

Федеральное государственное учреждение «Федеральный исследовательский центр «Информатика и управление» Российской академии наук»

На правах рукописи

Скорюкина Наталья Сергеевна

**Методы локализации и идентификации плоских
ригидных объектов на изображениях**

Специальность 2.3.1 —
«Системный анализ, управление и обработка информации, статистика»

Диссертация на соискание учёной степени
кандидата технических наук

Научный руководитель:
доктор технических наук
Арлазаров Владимир Викторович

Москва — 2024

Оглавление

Стр.

Введение	4
---------------------------	----------

Глава 1. Признаки изображений, их типы и методы детекции.	
Интерпретация и анализ признаков для локализации и классификации объектов на изображениях	
1.1 Введение: Особенности распознавания объектов на изображениях на сегодняшний день	12
1.2 Задача локализации и идентификации на изображениях	14
1.3 Типы признаков и методы их выделения	16
1.3.1 Границы, края	16
1.3.2 Прямые линии и сегменты	19
1.3.3 Точки схода и прямоугольники	23
1.3.4 Локальные признаки. Особые точки и дескрипторы	26
1.4 Сравнение представлений. Поиск ближайших соседей	34
1.5 Пространственная верификация сопоставлений	36
1.5.1 Модификации RANSAC с улучшениями отдельных этапов	38
1.5.2 Модификации RANSAC с обобщением улучшений	40
1.6 Иные методы сопоставления	41
1.7 Выводы. Постановка задачи	42
Глава 2. Выделение признаков с учётом структуры объекта и модели формирования изображения	
2.1 Формирование изображения. Камера обскура.	45
2.2 Методы выделения четырёхугольников	46
2.2.1 Поиск четырёхугольников в строгих ограничениях	47
2.2.2 Модификация 2.2.1 в более слабых ограничениях	53
2.2.3 Поиск четырёхугольников в неизвестных ограничениях . .	54
2.3 Методы выделения текстовых признаков	57
2.3.1 Поиск МЧЗ на видеокадрах	61
2.3.2 Поиск МЧЗ на произвольных изображениях	64
2.4 Эксперимент	68
2.4.1 Наборы данных	68
2.4.2 Поиск четырёхугольника	69

2.4.3	Поиск МЧЗ	70
2.5	Выводы по главе	71
Глава 3. Метод идентификации и локализации объекта на изображении		73
3.1	Параметры задачи	73
3.2	Модель изображения	74
3.3	Создание набора моделей	75
3.3.1	Создание шаблона	75
3.3.2	Создание поискового индекса	77
3.4	Анализ входного изображения	77
3.4.1	Выделение глобальных признаков	77
3.4.2	Выделение локальных признаков	78
3.5	Сопоставление изображений в модели	80
3.5.1	Отсечение и ранжирование классов по локальным признакам	80
3.5.2	Отсечение классов по структурным текстам	82
3.5.3	Геометрическая валидация	82
3.5.4	Реализация PESAC для проективного преобразования	89
3.5.5	Использование комбинированных входных данных	91
3.6	Эксперимент	92
3.6.1	Наборы данных	92
3.6.2	Оценка точности и скорости локализации и идентификации объектов	93
3.7	Выводы по главе	98
Заключение		100
Список литературы		102
Приложение А. Патенты и свидетельства		119
Приложение Б. Акты о внедрении		125

Введение

Распознавание изображений вместе с компьютерами прошло путь от специализированных научных и индустриальных систем к технологиям повседневным и общедоступным. Большой вклад в развитие релевантного научного направления внесли такие отечественных и зарубежных учёные, как Ю.И. Журавлев, В.Л. Арлазаров, Ю.В. Визильтер, Д.П. Николаев, В.А. Сойфер, О.А. Славин, И.Б. Гуревич, К.В. Воронцов, D.G. Lowe, R. Szeliski, P. Fua и другие.

Значимую долю объектов интереса составляют плоские или квази-плоские объекты – документы, регистрационные знаки транспорта, двумерные коды и графика, отдельные стороны-границы предметов. В некоторых вариантах задачи распознавания требуется определить точный тип объекта внутри общей категории, т.е. *идентифицировать* конкретный объект среди ему подобных. Например, на изображении из возможных документов – «паспорт РФ, дипломатический», а на снимке с экспозиции из всей представленной живописи – «Вокзал Сен-Лазар, номер W438». Внутри такого типа допускаются зоны изменяемых данных, в которых наблюдаемые особенности (*признаки*) могут варьироваться от экземпляра к экземпляру. Остальные, стабильные признаки образуют фиксированную структуру. *Локализация* объекта может быть и независимой задачей, и необходимым этапом для его дальнейшего качественного распознавания. Если считать эталоном объекта его фронтальное изображение в фиксированном разрешении без элементов фона, то определение параметров преобразования между образом и эталоном позволяют точно *локализовать* объект. Точность особенно значима при решении задач, связанных с анализом подлинности объектов и их образов, поскольку область объекта, сцена и граница между ними на изображении исследуются по отдельности. При «грубой» локализации как правило устанавливают только центр объекта и приблизительный размер, а значит граница не соответствует реальной.

