункте 1.4. Доля правильно классифицированных объектов по этой системе классификации составляет около 90%.

Распознавание кресла в кинотеатре как пустого, занятого человеком, либо занятого вещью, проходит по параметрам r_1 и r_2 согласно эвристическому подходу, описанному в п. 1.4: вначале проверяется, отличается ли текущее изображение кресла от пустого по параметру r_1 , и, если отличается, то по параметру r_2 устанавливается, было ли движение в кресле (т.е. в кресле находится человек), либо его нет (т.е. в кресле находятся вещи). Доля правильных ответов примерно равна 95%.

Распознавание с привлечением слейв-камер. Из-за частых промахов при наведении некоторых поворотных камер на транспортные средства возникла необходимость отнести полученные снимки к одной из двух категорий: на снимке присутствует транспортное средство, и транспортного средства на снимке нет. Причём параметры x'_{min} , x'_{max} , y'_{min} , y'_{max} не всегда доступны для изображения на поворотной камере, поскольку за время поворота слейва объект на мастере может покинуть поле зрения камеры. Оказывается, проверка на превышение параметра Λ некоторого порога позволяет провести такую классификацию с точностью 90%.

Достигнутые показатели классификации объектов позволяют расширить область применения аналитических видеосистем для решения ряда практических задач, таких, как наблюдение за городскими территориями с целью фиксации происшествий в базе данных, для наблюдения в кинозалах и для других приложений.

Выводы

1. Проведён функциональный анализ приборно-блочной схемы обобщённой системы технического зрения на примере системы «СПВ ОВТ», РАЯЖ 46652.001-ОС.П3, разработанной с участием автора диссертации. Показано, что данная система за счёт автоматизации функциональных блоков и семантической обработки изображений в 20 раз быстрее выполняет действия, аналогичные действиям систем, рассмотренных в главе 1. Проведено экспериментальное сравнение подобных систем с другими системами технического зрения, которое показало, что в рамках предложенной методики сравнения система «СПВ ОВТ» обладает наилучшими характеристиками достоверности идентификации событий в поле действия датчиков системы.

2. Сравнительные испытания по методике, предложенной автором, показали, что анализ панорамных изображений требует доработки как технических, так и программных средств, тем не менее, являясь при этом перспективным способом, позволяющим снизить в 3–10 раз количество используемых видеокамер в охранных системах технического зрения.

3. Разработаны новые методики измерения погрешности наведения поворотных видеокамер и погрешности сопряжения изображений неподвижных видеокамер с пересечением зон наблюдения. Измерена погрешность наведения поворотных видеокамер на неподвижные объекты при использовании методов сопряжения, описанных в диссертационной работе. Погрешность составила $0,5^\circ$, что совпадает с теоретической оценкой, полученной ранее.

4. Экспериментально обоснована эффективность алгоритма наведения с адаптивным упреждением, разработанного в диссертационной работе: измеренная погрешность наведения составляет 3° , что более чем в два раза меньше погрешности при

применении простейшего алгоритма наведения без упреждения. Наведение с линейным упреждением вместо наведения с адаптивным упреждением приводит к увеличению количества промахов более, чем в 10 раз.

5. Разработанный в диссертации алгоритм наведения с адаптивным упреждением позволяет автоматически получать изображения объектов с разрешением в 10 раз большим, чем разрешение изображения того же объекта на неподвижной камере, при этом не требуется увеличение количества видеокамер в сотни раз.

6. В результате обработки массива экспериментальных данных вычислена погрешность сопряжения неподвижных изображений для мастер-камер с общей зоной обзора, которая составила 4 пикселя. В то же время для подвижных изображений в условиях локальной синхронизации данных погрешность сопряжения 6 пикселей, а при сетевой синхронизации — 10 пикселей. Увеличение погрешности сопряжения в 2 раза привело к увеличению количества ошибок сопоставления более, чем в 10 раз, что подтверждает целесообразность обработки изображений сопряжённых мастеров на одной ЭВМ.

7. Измерена доля ошибок при сопоставлении изображений объектов, выделенных обработчиками мастер-камер, запущенных на одной и той же ЭВМ. По результатам измерений доля промахов составила 1%.

8. Экспериментально доказана эффективность предложенной в работе системы тестирования алгоритмов семантической обработки изображений: показано, что применение данной тестовой системы позволило разработать алгоритмы идентификации объектов в поле зрения камер и стереосистем со сниженной на 2 порядка частотой ошибок первого рода по сравнению с общеизвестными алгоритмами семантической обработки.

9. Проведён синтез наборов признаков классификации изображений, для которых измерена достоверность классификации при применении алгоритмов распознавания. Достоверность классификации достигает уровня 90%–95%.

10. Применение видеосистем «Orwell2k», в которой внедрены результаты диссертационной работы, позволяет реализовать принципиально новые подходы при обеспечении мер безопасности и повысить производительность труда операторов аналитических видеосистем. С помощью систем технического зрения «Orwell2k» охраняются стратегически важные объекты: аэропорты («Шереметьево», «Домодедово», «Пулково», «Череповец»), крупные промышленные предприятия (Бурейская ГЭС, Курская АЭС, нефтеперерабатывающий завод Оренбургской области), общественные территории городов (Нижний Новгород, Саранск, Сочи), объекты транспортной инфраструктуры и другие.

Заключение

При выполнении диссертационной работы в рамках решения поставленной научной проблемы по повышению эффективности приборных комплексов технического зрения на основе алгоритмов семантического сжатия, синхронизации и сопряжения семантических обработчиков и автоматического управления видеоприборами достигнуты следующие результаты.

1. Впервые разработана теория и выведены уравнения автоматизации начальной установки и последующего параллельного сопряжённого функционирования приборов семантической обработки изображений в комплексах технического зрения.
2. Выведены математические соотношения для сопряжения приборов технического зрения и разработаны новые алгоритмы сопоставления изображений в общей зоне контроля приборов и управления поворотными камерами с адаптивным упреждением, которые обеспечивают достоверность сопоставления изображений 99,0%, малые погрешности наведения для неподвижных ($0,5^\circ$) и движущихся (3°) объектов, а также повышение разрешения в 10 раз при значительном уменьшении количества приборов в комплексах технического зрения с обширными областями контроля, причём время автоматического наведения в 20 раз меньше времени ручного наведения.
3. Разработана теория тестирования и контроля алгоритмов семантического сжатия с применением формальной логики и теории множеств, и на основе этой теории построен прототип системы тестирования.
4. Создан и внедрён новый комбинированный алгоритм обработки моно- и стереоизображений для оценки плотности движения наблюдаемых объектов с погрешностью 3%.
5. Созданы новые алгоритмы семантического анализа панорамных изображений, позволяющие уменьшить в 3–10 раз количество видеоприборов в приборных комплексах технического зрения за счёт поворотных устройств.
6. Разработанные в диссертации алгоритмы и способы позволили создать приборные комплексы технического зрения нового поколения с параметрами выше мирового уровня, обеспечивающие автоматическое обнаружение, наведение и сопровождение объектов, значительное улучшение качества их визуализации, классификации и идентификации и автоматическое формирование сигналов управления устройствами предупреждения и заграждения.
7. Результаты диссертационной работы применены в программно-аппаратных комплексах «Orwell2k», разработанных при непосредственном участии автора, что подтверждается рядом свидетельств о регистрации программ и патентами на полезные модели и изобретения, и используются для эффективного решения задач

обеспечения безопасности, например, в аэропортах «Шереметьево», «Домодедово», «Пулково», «Череповец», на крупных промышленных предприятиях и объектах стратегического значения, например, Бурейской ГЭС, Курской АЭС, нефтеперерабатывающем заводе Оренбургской области, в городах Нижний Новгород, Саранск, Сочи при контроле общественных территорий, на железной дороге Сочи – Красная поляна (со времени проведения зимних Олимпийских игр 2014 года) и на других важных объектах государственной инфраструктуры.

Таким образом, представленная диссертация является законченной научно-квалификационной работой, в которой в рамках приоритетных направлений развития науки, технологий и техники, утверждённых указом Президента РФ от 7 июля 2011 г. №899, автором разработаны теоретические положения и научно обоснованные технические решения, а также созданы высокоэффективные алгоритмы, программные средства и приборные комплексы технического зрения, внедрение которых вносит значительный вклад в развитие экономики страны и в повышение её обороноспособности.

Список литературы

1. Петричкович Я.Я. Перспективы развития систем с применением компьютерного зрения. —М.: из-во «Компания Спутник+», Естественные и технические науки, 2013, №1, С. 238–246.
2. Гарсиа М. Проектирование и оценка систем физической защиты. —М.: Мир, 2002.
3. Введенский Б. Современные системы охраны периметров. Журнал «Алгоритм безопасности», 2003, №4.
4. Линев Н.В., Никитин А.А., Климов А.В. Раннее обнаружение несанкционированного проникновения: аспекты практической реализации. Журнал «Системы безопасности, связи и телекоммуникаций», 1999, №27, С. 24.
5. Ярославский Л.П. Введение в цифровую обработку изображений, Советское радио, 1979.
6. Грузман И.С., Киричук В.С., Косых В.П., Перетягин Г.И., Спектор А.А. Цифровая обработка изображений в информационных системах: Учебное пособие. —Новосибирск: Издательство НГТУ, 2000, 168с.
7. Применение цифровой обработки сигналов, под ред. Э. Оппенгейма. —М.: Мир, 1980.
8. Andrews H.C., Hunt B.R. Digital image restoration //Prentice-Hall Signal Processing Series, Englewood Cliffs: Prentice-Hall, 1977, Т. 1.
9. Злобин В.К., Еремеев В.В., Кузнецов А.Е. Обработка изображений в геоинформационных системах: Учебное пособие. Рязань: Рязан. гос. радиотехн. университет. 2006.
10. Прэtt Е. Цифровая обработка изображений. — М.: Мир, 1982, кн. 1.
11. Алпатов Б.А. и др. Методы автоматического обнаружения и сопровождения объектов. Обработка изображений и управление. —М.: Радиотехника, 2008, С. 176.
12. Алпатов Б.А., Муравьев С.И., Муравьев В.С. Пространственный алгоритм выделения объектов на основе адаптивной пороговой обработки. Тезисы докладов VIII международной конференции Цифровая обработка сигналов и её применение, 2006, С. 445–448.
13. Алпатов Б.А. Оптимальное оценивание параметров движущегося объекта в последовательности изображений. Автометрия, 1994, №2, С. 32–37.
14. Алпатов Б.А., Блохин А.Н. Модели и алгоритмы обнаружения и выделения движущихся фрагментов изображений. Автометрия, 1995, №4, С. 100–104.
15. Kanade T., Collins R., Lipton A., Anandan P., Burt P., Cooperative multisensor video surveillance. Proc. of the DARPA Image Understanding Workshop, May 1997, volume 1, pp. 3–10.

16. Dorin Comaniciu V.R., Meer P. Kernel-based object tracking. *Transactions on Pattern Analysis and Machine Intelligence* 25, May 2003.
17. Beymer D., McLauchlan P., Coifman B., Malik J. A real-time computer vision system for measuring traffic parameters. *IEEE Proc. CVPR*, 1997, pp. 495–501.
18. Coifman B., Beymer D., McLauchlan P., Malik J. A real-time computer vision system for vehicle tracking and traffic surveillance. *Transportation Research: Part C*, vol. 6, №4, 1998, pp. 271–288.
19. Kollery D., Daniilidis K., Nagel H.-H. Model-based object tracking in monocular image sequences of road traffic scenes. *International journal of Computer Vision*, №10, 1993, pp. 257-281.
20. Sen-Ching S. C., Kamath C. Robust techniques for background subtraction in urban traffic video. *Electronic Imaging 2004*. International Society for Optics and Photonics, 2004, pp. 881-892.
21. Харкевич А.А. Опознание образов. *Радиотехника*, 1959, Т. 14, №5, С. 12–22.
22. Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 1958, Т. 65, №6, С. 386.
23. Ковалевский В.А. Корреляционный метод распознавания изображений. *Журнал вычислительной математики и математической физики*, 1962, Т. 2, №4, С. 684–694.
24. Айзerman М.А., Браверман Э.М., Розоноэр Л.И. Вероятностная задача об обучении автоматов распознаванию классов и метод потенциальных функций. *Автоматика и телемеханика*, 1964, Т. 25, №9.
25. Галушкин А.И., Единый подход к решению задач обучения и самообучения систем распознавания образов. *Труды МИЭМ*, 1970, вып. 6, С. 104–120.
26. Ивахненко А.Г. Самообучающиеся системы распознавания и автоматического управления. *Техника*, 1969.
27. Горелик А.Л., Скрипкин В.А. Некоторые вопросы построения систем распознавания. *Сов. радио*, 1974.
28. Вапник В.Н., Червоненкис А.Я. Теория распознавания образов. Статистические проблемы обучения. —М.: Наука, 1974, 274 С.
29. Журавлёв Ю.И. Непараметрические задачи распознавания образов. *Кибернетика*, 1976, №6.
30. Дуда Р., Харт П. Распознавание образов и анализ сцен. —М.: Мир, 1976.
31. Злобин В.К., Ручкин В.Н. Нейросети и нейрокомпьютеры: учеб. пособие. —СПб.: БХВ-Петербург, 2011.

32. Анисимов Б.В., Курганов В.Д., Злобин В.К. Распознавание и цифровая обработка изображений: Учеб. пособие для вузов. — Высш. шк., 1983.
33. Krizhevsky A., Sutskever I., Hinton G.E. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012, pp. 1097–1105.
34. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. — 2014.
35. Szegedy C. et al. Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
36. He K. et al. Deep Residual Learning for Image Recognition. arXiv preprint arXiv:1512.03385. — 2015.
37. Ren S. et al. Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems, 2015, pp. 91–99.
38. Kanade T., Collins R., Lipton A., Burt P., Wixson L. Advances in cooperative multisensor video surveillance. Proc. of the DARPA Image Understanding Workshop, November 1998, volume 1, pp. 3–24.
39. <http://pascallin.ecs.soton.ac.uk/challenges/VOC/> (дата обращения 29.07.2014).
40. <http://image-net.org/challenges/LSVRC/2013/> (дата обращения 29.07.2014).
41. Петрикович Я. Я. Семантическое сжатие. Вторая IP-революция. <http://elvees.ru/home/uploads/media/ip2.pdf> (дата обращения 02.05.2016).
42. Пименов А.В. Интеллектуальный видеоанализ событий в IP-системах видеонаблюдения. Журнал «Технологии защиты», 2010, №4.
43. Любимов А.В., Соколов Е.Г., Соколов Е.Е. Некоторые инновационные технологии в области пожарной безопасности. Журнал «Глобальная безопасность», специальный выпуск «Сочи 2014».
44. http://elvees.ru/home/uploads/media/publikacija032013_02.pdf (дата обращения 29.07.2014).
45. http://elvees.ru/home/uploads/media/stopgz_001_14_17.pdf (дата обращения 29.07.2014).
46. Хамухин А.В. Анализ применения стереосистем для решения задач бизнес-мониторинга. —М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки», 2013, №6.
47. Журавлёв Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации. Проблемы кибернетики, 1978, Т.33, с.5 –68.

48. Ахмед Н., Рао К.Р. Ортогональные преобразования при обработке цифровых сигналов/под ред. Фоменко И.Б. —М.: Связь, 1980.
49. Зворыкин В.К. Телевидение. —М.: Успехи физических наук, 1934, т. XIV, с. 778–807.
50. Бондарев М.В., Хоббихожин Ш.А., Щагин А.В. Сиситема видеоотображения дорожной ситуации для водителя автомобиля с выводом текущей информации на цветной дисплей установленный на передней панели автомобиля, перед водителем. Международный сертификат объекта интеллектуальной собственности RU02R1RU20020020 SR1.
51. <http://www.sanyo-electric.ru/products/positions/231.htm>
52. Boyle W.S., Smith G.E. Charge coupled semiconductor devices. Bell Syst. Tech. J., 1970, №49, pp. 587–93.
53. Haritaoglu I., Harwood D., Davis L. S. W4: real-time surveillance of people and their activities. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000, №8, vol. 22, pp. 809–830
54. <http://www.iomniscient.com/>
55. <http://www.ioimage.com/?p=Category&CategoryID=180&ParentID=624>
56. Интелект — интегрированный комплекс безопасности.
<http://www.itv.ru/products/Intelligence/>
57. Goal v8. Руководство пользователя». http://www.goal.ru/manual/instr_GOALv8.zip, 2004
58. Петричкович Я.Я., Солохина Т.В. SoC серии «МУЛЬТИКОР» — первый шаг и положительная динамика развития. Компоненты и технологии, 2003, №5, С. 104–106.
59. Booch G. Object-Oriented Analysis and Design with Applications. Benjamin/Cummings, 1994, ISBN 0-8053-5340-2.
60. Standard for the C++ programming language. ISO/IEC 14882, 1998.
61. Stroustrup B. The C++ programming language. AT&T Labs Florham Park, New Jersey, 2004.
62. Даджеон Д., Мерсеро Р. Цифровая обработка многомерных сигналов». —М.: Мир, 1988.
63. Keys R. G. Cubic convolution interpolation for digital image processing. Acoustics, Speech and Signal Processing, IEEE Transactions on, 1981, Т. 29, №. 6, pp. 1153-1160.
64. Kindermann R. et al. Markov random fields and their applications. —Providence, RI: American Mathematical Society, 1980, Т. 1.

65. Onsager L. Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review*, 1944, Т. 65, №. 3–4, pp.117–149.
66. Ising E. Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei*, 1925, Т. 31, №. 1, pp. 253-258.
67. Lenz W. Beiträge zum Verständnis der magnetischen Eigenschaften in festen Körpern. *Physikalische Zeitschrift* 21, 1920, pp. 613–615.
68. Freeman W. T., Jones T. R., Pasztor E. C. Example-based super-resolution. *Computer Graphics and Applications*, IEEE, 2002, Т. 22, №. 2, pp. 56–65.
69. Besag J. Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 1974, pp. 192–236.
70. Geman S., Geman D. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 1984, №. 6, pp. 721–741.
71. Блейхут Р. Быстрые алгоритмы цифровой обработки сигналов. —М.: Мир, 1989.
72. Barnea D.I., Silverman H.F. A class of algorithms for fast digital image registration. *IEEE Trans. Computers*, №21, 1972, pp. 179–186.
73. Lewis J.P. Fast template matching. *Vision Interface conference*, 1995, pp. 120–123.
74. Steven L. Kilthau, Mark S. Drew and Torsten Möller. Full search content independent block matching based on the fast Fourier transform. *IEEE ICIP*, I, 2002, pp. 669–672.
75. Lucas B.D. et al. An iterative image registration technique with an application to stereo vision. *IJCAI*, 1981, Т. 81, pp. 674–679.
76. Солдатов С.А., Стрельников К.Н., Ватолин Д.С. Быстрое и надежное определение глобального движения в видеопоследовательностях. 16-я Международная конференция по компьютерной графике и ее приложениям, Институт вычислительной математики и математической геофизики СО РАН, 2006, с. 430–437.
77. Stauffer C., Grimson W. E. L. Learning patterns of activity using real-time tracking. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2000, Т. 22, №. 8, pp. 747–757.
78. Matsushita Y. et al. Illumination normalization with time-dependent intrinsic images for video surveillance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2004, Т. 26, №. 10, pp. 1336–1347.
79. Prati A. et al. Shadow detection algorithms for traffic flow analysis: a comparative study. *Intelligent Transportation Systems, Proceedings, IEEE*, 2001, pp. 340–345.
80. Stander J., Mech R., Ostermann J. Detection of moving cast shadows for object segmentation. *Multimedia, IEEE Transactions on*, 1999, Т. 1, №. 1, pp. 65–76.

81. Horprasert T., Harwood D., Davis L. A statistical approach for real time robust background subtraction and shadow detection. IEEE Frame Rate Workshop, 1999.
82. Tian Y. L., Hampapur A. Robust salient motion detection with complex background for real-time video surveillance. Application of Computer Vision, 2005. WACV/MOTIONS'05 Volume 1. Seventh IEEE Workshops on, 2005, T. 2, pp. 30–35.
83. Adelson E.H., Bergen J.R. The plenoptic function and the elements of early vision in Landy M.S., Movshon J.A. Computational Models of Visual Processing. — Cambridge, Massachusetts: MIT Press, 1991, pp. 3–20.
84. Bradshaw K. J., Reid I. D., Murray D. W. The active recovery of 3d motion trajectories and their use in prediction. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1997, T. 19, №. 3, pp. 219-234.
85. Collins R., Tsin Y., Miller J.R., Lipton A. Using a DEM to determine geospatial object trajectories. in Proc. DARPA Image Understanding Workshop, Monterey, CA, Nov. 1998, pp. 115–122.
86. Stein G., Romano R., Lee L. Monitoring activities from multiple video streams - establishing a common coordinate frame. IEEE Trans. PAMI, vol. 22, Aug. 2000, pp. 758–767.
87. Khan S., Shah M. Consistent labeling of tracked objects in multiple cameras with overlapping fields of view. IEEE PAMI, 25(10), 2003, pp. 1355–1360.
88. Ng K. C. et al. An integrated surveillance system—human tracking and view synthesis using multiple omni-directional vision sensors. Image and Vision Computing, july 2004, vol. 22, issue 7, pp. 551–561.
89. Trivedi M., Huang K., Mikic L. Intelligent environments and active camera networks. Systems, Man, and Cybernetics, 2000 IEEE International Conference on. – IEEE, Oct. 2000, vol. 2, pp. 804–809.
90. Huang K.S., Trivedi M.M. Video arrays for real-time tracking of persons, head, and face in an intelligent room. Machine Vision Applications, 2003, vol. 14, №2, pp. 103–111.
91. Hutchinson S.A., Hager G.D., Corke P.I. A tutorial on visual servo control. IEEE Trans. Robotics and Automation, Oct. 1996, vol. 12, №5, pp. 651–670.
92. Collins R. T., Amidi O., Kanade T. An active camera system for acquiring multi-view video. Carnegie Mellon Univ, IEEE ICIP 2002.
93. Айзerman М.А., Броверман Э.М., Розоноэр Л.И. Метод потенциальных функций в теории обучения машин. —М.: «Наука», 1970.

94. Журавлëв Ю.И. Построение алгоритмов распознавания, корректных для заданной выборки. Журнал вычислительной математики и математической физики, том 19, №3, май-июнь 1979.
95. Журавлëв Ю.И. Об алгоритмах распознавания с представительными наборами (о логических алгоритмах). Журнал вычислительной математики и математической физики, том 42, №9, 2002, с. 1425–1435.
96. Лапко А.В., Лапко В.А., Соколов М.И., Ченцов С.В. Непараметрические системы классификации. —Новосибирск.: «Наука», 2000.
97. Горелик А.Л., Скрипкин В.А. Методы распознавания. —М.: «Высшая школа», 1977.
98. Айвазян С.А., Бухштабер В.М., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: классификация и снижение размерности, справочное издание. —М.: Финансы и статистика, 1989.
99. Haykin S. Nueral networks — a comprehensive foundation. —2nd Ed., Prentice-Hall of India Private Ltd., New Delhi, 1999.
100. Salvi J., Armanguè X., Batlle J. A comparative review of camera calibrating methods with accuracy evaluation. Pattern recognition, 2002, Т. 35, №. 7, pp. 1617–1635.
101. Zhang Z. A flexible new technique for camera calibration. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2000, Т. 22, №. 11, pp. 1330–1334.
102. http://docs.opencv.org/modules/calib3d/doc/camera_calibration_and_3d_reconstruction.html (дата обращения 17.04.2015.)
103. http://www.vision.caltech.edu/bouguetj/calib_doc/ (дата обращения 17.04.2015.)
104. Волосов Д. С. Фотографическая оптика. —М.: Искусство, 1978, 543 с.
105. Conrady A. Decentering lens systems. Monthly notices of the royal astronomical society of america, 1919, Т. 79, pp. 951–954.
106. Duane C. B. Close-range camera calibration. —Photogramm. Eng., 1971, Т. 37, pp. 855–866.
107. Mallon J., Whelan P. F. Precise radial un-distortion of images. Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, IEEE, 2004, Т. 1, pp. 18–21.
108. Levenberg K. A method for the solution of certain nonlinear problems in least squares. —Quart. Appl. Math, 1944, Т. 2, pp. 164–168.
109. Marquardt D. W. An algorithm for least-squares estimation of nonlinear parameters. Journal of the Society for Industrial & Applied Mathematics, 1963, Т. 11, № 2, pp. 431–441.

110. Morè J. J. The Levenberg-Marquardt algorithm: implementation and theory. Numerical analysis. Springer Berlin Heidelberg, 1978, pp. 105–116.
111. Aschwanden P., Guggenbuhl W. Experimental results from a comparative study on correlation-type registration algorithms. Robust computer vision, 1992, pp. 268–289.
112. Bhat D. N., Nayar S. K. Ordinal measures for image correspondence. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1998, T. 20, № 4, pp. 415–423.
113. Zabih R., Woodfill J. Non-parametric local transforms for computing visual correspondence. Computer Vision-ECCV'94. Springer Berlin Heidelberg, 1994, pp. 151–158.
114. Belhumeur P. N. A Bayesian approach to binocular stereopsis. International Journal of Computer Vision. 1996, T. 19, № 3, pp. 237–260.
115. Ohta Y., Kanade T. Stereo by intra-and inter-scanline search using dynamic programming. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1985, № 2, pp. 139–154.
116. Hirschmuller H. Stereo processing by semiglobal matching and mutual information. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2008, T. 30, № 2, pp. 328–341.
117. Boykov Y., Veksler O., Zabih R. Fast approximate energy minimization via graph cuts. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2001, T. 23, № 11, pp. 1222–1239.
118. Kolmogorov V., Zabih R. Computing visual correspondence with occlusions using graph cuts. Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, IEEE, 2001, T. 2, pp. 508–515.
119. Sun J., Zheng N. N., Shum H. Y. Stereo matching using belief propagation. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2003, T. 25, № 7, pp. 787–800.
120. Brown M. Z., Burschka D., Hager G. D. Advances in computational stereo. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2003, T. 25, № 8, pp. 993–1008.
121. http://www.cvlabs.net/datasets/kitti/eval_stereo_flow.php (дата обращения 15.05.2015.)
122. Everingham M. et al. The pascal visual object classes (voc) challenge. International journal of computer vision, 2010, T. 88, № 2, pp. 303–338.
123. Goyette N. et al. Changelogdetection. net: A new change detection benchmark dataset. Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on. IEEE, 2012, pp. 1–8.

124. Ferryman J., Ellis A. PETS2010: Dataset and challenge. Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on. IEEE, 2010, pp. 143–150.
125. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. —М.: Мир, 1979.
126. Naugle M.G. Illustrated TCP/IP. —Wiley computer publishing, 1999.
127. Новиков Ю.В., Кондратенко С.В. Локальные сети: архитектура, алгоритмы, проектирование. — М.: «Издательство ЭКОН», 2000.
128. Хамухин А.В. Алгоритм управления синхронизацией распределённых вычислений в системах видеонаблюдения. —М.: из-во «Компания Спутник+», научно-технический журнал «Техника и технология», №5, 2007, с. 31–33.
129. Хамухин А.В. Преобразование координат с изображения камеры на топографический план зоны обзора —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2014, № 1.
130. Хамухин А.В. Моделирование поверхности земли при преобразовании координат с изображения камеры на топографический план зоны обзора. —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2007, №5, с. 199–200.
131. Хамухин А.В. Вычисление реперной точки юстировки оптической оси видеокамеры по нескольким измерениям дальности до топографических объектов на местности. —М.: из-во «Компания Спутник+», научно-технический журнал «Аспирант и соискатель», 2007, №5, с. 161–162.
132. Хамухин А.В. Вычисление линии горизонта на изображении видеокамеры». — М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2007, №5, с. 197–198.
133. Калиткин Н.Н. Численные методы». —М.: «Наука», 1978.
134. Рябенький В.С. Введение в вычислительную математику. — М.: Физматлит, 2000.
135. Малистов А.С., Солохин А.А., Хамухин А.В. Калибровка поворотных камер в системах видеонаблюдения с компьютерным зрением. —Избранные труды XVII Международной интернет-конференции молодых ученых, аспирантов и студентов по современным проблемам машиноведения, Москва, 2005, с. 159–162.
136. Хамухин А.В. Алгоритм управления положением поворотной видеокамеры при смещении центра изображения на заданный вектор». —М.: из-во «Компания Спутник+», научно-технический журнал «Аспирант и соискатель», 2007, №5, с. 159–160.

137. Кудрявцев Л.Д. Курс математического анализа (в двух томах). —М.: «Высшая школа», 1981.
138. Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. — М.: Мир, 1989, 486 с.
139. Хамухин А.В. Измерение зависимости координат увеличения и фокусировки от кратности увеличения для объективов с координатным управлением. —М.: из-во «Компания Спутник+», научно-технический журнал «Аспирант и соискатель», 2007, №5, с. 163–164.
140. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ, под ред. А. Шеня. —М.: МЦНМО: БИНОМ. Лаборатория знаний, 2004, 960 с.
141. Малистов А.С., Солохин А.А., Хамухин А.В. Сложение за целями в мультисенсорных системах видеонаблюдения с компьютерным зрением. Труды XLVIII научной конференции МФТИ «Современные проблемы фундаментальных и прикладных наук», Москва, 2005.
142. Качановский Ю.П., Явтухович А.Г. Алгоритм работы распределенной системы распознавания автомобильных номеров на КПП. «Системы управления и информационные технологии», 2007, №1.1(27), с. 160–162.
143. Хамухин А.В. Вычисление угла упреждения при управлении наведением поворотных видеокамер на движущуюся цель. —М.: из-во «Компания Спутник+», научно-технический журнал «Техника и технология», 2007, №5, с. 34–35.
144. Бирюков С.И. Оптимизация. —М.: МФТИ, 1995.
145. Хамухин А.В. Критерий идентичности объектов на изображениях видеокамер с общей зоной обзора. —М.: из-во «Компания Спутник+», научно-технический журнал «Техника и технология», 2007, №5, с. 36–37.
146. Рихтер Д. Windows для профессионалов: создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows. —СПб: Питер, 2001.
147. Малистов А.С., Солохин А.А., Хамухин А.В. Методы оценки эффективности алгоритмов в интеллектуальных системах видеонаблюдения. Труды XLVII научной конференции МФТИ «Современные проблемы фундаментальных и прикладных наук», 2004, ч. 3, с. 216–217.
148. Малистов А.С., Солохин А.А., Хамухин А.В. Оценка эффективности алгоритмов в системах видеонаблюдения. XVI Международная Интернет-конференция молодых ученых, аспирантов и студентов по современным проблемам машино-ведения, тезисы докладов, из-во ИМАШ РАН, Москва, 2004, с. 177.

149. Хамухин А.В. Классификация объектов с двумя признаками на основе триангуляции. —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2007, №5, с. 81–82.
150. Margolin R., Zelnik-Manor L., Tal A. How to evaluate foreground maps. Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. IEEE, 2014, pp. 248–255.
151. Fornaciari M., Cucchiara R., Prati A. A mobile vision system for fast and accurate ellipse detection. Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on. IEEE, 2013, pp. 52–53.
152. Заде Л. А. Понятие лингвистической переменной и его применение к принятию приближенных решений, 1976.
153. Хамухин А.В. Анализ современного состояния разработок детекторов изменений изображений видеокамеры. —М.: из-во «Компания Спутник+» — научно-технический журнал «Актуальные проблемы современной науки», 2015, №1.
154. Barnich O., Van Droogenbroeck M. ViBe: A universal background subtraction algorithm for video sequences. Image Processing, IEEE Transactions on, 2011, Т. 20, №. 6, pp. 1709-1724.
155. Maddalena L., Petrosino A. The SOBS algorithm: what are the limits? IEEE Workshop on Change Detection, 2012.
156. Hofmann.M. Background segmentation with feedback: The pixelbased adaptive segmenter. IEEE Workshop on Change Detection, 2012.
157. Elgammal A., Duraiswami R., Harwood D., Davis L. Background and foreground modeling using nonparametric kernel density for visual surveillance. Proc. IEEE, 90:1151–1163, 2002.
158. Yuen J. et al. Labelme video: Building a video database with human annotations. Computer Vision, 2009 IEEE 12th International Conference on. IEEE, 2009, pp. 1451-1458.
159. Хамухин А.В. Разработка и анализ эффективных алгоритмов для достоверной идентификации номерных знаков транспортных средств. —М.: ЗАО НТЦ «Информтехника», журнал «Оборонная техника», №8–9, 2012.
160. LeCun Y. et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 1998, Т. 86, №. 11, pp. 2278–2324.
161. Viola P., Johnes M. Robust real-time object detection. International Journal of Computer Vision, 2001.
162. Bertozzi M., Broggi A., Del Rose M., Felisa M., Rakotomamonjy A., Suard. F. A pedestrian detector using Histograms of Oriented Gradients and a Support Vector

Machine Classifier. Intelligent Transportation Systems Conference, 2007, ITSC 2007, IEEE.

163. ГОСТ Р 50577-93. Знаки государственные регистрационные транспортных средств, типы и основные размеры. Технические требования. —Введ. 01.01.94. — М.: Технический комитет по стандартизации ТК 278, 2001.(Государственный стандарт Российской Федерации).
164. Хамухин А.В. Оценка объёма тестовой выборки для вычисления ожидаемой доли ошибки при классификации объектов видеонаблюдения. —М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки», 2007, №6, с. 169–170.
165. Хамухин А.В. Архитектура информационных систем хранения видеоданных для разработки алгоритмов компьютерного зрения. —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2013, № 6.
166. Хамухин А.В. Система тестирования алгоритмов компьютерного зрения. — М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки», 2014, № 1.
167. Хамухин А.В. Анализ методов тестирования алгоритмов автоматической обработки изображений —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2014, № 1.
168. Gropp W. et al. A high-performance, portable implementation of the MPI message passing interface standard. Parallel computing, 1996, Т. 22, №. 6, pp. 789–828.
169. Smart J.F. Jenkins: the definitive guide. — "O'Reilly Media, Inc.", 2011.
170. Thain D., Tannenbaum T., Livny M. Distributed computing in practice: The Condor experience. Concurrency-Practice and Experience, 2005, Т. 17, №. 2-4, pp. 323–356.
171. Amdahl G. M. Validity of the Single Processor Approach to Achieving. — Large Scale Computing Capabilities AFIPS Conf Proc, 1967.
172. Хамухин А.В. Алгоритмы калибровки видеокамер для фотометрической оценки размеров объектов. —М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2013, № 6.
173. Harris C., Stephens M. A combined corner and edge detector. Alvey vision conference, 1988, pp. 147–151.
174. Gabow H. N. Data structures for weighted matching and nearest common ancestors with linking. — University of Colorado, Boulder, Department of Computer Science, 1990, pp. 434-443.

175. Подсчёт посетителей, URL: <http://подсчёт-посетителей.рф> (дата обращения: 25.10.2015.)
176. Хамухин А.В. Алгоритм синхронизации видеопотоков в стереосистемах. — М.: из-во «Компания Спутник+», научно-технический журнал «Естественные и технические науки, 2013, № 6.
177. Edmonds J. Paths, trees, and flowers. Canadian Journal of mathematics, 1965, Т. 17, № 3, pp. 449-467.
178. Хамухин А.В. Обнаружение движения объектов компьютерными информационными видеосистемами с помощью двухмерных и стереоскопических характеристик изображения. —М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки, 2015, № 1.
179. Stenger B., Woodley T., Cipolla R. Learning to track with multiple observers, CVPR, 2009, pp. 2647–2654.
180. Comaniciu D., Ramesh V., Meer P. Real-time tracking of non-rigid objects using mean shift. Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. IEEE, 2000, Т. 2, pp 142-149.
181. Kass M., Witkin A., Terzopoulos D. Snakes: Active contour models. International journal of computer vision, 1988, Т. 1, № 4, pp 321-331.
182. Kölsch M., Turk M. Fast 2D hand tracking with flocks of features and multi-cue integration. — Workshop on Real-Time Vision for HCI, Washington DC, July 2004.
183. Stenger B., Woodley T., Kim T.-K., Hernandez C., Cipolla R. AIDIA — adaptive interface for display interaction. —In Proc. BMVC, October 2008.
184. Birchfield S., Tomasi C. A pixel dissimilarity measure that is insensitive to image sampling. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1998, Т. 20, №4, pp. 401–406.
185. Stiching pipeline,
URL: <http://docs.opencv.org/2.4/modules/stitching/doc/introduction.html> (дата обращения: 29.10.2015.)
186. Lowe D. G. Distinctive image features from scale-invariant keypoints. International journal of computer vision, 2004, Т. 60, № 2, pp. 91–110.
187. Brown M., Lowe D. G. Automatic panoramic image stitching using invariant features. International journal of computer vision, 2007, Т. 74, № 1, pp. 59–73.
188. Muja M., Lowe D. G. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. VISAPP (1), 2009, Т. 2.

189. Fischler M. A., Bolles R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, 1981, Т. 24, № 6, pp. 381–395.
190. Lindeberg T. Feature detection with automatic scale selection. International journal of computer vision, 1998, Т. 30, №. 2, pp. 79–116.
191. Koenderink J. J. The structure of images. Biological cybernetics, 1984, Т. 50, №. 5, pp. 363–370.
192. Babaud J. et al. Uniqueness of the Gaussian kernel for scale-space filtering. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 1986, №. 1, pp. 26–33.
193. Mikolajczyk K., Schmid C. An affine invariant interest point detector. Computer Vision-ECCV 2002. Springer Berlin Heidelberg, 2002, pp. 128–142.
194. Картапев В.Г. Основы теории дискретных сигналов и цифровых фильтров: Учеб. пособие для вузов. — М.: Высш. школа, 1982, 109 с.
195. Deriche R. Separable recursive filtering for efficient multi-scale edge detection. Proc. Int. Workshop on Machine Vision and Machine Intelligence, 1987, pp. 18–23.
196. Van Vliet L. J., Young I. T., Verbeek P. W. Recursive Gaussian derivative filters. Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on. IEEE, 1998, Т. 1, pp. 509–514.
197. Rublee E. et al. ORB: an efficient alternative to SIFT or SURF. Computer Vision (ICCV), 2011 IEEE International Conference on. IEEE, 2011, pp. 2564–2571.
198. Rosten E., Porter R., Drummond T. Faster and better: A machine learning approach to corner detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2010, Т. 32, №. 1, pp. 105–119.
199. Calonder M. et al. Brief: Binary robust independent elementary features. Computer Vision-ECCV, 2010, pp. 778–792.
200. Xu G., Terai J., Shum H. Y. A linear algorithm for camera self-calibration, motion and structure recovery for multi-planar scenes from two perspective images. Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on. IEEE, 2000, Т. 2, pp. 474–479.
201. Хамухин А.В. Разработка и анализ высокоэффективных способов и алгоритмов автоматического сопряжения, синхронизации, юстировки изображений, управления поворотными камерами и обработки информации в приборах и системах видеонаблюдения. —Кандидатская диссертация, Москва, 2008, 150 С.
202. Воеводин В. В. Параллельные вычисления. — БХВ-Петербург, 2004.
203. Хамухин А.В.,Петрикович Я.Я. Особенности преобразования сопряжения координат панорамного изображения с координатами плана местности. — М.: из-во

«Компания Спутник+», научно-технический журнал «Естественные и технические науки», 2014, №11–12.

204. Хамухин А.В. Анализ ресурсных затрат при эксплуатации автоматизированных систем видеонаблюдения. — М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки», 2013, №6.
205. Хамухин А.В. Расчёт ресурсных затрат при применении автоматизированных систем видеонаблюдения. — М.: из-во «Компания Спутник+», научно-технический журнал «Актуальные проблемы современной науки», 2014, №4.
206. Петрикович Я.Я., Кан И.А., Сомиков В.П., Лядвинский К.В., Лунин К.В., Хамухин А.В., Малистов А.С., Солохин А.А., Ахриев А.Х., Горбачев Е.В., Мурга С.Л., Болтнев А.А. Устройство автоматизированного контроля обстановки в зрительных залах. Патент РФ на полезную модель №47546, бюл. №24, 2005.
207. Петрикович Я.Я. Охрана периметра и городских объектов: шаг в будущее. Журнал «CCTV», сентябрь 2004, с. 29–31.
208. Петрикович Я.Я., Сомиков В.П., Солохин А.А., Малистов А.С., Хамухин А.В. Интеллектуальная система телевизионного наблюдения с компьютерным зрением “Orwell 2K”. Журнал «Техника средств связи», серия «Техника телевидения», научно-технический сборник, из-во ЗАО «МНИТИ», 2005, вып. 1, с. 54–57.
209. Ахриев А.Х., Малистов А.С., Хамухин А.В., Александров П.А. Комплексный подход к созданию систем автоматического видеонаблюдения и видеоконтроля на объектах высокой сложности типа ITER и атомных станций. «Вопросы атомной науки и техники», серия «Термоядерный синтез», 2006, вып. 3., с. 69–81.
210. Щавелев А.В., Малистов А.В., Гришин А.И. Система видеонаблюдения, регистрации и распознавания объектов и ситуаций “«Orwell2k»”. XXII Межведомственная Научно-техническая Конференция «Проблемы обеспечения эффективности и устойчивости функционирования сложных технических систем», сборник трудов, ч. 3, Серпухов, 2003, с. 343–345.
211. Малистов А.С., Солохин А.А., Хамухин А.В. Формальный подход к оценке качества алгоритмов обработки изображений в интеллектуальных системах видеонаблюдения. Журнал «Вопросы радиоэлектроники», серия «общетехническая(ОТ)», выпуск 2, ОАО ЦНИИ «Электроника», 2006.
212. Иванченко С.Т., Кан И.А., Лунин К.В., Малистов А.С., Петрикович Я.Я., Солохин А.А., Сомиков В.П., Хамухин А.В. Система обеспечения безопасности и мониторинга мобильных объектов. Патент РФ на полезную модель №36315, бюл. №7, 2004.

213. Иванченко С.Т., Кан И.А., Лунин К.В., Малистов А.С., Петричкович Я.Я., Солохин А.А., Сомиков В.П., Хамухин А.В. Система обеспечения безопасности и мониторинга мобильных объектов. Патент РФ №2265 531, бюл. №34, 2005.
214. Кан И.А., Лунин К.В., Малистов А.С., Петричкович Я.Я., Солохин А.А., Сомиков В.П., Хамухин А.В. Система и способ автоматизированного видеонаблюдения и распознавания объектов и ситуаций. Патент РФ на полезную модель №36912, бюл. №9, 2004.
215. Кан И.А., Лунин К.В., Малистов А.С., Петричкович Я.Я., Солохин А.А., Сомиков В.П., Хамухин А.В. Система и способ автоматизированного видеонаблюдения и распознавания объектов и ситуаций. Патент РФ №2268497, бюл. №02, 2006.

На правах рукописи

ПРИЛОЖЕНИЕ

Хамухин Анатолий Владимирович

**Высокоэффективные алгоритмы семантической обработки
видеоизображений и управления приборными комплексами
технического зрения**

05.13.01 – «Системный анализ, управление и обработка информации»
(информационно-вычислительное обеспечение)

Том 2
ДИССЕРТАЦИЯ
на соискание учёной степени
доктора технических наук

Научный консультант,
доктор технических наук,
профессор Петрикович Я.Я.

Москва — 2016

Содержание

Приложение 6. Свидетельства о регистрации интеллектуальной собственности	4
Приложение 7. Протоколы испытаний	10
Протокол испытания точности наведения поворотных видеокамер на подвижные объекты	10
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Курской АЭС	13
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Бурейской ГЭС совместно с ОАО «РусГидро»	19
Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» в губе Териберская совместно с ООО «Питер Газ»	22
Протокол испытания приборов подсчёта посетителей ВАРШ.463135.002-01 в Международном аэропорту «Шереметьево»	26
Протокол испытания системы технического зрения «Orwell2k-IP» в Международном аэропорту «Шереметьево»	27
Приложение 8. Акты внедрения алгоритмов семантической обработки видеоизображений и управления приборными комплексами технического зрения в серийном производстве, разработке и эксплуатации	32
Акт АО «ЭЛВИС Неотек» о внедрении алгоритмов управления видеоприборами, синхронизации и семантического сжатия в системе «Orwell2k» .	33
Акт АО «ЭЛВИС Неотек» о внедрении алгоритмов семантической обработки стерео- и моноизображений в устройствах «Statistics»(ВАРШ.468196.001) и «Statistics-3d»(ВАРШ.463135.002-01) .	35
Акт ОАО НПЦ «ЭЛВИС» о внедрении результатов диссертации в процесс проектирования архитектуры микросхем	36
Акт системного интегратора «ИнфинитиРус» об использовании результатов диссертации в устройствах ВАРШ.468196.001 и ВАРШ.463135.002-01 .	38
Акт системного интегратора «Лилана» об использовании результатов диссертации в комплексах технического зрения, предназначенных для охраны сооружений Вазузского гидроузла	39

Акт ООО «АДС» о внедрении описанных в диссертации алгоритмов в системы безопасности транспортной инфраструктуры олимпийских объектов Сочи	41
Приложение 9. Перечень патентов, относящихся к системам синхронной семантической обработки изображений	43
Приложение 10. Участие в выставках приборов, систем и технологий безопасности	44

Приложение 1. Свидетельства о регистрации интеллектуальной собственности



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012660849

«Программный комплекс «Orwell 2k-touch»

Правообладатель(ли): **Закрытое акционерное общество
«ЭЛВИС-НеоТек» (ЗАО «ЭЛВИС-НеоТек») (RU)**

Автор(ы): (*см. на обороте*)

Заявка № 2012618571

Дата поступления **9 октября 2012 г.**

Зарегистрировано в Реестре программ для ЭВМ
29 ноября 2012 г.

Руководитель Федеральной службы
по интеллектуальной собственности

Б.П. Симонов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012619211

«Программный комплекс «Statistics»

Правообладатель(ли): *Закрытое акционерное общество
«ЭЛВИС-НеоТек» (ЗАО «ЭЛВИС-НеоТек») (RU)*

Автор(ы): *Ерепилов Антон Михайлович,
Хамухин Андрей Владимирович, Баскаков Андрей Евгеньевич,
Ерёменко Сергей Викторович, Галкин Василий Витальевич,
Мятина Валентина Михайловна, Малистов Алексей Сергеевич,
Хамухин Анатолий Владимирович (RU)*

Заявка № 2012617247

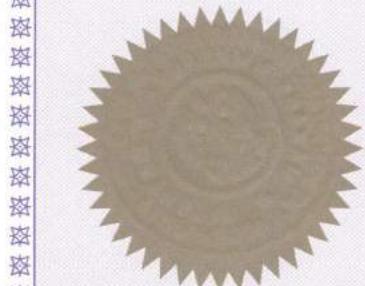
Дата поступления 27 августа 2012 г.

Зарегистрировано в Реестре программ для ЭВМ
12 октября 2012 г.

Руководитель Федеральной службы
по интеллектуальной собственности

A handwritten signature in black ink, appearing to read "Реесиц" followed by "Б.П. Симонов".

Б.П. Симонов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012619212

«Программный комплекс «Tra Vio SDK»

Правообладатель(ли): *Закрытое акционерное общество
«ЭЛВИС-НеоТек» (ЗАО «ЭЛВИС-НеоТек») (RU)*

Автор(ы): *Хамухин Анатолий Владимирович,
Малистов Алексей Сергеевич (RU)*

Заявка № 2012617248

Дата поступления 27 августа 2012 г.

Зарегистрировано в Реестре программ для ЭВМ

12 октября 2012 г.

Руководитель Федеральной службы
по интеллектуальной собственности

A handwritten signature in black ink, appearing to read "Б.П. Симонов".

Б.П. Симонов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2014613905

**Автоматическая система фотовидеофиксации нарушений
правил дорожного движения "Перекрёсток"**

Правообладатель: *Закрытое акционерное общество
«ЭЛВИС-НеоТек» (ЗАО «ЭЛВИС-НеоТек») (RU)*

Авторы: *см. на обороте*

Заявка № **2014611462**

Дата поступления **25 февраля 2014 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **10 апреля 2014 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

A handwritten signature in black ink, appearing to read "Б.П. Симонов".

Б.П. Симонов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2012615283

«Программный комплекс «Orwell 2k-base»

Правообладатель(ли): *Закрытое акционерное общество
«ЭЛВИС-НеоТек» (RU)*

Автор(ы): *(см. на обороте)*

Заявка № **2012613106**

Дата поступления **19 апреля 2012 г.**

Зарегистрировано в Реестре программ для ЭВМ
13 июня 2012 г.

Руководитель Федеральной службы
по интеллектуальной собственности

A handwritten signature in black ink, appearing to read "Б.П. Симонов".

Б.П. Симонов

Приложение 2. Протоколы испытаний

Протокол испытания точности наведения поворотных видеокамер на подвижные объекты

„УТВЕРЖДАЮ”
начальник КНТЛ-4

Сомиков В.П.
«28» 02 2006г.

Протокол испытаний №3

Точности наведения поворотных видеокамер на неподвижные объекты.

Дата проверки: 27 февраля 2006 года.

1. Цель испытаний.

Определение погрешности наведения оптической оси поворотной видеокамеры «Panasonic WV-CS860» на неподвижные объекты.

2. Объект испытаний.

Комплект оборудования, аналогичный системе «СПВ ОВТ – МАД», РАЯЖ 46652.001-ОС.ПЗ.

3. Метод испытаний.

Для оценки погрешности наведения проводится следующий ряд действий. Для работы с программным обеспечением см. документацию системы РАЯЖ 46652.001-ОС.ПЗ.

1. В программе настройки *CamsCalibrator.exe* выбирается сопряжённая пара видеокамер мастер и слейв.

2. На изображении неподвижной камеры выбирается точка, которая хорошо различима.

3. Посредством программы настройки в выбранную точку мастер-камеры наводится слейв-камера при помощи настроенного преобразования сопряжения. Значения координат углов азимута α^n и склонения β^n , отображаемые в программе настройки, записываются в таблицу вида 1. В данном случае номер измерения обозначается как n .

4. С помощью ручного управления программы настройки сопряжения производится точное наведение в необходимую точку: выбранная точка должна находиться ровно в центре изображения слейв-камеры. Считываются угловые координаты α_0^n и β_0^n , соответствующие точному наведению на точку, и затем записываются в ту же строчку таблицы 1, в которую производилась запись в предыдущем пункте.

5. Вычисляются отклонения действительных значений от расчётных,

$$\Delta\alpha^n = \alpha^n - \alpha_0^n, \quad \Delta\beta^n = \beta^n - \beta_0^n,$$

и записываются в таблицу 1 для измерения n .

6. Проводится N различных измерений, где N не менее 20.

7. Итоговые погрешности наведения Δ_α и Δ_β равны $3\sigma_\alpha$ по азимуту и $3\sigma_\beta$ по склонению, соответственно, то есть

$$\Delta_\alpha = 3\sigma_\alpha = 3 \cdot \sqrt{\frac{1}{N-1} \sum_{n=1}^N (\Delta\alpha^n)^2}, \quad \Delta_\beta = 3\sigma_\beta = 3 \cdot \sqrt{\frac{1}{N-1} \sum_{n=1}^N (\Delta\beta^n)^2}.$$

Общую норму отклонения обоих углов при наведении можно оценить как $\|\Delta\vec{r}\| = \sqrt{(\Delta_\alpha^2 + \Delta_\beta^2)/2}$, где $\vec{r} = (\alpha, \beta)$ — вектор углов наведения. Конкретные значения погрешностей приводятся в следующем пункте.

n	α	β	α_0	β_0	$\Delta\alpha$	$\Delta\beta$
1	276,8	84,8	276,9	84,75	-0,1	0,05
2	269,6	72,15	269,6	72,25	0,0	-0,1
3	293,0	80,1	293,0	80,2	0,0	-0,1
4	295,7	71,4	295,7	71,45	0,0	-0,05
5	284,5	86,75	284,2	86,55	0,3	0,2
6	287,8	86,4	287,5	86,35	0,3	0,05
7	293,6	87,4	293,6	87,15	0,0	0,25
8	270,2	87,25	270,4	87,25	-0,2	0,0
9	278,8	89,0	278,9	88,95	-0,1	0,05
10	292,6	91,4	292,5	91,15	0,1	0,25
11	279,7	93,15	279,6	92,95	0,1	0,2
12	272,5	93,65	272,6	93,55	-0,1	0,1
13	292,9	84,6	292,8	84,7	0,1	-0,1
14	272,9	81,8	273,1	81,9	-0,2	-0,1
15	292,6	94,25	292,4	94,0	0,2	0,25
16	288,0	73,6	288,1	74,05	-0,1	-0,45
17	267,8	88,85	268,0	88,75	-0,2	0,1
18	275,1	88,7	275,2	88,6	-0,1	0,1
19	284,4	90,2	284,3	89,85	0,1	0,35
20	288,0	93,25	287,9	92,95	0,1	0,3
21	270,9	90,7	271,1	90,55	-0,2	0,15
22	287,4	74,85	287,4	75,15	0,0	-0,3

Таблица 1. Значения измеренных углов при оценке точности наведения. n — порядковый номер измерения, α и β — экспериментальное значение азимута и склонения, α_0 и β_0 — точное значение азимута и склонения, $\Delta\alpha$ и $\Delta\beta$ — отклонение по азимуту и по склонению. Величины углов приведены в градусах.

8. После настройки сопряжения пары мастер и слейв, запустить сервер системы. Необходимо при этом установить такие параметры сервера, чтобы слейв-камера наводилась с одним и тем же углом зрения на все объекты.

9. Оставить сервер в режиме автоматического наведения на любые движущиеся объекты, которые выделяют алгоритмы обработки видеопотока от мастер-камеры. Необходимо, чтобы слейв-камера совершила S попыток наведения на движущиеся цели, где $S = 1000$.

10. Запустить терминал системы и войти в режим вывода списков объектов, которые система выделила на мастер-камере, и у которых есть снимки слейв-камеры.

11. Среди S изображений объектов от поворотной камеры, полученных в результате наведения, выделить S_α изображений, в которые объект не попал из-за слишком большой погрешности наведения по азимуту, и S_β изображений, в которые объект не попал из-за слишком большой погрешности наведения по склонению.

12. Оценить горизонтальный угол зрения α слейв камеры, при котором она наводилась на объекты. Поскольку координата увеличения слейв-камеры в данном эксперименте зафиксирована, и известна, то для получения угла зрения необходимо либо воспользоваться техническим паспортом камеры, в котором указана зависимость угла зрения от координаты увеличения объектива камеры, либо необходимо оценить эту зависимость с помощью лабораторных измерений.

13. Вычислить вертикальный угол зрения слейв-камеры β , который удовлетворяет со-

отношению

$$\beta = 2 \arctg\left(\frac{3}{4} \tg \frac{\alpha}{2}\right).$$

Значение угла приведено ниже в результатах испытаний, см. таблицу 2.

14. Вычислить оценку погрешности по азимуту и склонению как, соответственно,

$$\Delta_\alpha = 3 \cdot \frac{\alpha/2}{F^{-1}(1 - S_\alpha/2S)} \text{ и } \Delta_\beta = 3 \cdot \frac{\beta/2}{F^{-1}(1 - S_\beta/2S)},$$

где F^{-1} функция ошибок, обратная к функции стандартного нормального одномерного распределения. Результат показан в таблице 2.

4. Результат испытаний.

В результате испытаний получаем таблицу 1 измеренных значений отклонений. По результатам расчётов составлена таблица 2:

Параметр испытаний	Значение
погрешность Δ_α	0,45°
погрешность Δ_β	0,55°
$\ \Delta\vec{r}\ $	0,5°
Объём статистики наведения	1000 объектов
Количество промахов наведения	8 промахов
Угол зрения камеры при наведении	5°
Погрешность наведения на подвижные объекты	3°

Таблица 2. Результаты испытаний

5. Выводы

Погрешность наведения 0,5° позволяет рассчитывать на попадание в поле зрения слейва объектов, если его угол зрения больше 1°. Поскольку даже при максимальном фокусном расстоянии 86мм слейва «Panasonic WV-CS860» его угол зрения заведомо больше 3°, то при найденной величине погрешности невозможна ошибка наведения на неподвижные объекты, из-за которой объекта не будет в поле зрения поворотной видеокамеры. При наведении на подвижные объекты погрешность равна 3°, и при фокусном расстоянии, соответствующем углу объектива 5°, вероятность промаха около 1%.

ИСПЫТАНИЯ ПРОВОДИЛИ:

ведущий инженер

Хамухин А.В.

ведущий инженер

Малистов А.С.

ведущий инженер

Лядвинский К.В.

Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Курской АЭС

Утверждено
Генеральный директор ЗАО «ЭЛВИИС»

Андрей С. Т. Иванченко
«_____» 2011 г.

Утверждаю

Курской АЭС
А. А. Иванов
«29» 2011 г.

Протокол испытаний станции радиолокационной "ORWEI-В"

г. Курчатов

«26» април 2011 г.

1 Общие положения

1.1 Испытания проводятся из соображений

- Решения по пункту 2 протокола совещания начальников службы безопасности ОАО «Концерн Росэнергоатом» от 15 апреля 2011 г.
 - Письма Курской АЭС исх. №116-33/8535 от 30.05.2011г
 - «Программы и методик проведения испытаний станции радиолокационной ORWELL-R совместно с поворотной телекамерой» (далее – Программа).

1.2 Заказчик испытаний: Курская АЭС (далее – Заказчик), исполнитель – ЗАО «ЭЛВИИС» (далее – Исполнитель).

2 Объект, цель и задачи испытаний

2.1 Объектом испытаний является станция радиолокационная ORWELL-R ЛЦКБ.464412.001 (далее – Изделие) производства ЗАО «ЭЛВИИС» совместно с купольной поворотной телекамерой «Apix-36ZDome/D1 EXT» и тепловизионной камерой «Термофрейм-М-03161-Т» на поворотной платформе.

2.2 Целью испытаний является подтверждение тактико-технических характеристик Изделия в условиях реального облака Китайской АЭС.

2.3.3. Задачи на выявление и уточнение

- определение соответствия функциональных характеристик Изделия заявленным в эксплуатационной документации;
 - исследование границ применимости Изделия в части тактико-технических характеристик, не внесенных в эксплуатационную документацию Изделия;
 - определение целесообразности использования Изделия на объекте в реальных условиях эксплуатации.

3 Оцениваемые показатели и расчетные соотношения

3.1 Выполнение требований к реализуемым функциям Изделия:

- обнаружение движущейся цели;
- автоматическое определение дальности и азимута обнаруженных целей;
- автоматическое измерение радиальной скорости цели;
- автосопровождение обнаруженных целей;
- выдачу сигналов тревоги по факту обнаружения цели внутри тревожной зоны;
- отображение радиолокационной карты охраняемого объекта с возможностью привязки графического плана объекта к радиолокационной карте;
- отображение мнемоник обнаруженных целей на графическом плане объекта;
- отображение траекторий движения обнаруженных целей;
- ведение архива функционирования (системные события, радиолокационная карта и целевая обстановка), возможность чтения и визуализации архивных данных средствами АРМ;
- индикацию текущего состояния;
- автоматическое наведение поворотной телекамеры и тепловизора на обнаруженную цель и отображение видеоизображения цели на экране АРМ;
- возможность ручного и полуавтоматического (по карте) управления поворотной телекамерой;
- видеозапись сопровождения цели с телекамеры и тепловизора.

3.2 Измерение максимальной и минимальной дальности обнаружения выполняется с помощью о типовой цели (человек, надувная лодка), перемещающейся внутри арены обзора в направлении от внешнего модуля Изделия или к внешнему модулю Изделия. Радиальная скорость перемещения цели должна составлять не менее 1 км/ч.

3.3 Оцениваемые показатели дальности цели для режима 1(ЛЧМ):

- максимальная дальность отметки цели составляет не менее 950 м;
- минимальная дальность отметки цели составляет не более 150 м.

Оцениваемые показатели дальности цели для режима 2(КИ):

- максимальная дальность отметки цели составляет не менее 450 м;
- минимальная дальность отметки цели составляет не более 50 м;

4 Материально-техническое обеспечение испытаний

4.1 Изделие размещалось на крыше здания береговая насосная станция №3 на высоте 12м от уровня земли на дамбе водоема Курской АЭС, АРМ оператора размещался в помещении часового КПП БНС-3. Полигоном испытания являлась территория дамбы и водоем.

4.2 Испытаниям подвергается комплект Изделия в составе, приведенном в таблице 1.

Таблица 1 – Состав испытательного комплекта

Обозначение	Наименование	Количество	Примечание
ЛЦКБ.464412.001	Антенна	1	
ЛЦКБ.685631.001	Станция радиолокационная Orwell-R	1	
ЛЦКБ.685664.001	Сборка проводная питания	1	
	Сборка кабельная передачи данных	1	
	Сервер	1	
	Компьютер АРМ	1	
ЛЦКБ.465275.001-01	Модуль коммутационный Orwell-R Adapter	1	Подключение РЛС и видеокамеры
ЛЦКБ.465275.001-03	Модуль коммутационный Orwell-R Adapter	1	Подключение тепловизора
ЛЦКБ.467145.001	Ключ защиты ElKey	1	
Apix-36ZDome/D1 EXT	Купольная телекамера	1	
Термофрейм-М-03161-Т	Тепловизор на поворотной платформе PTH355P	1	

4.3 На испытаниях использовалось устройство антенное (Orwell-R Antenna 16, ширина угломестной диаграммы направленности 16 градусов), которое при высоте установки 12 метров обеспечивала зону действия РЛС в диапазоне 42...∞ метров без изменения угла наклона антенного устройства.

4.4 Типовая цель: человек, надувная лодка с человеком на веслах (предоставлена Заказчиком).

5 Условия проведения испытаний

5.1 Даты проведения испытаний – 23.08.2011-25.08.2011.

5.2 Место проведения испытаний – БНС-3 Курская АЭС.

5.3 Температура воздуха: 10-22 °C.

5.4 Ветер: 2 -10 м/с.

5.5 Осадки: временами дождь.

6 Результаты испытаний

Перечень проверок и результаты испытаний приведены в таблице 2.

Таблица 2 – Результаты испытаний

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Проверка выполнения требований к реализуемым функциям	5.1	требованиям удовлетворяет	требованиям удовлетворяет	Журнал испытаний п. 1
Измерение максимальной и минимальной дальности обнаружения и сопровождения для типовой цели	5.2	Режим ЛЧМ: мин. 150, макс. 950. Режим КИ: мин. 50, макс. 450	Режим ЛЧМ: мин. 158, макс. 403. Режим КИ: мин. 46, макс. 383	Журнал испытаний п.2. Испытания проводились в рамках ограниченной прямой видимости
Измерение максимальной и минимальной дальности обнаружения и сопровождения для разных типов целей(надувная лодка)	5.3	Режим ЛЧМ: мин. 150, макс. 950. Режим КИ: мин. 50, макс. 450	Режим ЛЧМ: мин. -, макс. 1003. Режим КИ: мин. 73,7, макс. -	Журнал испытаний п.3. Испытания проводились в рамках ограничений объекта
Проверка диапазона дальности визуальной фиксации объекта оператором по изображению поворотной телекамеры	5.4			Журнал испытаний п.4
Проверка максимальной дальности цели, при которой проводилось автоматическое наведение телекамеры			274 метра	В рамках зоны калибровки

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Процент ошибочных (неточных) наведений от общего числа наведений за время нахождения цели в арене обзора				
Проверка точности измерения дальности	5.5			Не измерялся
Проверка требований отказоустойчивости оборудования и программного обеспечения	5.6	Сбоев не зафиксировано	Сбоев не зафиксировано	Не проводилась
Проверка выполнения требований к реализуемым функциям в нормальных климатических условиях в течение 24 часов	5.7	требованиям удовлетворяет	требованиям удовлетворяет	Журнал испытаний п.6
				Журнал испытаний п.7

7 Выводы

7.1 Функциональные характеристики Изделия соответствуют заявленным в эксплуатационной документации и техническим условиям.

7.2 Изделие может быть применимо для обнаружения нарушителей как в зоне акватории БНС-3, так и на подступах к БНС-3 на территории дамбы, находящихся в прямой видимости на расстоянии от 50м от места установки Изделия.

7.3 Изделие не обеспечивает обнаружение нарушителей за формами рельефа, которые препятствуют прямой видимости: стены строений, деревья, кустарники.

7.4 Изделие подтвердило свои функциональные характеристики в реальных условиях эксплуатации.

8 Замечания и рекомендации

8.1 Технические характеристики купольной телекамеры Apix не позволяли обеспечить ее наведение на цели с требуемой точностью в ближней зоне действия Изделия: на расстоянии до 200 метров цели не всегда попадали в зону обзора телекамеры при ее автоматическом наведении.

8.2 В дополнении к испытаниям в соответствии с Программой были проведены испытания работы Изделия при перемещении человека на границы ближней зоны действия Изделия: человек двигался от Изделия с расстояния 40м до 50м. В ходе такой проверки человек обнаруживался в требуемом секторе обзора 7 раз из 11 проходов. Так как данная зона должна находиться под охраной, то рекомендуется дополнить охрану зоны посредством системы видеонаблюдения

Orwell2k-IP, работа которой интегрирована с работой Изделия. От исполнителя требуется предоставить информацию по системе Orwell 2k-IP и предложение по техническим средствам (видеокамерам или тепловизионным камерам) обеспечивающим выполнение данной задачи.

Альтернативой использованию дополнительного обсрудования для видеонаблюдения может быть перенос места установки Изделия на расстояние не менее 70м от границы водосброса.

8.3 Для использования изделия на Курской АЭС необходимо:

- сертификация изделия по системе ОИТ;
- получение решения ГКРЧ о выделении полосы радиочастот;
- разрешение Россвязьнадзора на использование радиочастот или радиоканалов;
- свидетельство Россвязьнадзора о регистрации радиоэлектронного средства.

9 Приложения

9.1 «Программа и методика проведения испытаний станции радиолокационной ORWELL-R совместно с поворотной телекамерой» - 1экз. на 10 листах

9.2 «Журнал испытаний станции радиолокационной ORWELL-R в комплекте с купольной видеокамерой и тепловизором на поворотной платформе» - 1экз. на 26 листах

От Курской АЭС

Начальник СБ: Тонконоженко Н.Н.

Начальник ИТО СБ: Гетманский И.Н.

Инженер по ИТСО СБ: Шмалиук С.Ю.

От ЗАО «ЭЛВИИС»

Инженер НТЛ -22: Мурга С.Л.

Инженер НТЛ -21: Квасов А.В.

Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» на Бурейской ГЭС совместно с ОАО «РусГидро»

Утверждаю
Директор по безопасности Филиала ОАО
“РусГидро”-“Бурейская ГЭС”

Г.Ф.Епишкін
2010

Протокол испытаний станции радиолокационной “ORWELL-R”

« 15 » декабря 2010 года

1 Общие положения

1.1 Испытания проводятся на основании:

- договора №_09.11.10(1)/Д(1526-10/ДО)
- «Программы и методики проведения испытаний станции радиолокационной ORWELL-R совместно с поворотной телекамерой» (далее – Программа).

2 Объект, цель и задачи испытаний

2.1 Объектом испытаний является станция радиолокационная ORWELL-R ЛЦКБ.464412.001 (далее – Изделие) производства ЗАО «ЭЛВИИС» совместно с купольной поворотной телекамерой Panasonic WV-CW960.

2.2 Состав испытуемого комплекта, а также цель и задачи испытаний приведены в Программе.

3 Оцениваемые показатели и расчетные соотношения

3.1 Пункты и методы проверок описаны в Программе.

4 Материально-техническое обеспечение испытаний:

4.1 Испытательный полигон, в том числе помещения для размещения внутренней аппаратуры Изделия (предоставляется Заказчиком).

4.2 Изделие, укомплектованное в соответствии с п. 2.2 Программы.

4.3 Типовая цель: человек или автомобиль (предоставляется Заказчиком).

5 Условия проведения испытаний:

- | | |
|----------------------------------|-------------------------------|
| 5.1 Даты проведения испытаний – | 10 декабря-15 декабря 2010 г. |
| 5.2 Место проведения испытаний – | п. Талакан, Бурейская ГЭС. |
| 5.3 Температура воздуха: | от -20 до -40 град. цельсия. |

5.4 Ветер: 2 – 5 м/с
 5.5 Осадки: снег
 5.6 Особые условия: высота установки Изделия 115 м.

6 Результаты испытаний

Перечень проверок и результаты испытаний приведены в таблице 2.

Таблица 2 – Результаты испытаний РЛС

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Проверка выполнения требований к реализуемым функциям	5.1	Соотв.	Соотв.	
Измерение параметров сектора обзора (отклонение измеренной ширины сектора от заданной)	5.2	5,0°	1°	
Измерение максимальной дальности обнаружения и сопровождения	5.3	1000 м.	1203 м.	
Проверка минимальной дальности обнаружения		500 м.	493 м.	
Проверка диапазона дальности визуальной фиксации объекта оператором по изображению поворотной телекамеры	5.4			
Проверка максимальной дальности цели, при которой проводилось автоматическое наведение телекамеры		1000 м.	1235 м.	
Проверка минимальной дальности цели, при которой проводилось автоматическое наведение телекамеры		500 м.	499 м.	
Проверка способов управления телекамерой	5.5	Соотв.		
Измерение времени реакции поворотной камеры на действия оператора		2 с.	1 с.	
Измерение времени переключения режимов с автоматического на полуавтоматический или ручной и обратно		2 с.	1с.	

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Проверка точности измерения дальности	5.6	10 м.	9 м.	
Проверка требований отказоустойчивости оборудования и программного обеспечения	5.7	Соотв.	Соотв., РЛС функционировала в течение 3 суток	

7 Выводы

7.1 Функциональные характеристики Изделия соответствуют заявленным в эксплуатационной документации и технических условиях.

7.2 Изделие может выполнять свои функции в реальных условиях.

8 Замечания и рекомендации

От филиал РусГидро Бурейская ГЭС
Старший Мастер участка СБ и КД
Габдрахманов В.Г.

От ООО "СТАЛТ ЛТД"
Главный инженер проекта
Данилов А.С.

От ЗАО «ЭЛВИИС»
Инженер
Машкин А.А.

Инженер
Мурга С.Л.

Протокол испытания алгоритмов наведения поворотных видеокамер на объекты в системе «Orwell2k-R» в губе Териберская совместно с ООО «Питер Газ»

Протокол испытаний станции радиолокационной “ORWELL-R”

«19 октября 2010 года

1 Общие положения

- 1.1 Испытания проводятся на основании:
- Договора

– «Программы и методики проведения испытаний станции радиолокационной ORWELL-R совместно с поворотной телекамерой в рамках специальных изысканий для комплекса инженерно-технических средств охраны по морскому порту и портовому транспортно-технологическому комплексу» (далее – Программа).

2 Объект, цель и задачи испытаний

2.1 Объектом испытаний является станция радиолокационная ORWELL-R ЛЦКБ.464412.001 (далее – Изделие) производства ЗАО «ЭЛВИИС» совместно с купольной поворотной телекамерой Panasonic WV-CW960.

2.2 Состав испытуемого комплекта, а также цель и задачи испытаний приведены в Программе.

3 Оцениваемые показатели и расчетные соотношения

3.1 Пункты и методы проверок описаны в Программе.

4 Материально-техническое обеспечение испытаний:

4.1 Испытательный полигон, в том числе помещения для размещения внутренней аппаратуры Изделия (предоставляется Заказчиком).

4.2 Изделие, укомплектованное в соответствии с п. 2.2 Программы.

4.3 Типовая цель: судно (предоставляется Заказчиком).

5 Условия проведения испытаний:

6.1 Даты проведения испытаний – 17–19 октября 2010 г.

6.2 Место проведения испытаний – полигон в губе Териберская.

6.3 Температура воздуха: от минус 1 до +6 °C.

6.4 Северный, до 30 м/с.

6.5 Осадки: дождь, мокрый снег, снежные и дождевые заряды.

6 Результаты испытаний

Перечень проверок и результаты испытаний приведены в таблице 2.

Таблица 2 – Результаты испытаний РЛС

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Проверка выполнения требований к реализуемым функциям	5.1	Соотв.	Соотв.	
Измерение параметров сектора обзора (отклонение измеренной ширины сектора от заданной)	5.2	5,0°	5,1°	
Измерение максимальной и минимальной дальности обнаружения и сопровождения	5.3			
Проверка минимальной дальности обнаружения (человек)		50 м	45,4 м	
Проверка минимальной дальности обнаружения (судно типа МРБ)		50 м	117,4 м	Возможность измерить минимальную дальность ограничена способностью судна приблизиться к берегу
Проверка максимальной дальности обнаружения (судно типа МРБ)		950 м	1 144,8 м	
Проверка максимальной дальности обнаружения (моторная лодка)		950 м	959,6 м	
Проверка диапазона дальности визуальной фиксации объекта оператором по изображению поворотной телекамеры	5.4			
Проверка максимальной дальности цели, при которой проводилось автоматическое наведение телекамеры		Не задано	831 м	При необходимости может быть увеличена.

Наименование проверки	Номер пункта методов контроля	Требуемое значение	Фактическое значение	Примечание
Процент ошибочных (неточных) наведений от общего числа наведений за время нахождения цели в арене обзора		Не задано	Одно неточное наведение из 14	Время жизни цели в арене обзора 71 с.
Проверка способов управления телекамерой	5.5	Соотв.	Соотв.	
Измерение времени реакции поворотной камеры на действия оператора		2 с	1 с	
Измерение времени переключения режимов с автоматического на полуавтоматический или ручной и обратно		2 с	1 с	
Проверка точности измерения дальности	5.6	6 м	6 м	
Проверка требований отказоустойчивости оборудования и программного обеспечения	5.7	Соотв.	Соотв.	За время испытаний сбоев оборудования или программного обеспечения не зафиксировано
Проверка выполнения требований к реализуемым функциям в нормальных климатических условиях в течение 24 часов	5.8	Соотв.	Соотв.	
Измерение потребляемой мощности	5.9	Не задано	Не более 700 Вт	Согласно паспортным данным на составные части

7 Выводы

7.1 Функциональные характеристики Изделия соответствуют заявленным в эксплуатационной документации и технических условиях.

7.2 Проблем электромагнитной совместимости Изделия с РЛС «Нева-Б», установленной на том же испытательном полигоне (расстояние между позициями 6 м, сектора обзора совпадают).

7.3 Изделие может выполнять свои функции в реальных условиях акватории губы Териберская.

8 Замечания и рекомендации

8.1 С учетом климатических условий региона рекомендуется в качестве поворотной камеры (камер) для фиксации цели использовать тепловизионную камеру с фокусным расстоянием, соответствующим требованиям к распознаванию цели.

8.2 Рекомендуется разработать механизм экспорта в файл (файлы) и последующего синхронного проигрывания нескольких потоков данных (радиолокационные и видеоданные).

От ООО «Питер Газ»

От ЗАО «Эскорт-Центр»

От ЗАО «ЭЛВИИС»

Начальник лаборатории радиолокационных систем охраны

О. Н. Зинченко

Инженер эксплуатационной лаборатории

С. Л. Мурга

**Протокол испытания приборов подсчёта посетителей ВАРШ.463135.002-01
в Международном аэропорту «Шереметьево»**

**Отчет о тестировании
видеосчетчика посетителей CountBOX SmartDevice 3D в
АО «Международный Аэропорт Шереметьево»**

Тестирование устройства проводилось с целью определения погрешности счетчиков посетителей относительно условий размещения устройств и внешних факторов освещенности. С целью анализа точности подсчета проходов посетителей были рассмотрены следующие схемы освещенности перехода:

- день и яркое солнце;
- день и облачно;
- вечер и искусственное освещение;
- вечер и естественное освещение.

В качестве эталонных значений для каждой из рассматриваемых схем использовались записи с камеры внутреннего видеонаблюдения, охватывающей участок контролируемой зоны под местами установки видеосчетчиков CountBOX SmartDevice 3D.

В качестве временного отрезка для анализа каждой из схем освещенности были выбраны 10-ти минутные интервалы.

Таблица 1. Результаты тестирования видеосчетчиков CountBOX SmartDevice 3D (после отладки системы).

№ п/п	Схема освещенности	Дата и время выборки	Зафиксировано счетчиком посетителей	Посетителей по данным с камеры видеонаблюдения	Точность подсчета посетителей (%)
1.	День и яркое солнце	16.10.2015 14:13-14:23	266	274	97,0
		16.10.2015 14:50-16:00	152	154	98,7
2.	День и облачно	15.10.2015 15:10-15:20	308	313	98,4
		16.10.2015 10:10-10:20	156	166	94,0
3.	Вечер и искусственное освещение	15.10.2015 22:00-22:10	65	65	100,0
		15.10.2015 20:50-21:00	92	97	94,8
		15.10.2015 18:30-18:40	143	144	99,3
4.	Вечер и естественное освещение	17.10.2015 16:10-16:20	221	223	99,1
		16.10.2015 16:05-16:17	293	295	99,3

Заместитель Генерального директора
по информационным технологиям
АО «МАШ»

И.Н. Хохлов

Исп. Рубцов Д.Е.
Тел.8(926)244-45-93.

Протокол испытания системы технического зрения «Orwell2k-IP» в Международном аэропорту «Шереметьево»

УТВЕРЖДАЮ
Заместитель генерального
директора ОАО «МАШ»
по безопасности
Л. П. Любезный
«_____» 2010г.



УТВЕРЖДАЮ
Исполнительный директор
ЗАО «Евразия Телеком Регионы»
А. П. Фирсов
2010г.



УТВЕРЖДАЮ
Генеральный директор
ГУП НПЦ «ЭЛВИС»
Я. Я. Петричкович
«_____» 2010г.



Протокол испытаний системы теленаблюдения с аналитическими функциями «Orwell2k-IP» на терминале «Е» ОАО «МАШ».

1. Объект испытаний

Объектом испытания является Система видеонаблюдения с компьютерным зрением «Orwell 2k-IP», установленная на терминале «Е» аэропорта «Шереметьево».

2. Цель испытаний

Проверка соответствия тактических, эксплуатационных и технических характеристик Системы видеонаблюдения с компьютерным зрением «Orwell 2k-IP» требованиям ОАО «МАШ» в реальных условиях эксплуатации.

3. Место проведения испытаний

Испытания проводились на объекте в реальных условиях эксплуатации системы.

4. Тестируемые характеристики системы

№№ п/п	Вид испытаний (проверок)	Номера пунктов методики испытаний	Примечание
1.	Оценка соответствия комплектности системы реальным	6.1	

№№ п/п	Вид испытаний (проверок)	Номера пунктов методики испытаний	Примечание
	условиям эксплуатации.		
2.	Проверка работоспособности системы при разных внешних условиях	6.2	
3.	Проверка соответствия поля наблюдения камер с планом местности.	6.3	
4.	Проверка функции видеоархива и сохранения информации о событиях на наблюдаемом объекте.	6.4	
5.	Проверка автономного функционирования системы.	6.5	
6.	Проверка режима отображения информации о наблюдаемом объекте.	6.6	
7.	Проверка режимов слежения поворотной камеры.	6.7	
8.	Проверка контроля за состоянием оборудования.	6.8	
9.	Проверка времени между появлением объекта и поступлением уведомления оператору.	6.9	
10.	Проверка управления настройками тревожных ситуаций.	6.10	
11.	Проверка функций администрирования системы.	6.11	
12.	Проверка обнаружения остановившихся целей и оставленных предметов.	6.12	
13.	Проверка обнаружения целей в зоне интереса	6.13	

5. Результаты испытаний

- 5.1. Комплектность соответствует заявленной в методике испытаний.
- 5.2. Проверки обнаружения оставленного предмета и проникновения в охраняемую зону не выявили отклонений в качестве работы системы «Orwell2k-IP» в зависимости от внешних условий.
- 5.3. Объекты, обнаруженные системой, позиционируются в реальном времени на графический план контролируемой территории и обозначаются мнемознаками.
- 5.4. Проверки, проведённые в области видимости камер согласно пунктам методики 6.9, 6.12, 6.13, зафиксированы в видеоархиве. Время поиска тревожного события по заявленным в программе испытаний параметрам не превышает 3с. Результаты поиска представляются в удобном для оператора виде, позволяющем оперативно получить требуемую видеинформацию как с обзорных, так и с поворотных камер.
- 5.5. При возникновении тревожной ситуации в видеоквадраторе появляется камера, которая зафиксировала событие, в списке тревог отображается комментарий с сообщением или инструкциями для оператора, а также проигрывается видеозапись произошедшего тревожного события.
- 5.6. В системе присутствует план охраняемого объекта с возможностью отображения зон наблюдения и функциями отображения мнемознаков людей и других целей
- 5.7. Все заявленные в методике испытаний режимы управления (ручной, автоматический, полуавтоматическое наведение по карте) функционируют в соответствии с требованиями методики.
- 5.8. Неисправности во время испытания системы не зарегистрированы. Соответствующие уведомления об отключении питания камеры, отключения питания ip-камер и переход в режим электропитания от источника бесперебойного питания(ИБП) отображаются в окне телеметрии. При обесточивании серверов системы ИБП обеспечивает электропитание испытываемой системы в течение требуемого времени, при этом функционирование системы не прерывается.
- 5.9. Время с момента возникновения тревожного события, указанного в п. 6.10 методики испытаний, до вывода сообщения оператору не превышает 1 секунду.
- 5.10. Система «Orwell2k-IP» позволяет устанавливать тревожные датчики с ограничениями по месту события, его времени, и его типу (оставленный предмет, обычный объект). Анализ архива тревожных событий подтверждает корректную работу тревожных датчиков с ограничениями.
- 5.11. Система «Orwell2k-IP» позволяет ограничивать функции оператора при работе с клиентским приложением в соответствии с п. 6.11 методики испытаний.

5.12. При проведении тестов, указанных в п. 6.12 методики, система обнаружила все оставленные предметы, не зависимо от времени суток и условий испытаний.

5.13. При проведении тестов, указанных в п. 6.13 методики, система обнаружила все проникновения в охраняемую зону, не зависимо от времени суток и условий испытаний. В архиве системы не зафиксировано значительного количества ложных срабатываний.

Результаты испытаний показали, что система видеонаблюдения с компьютерным зрением «Orwell 2k-IP», установленная на терминале «Е» аэропорта «Шереметьево», соответствует предъявляемым к ней требованиям и готова к промышленной эксплуатации на объекте.

От ОАО «МАШ»

Председатель комиссии:



А.Ю. Федяев

Заместитель начальника службы – начальник отдела технических систем авиационной безопасности Службы обеспечения и контроля авиационной безопасности Дирекции по авиационной безопасности

Члены комиссии:



К.В. Куликов

Начальник отдела эксплуатации информационных систем Дирекции по информационным технологиям



Н.В. Никифоров

Ведущий инженер отдела технических систем авиационной безопасности Службы обеспечения и контроля авиационной безопасности Дирекции по авиационной безопасности



С.С. Китюков

Инженер группы санкционированного доступа и телеконтроля отдела обеспечения авиационной безопасности в Терминале Ш-2 Службы обеспечения и контроля авиационной безопасности Дирекции по авиационной безопасности

Инженер службы обеспечения и контроля
авиационной безопасности Дирекции по
авиационной безопасности

~~И.А. Иволгин~~
Инженер службы обеспечения и контроля
авиационной безопасности Дирекции по
авиационной безопасности
От ЗАО «Евразия Телеком Регионы»

Менеджер проектов

 Н.А. Иволгин

 С.Н. Гуторов

 С.Л. Воробьев

От ГУП НПЦ «ЭЛВИС»:

Начальник НТЛ-52

 Солохин А.А.

Приложение 3. Акты внедрения алгоритмов семантической обработки видеоизображений и управления приборными комплексами технического зрения в серийном производстве, разработке и эксплуатации

№ _____ от « _____ » 2016г.
27.10.16(2) ИПН от 27.10.2016



АКТ

о внедрении результатов докторской диссертационной работы
Хамухина Анатолия Владимировича
«Высокоэффективные алгоритмы семантической обработки видеоизображений и
управления приборными комплексами технического зрения»
по специальности 05.13.01 — «Системный анализ, управление и обработка информации
(информационно-вычислительное обеспечение)»

Комиссия в составе: председатель комиссии — директор департамента технической поддержки Солохин А.А., советник генерального директора, д.т.н. Янакова Е.С., директор департамента разработки программного обеспечения Миллер С.Ю., составили настоящий акт о том, что результаты, полученные Хамухиным А.В. и изложенные в его диссертационной работе, используются при серийном производстве электронных приборных комплексов технического зрения на предприятии АО «ЭЛВИС Неотек». К таким результатам относятся:

- 1) математические соотношения для расчёта параметров сопряжения неподвижных камер между собой и с планом местности, а также неподвижных и поворотных видеокамер, позволяющих реализовать эффективные алгоритмы автоматического сопровождения объектов поворотными видеокамерами;
- 2) новые алгоритмы синхронизации данных между сопряжёнными распределёнными модулями обработки изображений от поворотных и неподвижных видеокамер;
- 3) способы автоматизации настройки параметров сопряжения модулей обработки изображений, что существенно снизило трудоёмкость настройки систем;
- 4) создание алгоритмов семантической обработки информации, использующих модули обработки изображений;
- 5) создание алгоритмов построения панорамы;
- 6) разработка способов верификации и тестирования эффективности алгоритмов компьютерного зрения.

На основе вышеперечисленных результатов созданы и производятся серийно целый ряд разработанных на предприятии приборных комплексов:

- «Система видеонаблюдения с компьютерным зрением Orwell 2k» ВАРШ.00001-01;
- «Программный комплекс «Тепловизионный локатор кругового обзора Филин» ВАРШ.00003-01;
- радарный приборный комплекс «Orwell-R» ЛЦКБ.464412.002 с поворотными видеокамерами;
- система контроля дорожного движения ВАРШ.466452.002 «Перекресток»;
- система контроля соблюдения правил парковки ВАРШ.466452.001 «ПаркИнспектор»;

Внедрение результатов, разработанных Хамухиным А.В., позволило:

- уменьшить в среднем в 5 раз количество видеокамер, необходимых при использовании программно-аппаратных комплексов, разработанных в компании АО «ЭЛВИС-Неотек», для обеспечения требуемых разрешений изображений за счёт применения поворотных видеокамер;
- повысить достоверность автоматической идентификации определённых типов изображений, уменьшив вероятность ложной идентификации в 10–20 раз при неизменной вероятности ошибки второго рода, за счёт применения способов тестирования и верификации алгоритмов семантической сегментации;
- уменьшить примерно в 10 раз трудозатраты специалистов по контролю обстановки в поле зрения видеокамер в условиях, когда семантическая обработка изображений применима, за счёт применения сопряжения, автоматизации и классификации информации как основы для улучшения эргономических характеристик рабочих мест операторов и упрощения работы по наблюдению за обстановкой в поле зрения видеокамер;
- уменьшить требования к сложности элементной вычислительной базы перечисленных выше приборов и комплексов за счёт применения быстродействующих алгоритмов управления видеокамерами и семантической обработки изображений.

Данные новые технические решения нашли своё отражение в патентах «Система обеспечения безопасности и мониторинга мобильных объектов» (патент РФ №2265531), «Система и способ автоматизированного видеонаблюдения и распознавания объектов и ситуаций» (патент РФ №2268497), «Устройство автоматизированного контроля обстановки в зрительных залах» (патент РФ №2296434), «Система обеспечения безопасности и мониторинга мобильных объектов» (патент РФ на полезную модель №36315), «Способ и система определения скорости транспортных средств» (патент РФ №2015102323), «Способ и система калибровки комплекса измерения скорости транспортных средств» (патент РФ №2015102320).

Результаты, описанные в диссертации Хамухина А.В., лежат в основе вышеупомянутых разработанных в нашей организации электронных приборных комплексах технического зрения, которые пользуются спросом в Российской Федерации и за рубежом, поскольку характеристики данных комплексов находятся на уровне мировых аналогов, а в части достоверности детектирования событий с помощью семантической обработки изображений эти комплексы превосходят мировой уровень. За счёт пониженных требований к вычислительной элементной базе стоимость их внедрения с использованием алгоритмов Хамухина А.В. ниже, чем стоимость внедрения аналогичных систем других производителей, что способствует проведению политики импортозамещения в области программного обеспечения, предназначенного для обработки изображений и управления видеокамерами.

ПРЕДСЕДАТЕЛЬ КОМИССИИ:

директор департамента технической поддержки



Солохин А.А.

ЧЛЕНЫ КОМИССИИ:

советник генерального директора, д.т.н.



Янакова Е.С.

директор департамента разработки

программного обеспечения



Миллер С.Ю.

№ от « 21 » 10 2016г.
21.10.16/11/8771

«УТВЕРЖДАЮ»

генеральный директор
АО «ЭЛВИС Неотек»
Поминов И.А.

АКТ

о внедрении результатов исследований, проведённых Хамухиным Анатолием
Владимировичем в докторской диссертационной работе
«Высокоэффективные алгоритмы семантической обработки видеоизображений и
управления приборными комплексами технического зрения»

Комиссия в составе: председатель комиссии — директор департамента по разработке аппаратной платформы Гусев В.В., заместитель начальника отдела видеоаналитики, к.т.н., Малистов А.С., руководитель отдела функциональной верификации и тестирования СБИС Лоторев В.Ю. составили настоящий акт о том, что результаты диссертационной работы Хамухина А.В. по созданию алгоритмов семантической обработки стереоизображений, а также способов верификации и тестирования эффективности алгоритмов компьютерного зрения, внедрены при разработке оптических счетчиков посетителей ВАРШ.468196.001 «Statistics» («CountBox») и счётчиков ВАРШ.463135.002-01 «Statistics-3d» («Countbox-3d»), построенных на принципах технического стереозрения.

Использование теоретических и практических достижений, полученных в диссертации Хамухина А.В. позволило улучшить качество подсчёта в перечисленных ранее устройствах до 98% за счёт использования разработанных в диссертации алгоритмов тестирования и верификации алгоритмов семантической сегментации, а также уменьшить сложность и объём элементной вычислительной базы перечисленных выше приборов, применяя быстродействующие алгоритмы семантической обработки изображений, в том числе от стереокамер, что, в свою очередь, снизило объём аппаратуры и энергопотребление.

Таким образом, результаты диссертации Хамухина А.В. являются базовыми для разработанных в нашей организации счетчиков, широко устанавливаемых и эксплуатируемых на многих объектах страны. Характеристики данных приборов находятся на уровне мировых аналогов, при этом в них используются вычислительные процессоры с пониженными требованиями к производительности и энергопотреблению, что позволяет существенно снизить стоимость производства подобных видеосистем и сделать их привлекательными для отечественных и иностранных потребителей, что способствует практической реализации политики импортозамещения.

ПРЕДСЕДАТЕЛЬ КОМИССИИ:

 Гусев В.В.

ЧЛЕНЫ КОМИССИИ:

 Малистов А.С.

 Лоторев В.Ю.

№ 26.10.16(1) / илр
от 26. 10. 2016г

"УТВЕРЖДАЮ"

Заместитель генерального директора



Кравченко П.С.

2016 г.

АКТ

использования результатов диссертационной работы Хамухина Анатолия Владимировича «Высокоэффективные алгоритмы семантической обработки видеоизображений и управления приборными комплексами технического зрения» на соискание учёной степени доктора технических наук по специальности 05.13.01.

Комиссия в составе:

председатель комиссии –

Беляев Андрей Александрович, учёный секретарь НТС, д.т.н.,

члены комиссии –

Путря Фёдор Михайлович, к.т.н., начальник лаборатории НТО-1,

Николаев Артём Валерьевич, к.т.н., старший научный сотрудник,

удостоверяет данным актом, что результаты диссертационной работы Хамухина А.В. использованы на нашем предприятии при проектировании вычислительных микросхем, предназначенных для обработки изображений и представляющих собой гетерогенные системы на кристалле. Непосредственное влияние на архитектуру блоков ядер DSP VIP, блоков быстрого расчёта глубины стереоизображений, а также на систему организации внутренней памяти и коммуникаций внутри систем на кристалле, которые проектирует ОАО НПЦ «ЭЛВИС», оказали следующие результаты, достигнутые в работе Хамухина А.В.

1. Приведённая в обзоре диссертационной работы классификация методов семантической обработки видеоизображений позволила выделить наиболее эффективные методы обработки изображений для поддержки аппаратной оптимизации анализа изображений в проектируемых на предприятии вычислительных системах на кристалле. К основным методам обработки изображений, проанализированных в обзорной части работы Хамухина А.В., относятся методы семантической классификации изображений с помощью нейронных сетей, методы сегментации подвижных и неподвижных объектов на изображении, методы сопоставления изображений, устойчивые к шуму.

2. Предложенные автором диссертации методы ускорения выделения подвижных объектов в поле зрения камеры и быстroredействующие методы анализа панорамных изображений учитывались при разработке системы команд DSP для ядра VIP и позволили оптимизировать характеристики энергопотребления новых вычислительных микросхем. Применение предложенных автором диссертации методов в комплексе с другими аппаратно-программными

решениями позволило существенно улучшить энергоэффективность разработанных микросхем и уменьшить энергозатраты в пересчете на каждый обрабатываемый пиксель, по предварительным данным, не менее чем в 3-5 раз по сравнению с наиболее современными процессорами Intel.

3. Система и методика тестирования программных приложений семантического анализа изображений, предложенная в диссертационной работе Хамухина А.В., используется для контроля отсутствия ошибок на этапе прототипирования программных моделей проектируемых на предприятиях вычислительных блоков в системах на кристалле, предназначенных для аппаратного ускорения алгоритмов обработки изображений.

Таким образом, результаты диссертационной работы Хамухина Анатолия Владимировича широко используются при проектировании, разработке и тестировании вычислительных платформ, таких, например, как SOLARIS и CAPRI. Характеристики этих процессоров во многом превосходят мировые аналоги, поэтому работа Хамухина А.В. способствует проведению политики импортозамещения, что определяет её существенный вклад в создание новых научно-обоснованных технических и технологических решений, внедрение которых вносит значительный вклад в развитие науки и укреплению экономики нашей страны.

Председатель комиссии:

учёный секретарь НТС, д.т.н.



Беляев А.А.

Члены комиссии:

начальник лаборатории НТО-1, к.т.н.,



Путря Ф.М.

старший научный сотрудник, к.т.н.



Николаев А.В.

Общество с ограниченной ответственностью

«ИнфинитиРус»



117545, г. Москва, ул.Подольских курсантов, д.3 стр.2

ИНН/КПП 7725706572/772501001

ОГРН 1107746874982, БИК: 044525593

Банк «АЛЬФА-БАНК» (АО), Москва

р/с: 40702810202430000127 к/с: 30101810200000000593

8 (495) 478-08-88, www.count-box.ru, ok@ctbx.ru

«УТВЕРЖДАЮ»

Генеральный директор
компании ООО «ИнфинитиРус»

Бугарь Наталья Викторовна
исх.№ 16101301
от "13" октября 2016г.

Акт внедрения

результатов работы Хамухина Анатолия Владимировича «Высокоэффективные алгоритмы семантической обработки видеоизображений и управления приборными комплексами технического зрения» на соискание учёной степени доктора технических наук по специальности 05.13.01 «Системный анализ, управление и обработка информации (информационно-вычислительное обеспечение)».

Настоящий акт составлен о том, что практические результаты, полученные в ходе диссертационной работы Хамухина А. В. в частях, касающихся систем подсчёта посетителей, внедрены в приборы «CountBox» (ВАРШ.468196.001) и «CountBox SmartDevice 3d» (ВАРШ.463135.002-01), которые используются в распределённых облачных системах, предназначенных для оценки эффективности и планирования развития розничных торговых сетей и других предприятий, занимающихся массовым предоставлением услуг.

Благодаря быстродействующим алгоритмам компьютерного зрения для подсчёта посетителей, которые предложены в диссертации Хамухина А. В., компактные устройства «CountBox» и «CountBox SmartDevice 3d» работают не менее эффективно, чем ряд зарубежных аналогов, достигая (при использовании обработки стереоизображений) точности подсчёта 98%, при этом их стоимость значительно ниже благодаря пониженным требованиям к вычислительной мощности компонент устройства.

Обозначенные преимущества позволили нашей компании выполнить крупные поставки облачных систем подсчёта посетителей во множество предприятий обслуживания и розничной торговли. Фактически, исходя в основном из экономических предпосылок, мы на практике последовали принятому в последнее время в Российской Федерации курсу импортозамещение иностранной продукции.

Генеральный директор

компании ООО «ИнфинитиРус»



Бугарь Наталья Викторовна



ООО «ЛИЛАНА» 111674, г. Москва, ул. 2-я Вольская, 20. тел. (499)211-35-40,
факс (495)746-79-40. E-mail: lilana-tv@mail.ru, WWW.LILANA.RU

Лицензии:

МЧС РФ ГУ ГПС 8-2/00264 от 01 июня 2009г.

Допуск СРО 0515.4-2010-7721001275-С-018 от 03.09.2013г.

Допуск СРО 0381-2010-7721001275-П-011 от 28.02.2012г.

Исх. № 2316 От 27 октября 2016 года.

«УТВЕРЖДАЮ»



Акт внедрения результатов

диссертационной работы Хамухина Анатолия Владимировича «Высокоэффективные алгоритмы семантической обработки видеоизображений и управления приборными комплексами технического зрения» на соискание степени доктора технических наук по специальности 05.13.01 «Системный анализ, управление и обработка информации (информационно-вычислительное обеспечение)».

Комиссия в составе:

- председатель комиссии Технический директор ООО «ЛИЛАНА» Яцков В.И.;
- член комиссии Главный инженер ООО «ЛИЛАНА» Чичканов А.А.;
- член комиссии Инженер систем безопасности ООО «ЛИЛАНА» Шкаликов Е.В.

составила настоящий акт о том, что результаты диссертационной работы Хамухина А. В., на которых основаны функциональные характеристики системы видеонаблюдения ВАРШ.00001-01 «Система видеонаблюдения с компьютерным зрением Orwell 2k» установлена нашей организацией для обеспечения комплексной безопасности особо важных объектов на Вазузском гидротехническом узле АО «Мосводоканал»

Комплекс «Orwell 2k» используется для защиты и ограничения доступа к сооружениям Вазузского гидроузла, обеспечивая их безопасную эксплуатацию. Комплекс поддерживает стандартные для систем видеонаблюдения функции вывода изображений видеокамер на

пользовательские терминалы в реальном времени и записи изображений в видеоархив с глубиной архива 1 месяц. Помимо стандартных функций, комплекс «Orwell 2k» реализует функции автоматического анализа видеопотока и выделяет без участия операторов важные с точки зрения обеспечения безопасности охраняемых объектов события, а также предоставляет детальные изображения происходящего после автоматического наведения поворотных видеокамер. Эти дополнительные функции реализованы на основе алгоритмов, представленных в диссертации Хамухина А.В..

Алгоритмы семантической обработки изображений, выделяющие в реальном времени потенциально опасные события, а именно — возгорание, оставленный без присмотра предмет, проникновение в запретные зоны и др., а также алгоритмы сопряжения изображений между собой и алгоритмы сопоставления места события с планом местности, описанные в диссертационной работе Хамухина А. В., позволяют эффективно применять сложные видеосистемы для охраны важных объектов инфраструктуры АО «Мосводоканал» при меньших трудозатратах со стороны персонала, ответственного за безопасность.

Председатель комиссии:

Технический директор ООО «ЛИЛАНА».

Яцков В.И.

Члены комиссии:

Главный инженер ООО «ЛИЛАНА».

Чичканов А.А.

Инженер систем безопасности ООО «ЛИЛАНА»

Шкаликов Е.В.



ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ

"АДС"

127018, г. Москва, ул. Сущевский Вал, дом 5, стр. 3, ком. 22. Телефон: +7(495) 665-51-65

Исх. № 44

"УТВЕРЖДАЮ"

генеральный директор ООО «АДС», к.т.н.

К. Н. Белоусов

2016 г.

А К Т



о внедрении результатов диссертационной работы Хамухина Анатолия Владимировича на тему: «Высокоэффективные алгоритмы семантической обработки видеоизображений и управления приборными комплексами технического зрения» на соискание ученой степени доктора технических наук по специальности 05.13.01 «Системный анализ, управление и обработка информации (информационно-вычислительное обеспечение)»

Комиссия в составе:

председатель комиссии – Скворцов Максим Сергеевич, директор дивизиона СМР;

члены комиссии – Кобзев Александр Андреевич, инженер 1-й категории
составила настоящий акт о том, что результаты диссертационной работы А.В. Хамухина, а именно созданные на основе полученных теоретических результатов модули сопряжения и управления поворотными видеокамерами, внедрены в радиоэлектронных приборных комплексах серии «Orwell-R», которые установлены и приняты на техническое обслуживание нашей организацией для обеспечения безопасности транспортной инфраструктуры олимпийских объектов в Сочи. Приборный комплекс «Orwell-R» обеспечивает автоматическое обнаружение и сопровождение объектов типа «человек», «машина», «группа людей» на дальностях до 3000 м в угловом секторе обзора 360 градусов в любое время суток, при любых погодных условиях.

Информационно-измерительная охранныя система «Orwell-R» состоит из датчиков, предназначенных для обнаружения целей возле контрольно-пропускных пунктов и на территориях, прилегающих к периметру охраняемых объектов как

внутри, так и снаружи. Как правило, комплект оборудования включает в себя приборный комплекс с активным датчиком «Orwell-R» ЛЦКБ.464412.002, модуль коммутационного адаптера «Orwell-R Adapter» ЛЦКБ.465275.001-13, а также управляемые поворотные устройства с видеокамерами и устройства сопряжения.

Задачами каждой подсистемы являются:

- а) автоматическое обнаружение движущихся целей на подступах к охраняемым объектам;
- б) отображение подробной информации о целевой обстановке на мониторе оператора, в том числе вывод изображений движущихся целей, полученных с помощью разработанных автором диссертации алгоритмов автоматического наведения установленных на поворотных устройствах видеокамер с управляемым увеличением. Наведение поворотных видеокамер на цели производится в течение 0,5сек после их обнаружения датчиками РЛС, тем самым обеспечивается высокая вероятность правильной идентификации целей операторами видеокомплекса, что является безусловным достоинством комплекса «Orwell-R».

Внедрение результатов диссертационной работы А.В. Хамухина, основанных на алгоритмах синхронной обработки сопряжённых видео и радиолокационных информационных сигналов, в виде программно-аппаратных комплексов сопряжения датчиков «Orwell-R» и поворотных видеокамер, позволило обеспечить комплексную безопасность на объекте и подступах к нему с вероятностью правильного обнаружения 0,99 и вероятностью ложной тревоги 10^{-6} .

Таким образом, результаты диссертационной работы А.В. Хамухина в виде алгоритмов и программ для сопряжения поворотных видеокамер с комплексами технического зрения, позволяют повысить эффективность использования системы «Orwell-R» и обеспечивать большую надёжность технических и организационных параметров по обеспечению безопасности стратегически важных объектов инфраструктуры.

Председатель комиссии:
директор дивизиона СМР

Члены комиссии:

инженер 1-й категории



Скворцов М.С.



Кобзев А.А

Приложение 4. Перечень патентов, относящихся к системам синхронной семантической обработки изображений

1. «Система обеспечения безопасности и мониторинга мобильных объектов» (патент РФ №2265531)
2. «Система и способ автоматизированного видеонаблюдения и распознавания объектов и ситуаций» (патент РФ №2268497)
3. «Устройство и способ автоматизированного контроля обстановки в зрительных залах» (патент РФ №2296434)
4. «Система и способ определения государственного регистрационного номера транспортного средства» (патент РФ №2014104763)
5. «Способ и система определения скорости транспортных средств» (патент РФ №2015102323)
6. «Способ и система калибровки комплекса измерения скорости транспортных средств» (патент РФ №2015102320)
7. «Система и способ автоматической фиксации несанкционированного прохода через зону контроля» (патент РФ №2014152194)
8. «Система обеспечения безопасности и мониторинга мобильных объектов» (патент РФ на полезную модель №36315)
9. «Система и способ автоматизированного видеонаблюдения и распознавания объектов и ситуаций» (патент РФ на полезную модель №36912)
10. «Устройство автоматизированного контроля обстановки в зрительных залах» (патент РФ на полезную модель №47546)
11. «Система определения государственного регистрационного номера транспортного средства» (патент РФ на полезную модель №2014104762)
12. «Система определения скорости транспортных средств» (патент РФ на полезную модель №2015102309)
13. «Система калибровки комплекса измерения скорости транспортных средств» (Патент РФ на полезную модель №2015102315)

Приложение 5. Участие в выставках приборов, систем и технологий безопасности

1. IX Московская Международная выставка «Охрана, безопасность и противопожарная защита» 2003, 2013, 2014.

Сертификат участника IX Международной выставки «Охрана, безопасность и противопожарная защита» mips2003,

Диплом за высокое качество и технический уровень аппаратно-программного комплекса «Orwell2k».

2. Международная выставка систем безопасности «Security Israel», Тель-Авив, Израиль (2003, 2005, 2013).

3. Выставка «Наука Москвы - вчера, сегодня, завтра» (03–06 октября 2003).

4. Выставка «Security Israel 2004» (Тель-Авив, 22–24 июня 2004).

5. II Международная выставка полицейской техники и вооружений «Чайна Полис-2004» в Пекине (23–26 июня 2004).

6. Выставка «Москва - город науки» (16–19 ноября 2004).

7. Международная выставка «Высокие технологии XXI века» — «ВТ XXI-2004».

8. VII Международный Авиационно-Космический Салон (МАКС-2005) (09 октября 2005).

9. Международная выставка «Высокие технологии XXI века» — «ВТ XXI-2005» (18–22 апреля 2005).

10. Международный военно-морской салон (29 июня–03 июля 2005), Санкт-Петербург.

11. Международная выставка «Высокие технологии XXI века» — «ВТ XXI-2006» (24–27 апреля 2006).

12. День инноваций Минобороны РФ 2013, 2014, 2015

13. Международная выставка «All-over-IP» 2011, 2012, 2013

14. Interpolitex 2012, 2013, 2014

15. Безопасность на дорогах ради безопасности жизни 2014

16. Армия 2015