С распространением мобильных устройств, которые объединили в себе возможности захвата, передачи и обработки изображения, возник естественный запрос выполнять съёмку и распознавание «на лету», отвязавшись от стационарных компьютеров и сканирующих устройств. Мобильные устройства как правило комплектуют малоформатными цифровыми камерами. Снимки с таких камер, полученные обычными пользователями, существенно отличаются и

от сканов, и от изображений со стационарных камер. Условия съёмки – произвольное освещение, ракурс и окружение – способствуют появлению таких дефектов, как блики, смазы и перепады яркости. Изображение может захватывать существенно большую область, чем занимает объект интереса, а значит – содержать элементы сцены. Поворот камеры относительно плоскости объекта порождает образ с искажением геометрии, в общем случае – проективным.

Достижение высокой точности распознавания объектов на снимках с маломатричных камер потребовало создания новых методов анализа изображений, устойчивых к перечисленным дефектам и особенностям. Проблемам распознавания образов на мобильных устройствах посвящены работы В. В. Арлазарова, Д.В. Полевого, К.Б. Булатова, Е.Е. Лимоновой, K. Mikolajczyk, V. Lepetit и других исследователей. На текущий момент международным сообществом проведена большая работа по адаптации существующих методов к характерным особенностям фото- и видеокадров, а также по созданию новых методов для их анализа и распознавания.

Есть два подхода к анализу изображений с помощью мобильных устройств. Первый заключается в передаче изображения на удалённый сервер для последующего распознавания. Исполнение на сервере не накладывает жёстких ограничений на вычислительную эффективность метода и позволяет работать с большими базами данных. Недостатками такого подхода является необходимость стабильного интернет-соединения и возможный расход средств на передачу данных. Помимо этого, изображения могут содержать приватные данные (изображения документов, удостоверяющих личность), а их передача и хранение в любой форме регулируется законодательно в РФ и других странах. Второй подход заключается в выполнении распознавания непосредственно на мобильном устройстве, что ограничивает допустимую сложность методов и размер используемых данных в сравнении с первым (серверным) подходом. Выбор подхода зависит от задачи, и в современных системах распознавания изображений представлены оба варианта.

Таким образом, помимо точности идентификации и локализации возникают требования к эффективности методов в отношении скорости и расхода памяти. Это необходимо для исполнения на большинстве пользовательских устройств, в т.ч. малопроизводительных, т.е. прямо связано с доступностью защиты информации. Законодательно регулируется порядок взаимодействия не только с информацией ограниченного доступа, но и с объектами авторского

права, и с культурными ценностями особого значения. Это осложняет формирование репрезентативных наборов данных. В ряде случаев на этапе создания решения доступно единственное изображение-образец при сотнях и тысячах возможных типов.

Методы сквозного обучения (end-to-end) сегодня позволяют получить достаточно точные решения. Ряд архитектур искусственных нейронных сетей подходит для исполнения на самых современных мобильных устройствах, однако для их обучения требуются репрезентативные данные. Модели, построенные с переносом обучения, характеризуются пониженной точностью и ориентированы к исполнению на GPU. Кроме того, дообучение необходимо при изменении набора типов (добавления новых и исключения неактуальных), что влечёт дополнительные затраты по времени. Это делает инструментарий сквозных нейронных сетей слабо применимым к решаемой задаче.

Другой известный подход, достигающий высокой точности - сравнение изображений как наборов признаков локальных окрестностей - *дескрипторов*, их координат и размеров. При построении набора можно выбросить зоны с вариативными данными и дефектами съёмки, что позволяет задать тип в таком представлении даже одним изображением-эталоном. За счёт геометрического согласования признаков с похожими дескрипторами можно исключить некоторые коллизии определения типа и локализовать объект. В процессе разбора подхода отмечено следующее: хотя и методы дескрипции, и методы согласования развиваются в контексте достижения большей точности, компактности и скорости, они абстрагированы от реальных задач и совместного использования. Поэтому подход может быть улучшен в целом по точности, скорости и памяти за счёт ограничений задачи.

Представление изображения в виде дескрипторов окрестностей само по себе позволяет оптимизировать вычисления и память при сопоставлении изображений. При этом в самом представлении, как правило, только один тип признаков. Это означает, что устойчивость к шуму и искажениям достигается за счёт дескрипторов с большей информативной ёмкостью (что отражается в т.ч. на увеличении размера дескриптора в битах). Процесс геометрического согласования признаков со своей стороны должен исключить не только параметры, нереализуемые камерой, но и ракурсы с сильным искажением, мешающим распознаванию деталей объектов.

В то же время для ригидного объекта его границы, характерные протяжённые (линейные и текстовые) признаки даже при неполной видимости позволяют оценить масштаб, поворот и отбросить часть сцены. Такие признаки могут быть представлены только геометрическими примитивами, т.е. малым набором параметров без описательной составляющей (дескриптора). Это существенно экономней в отношении памяти, чем локальные признаки, т.о. основные затраты при выделении и хранении состоят в дополнительных операциях.

Совокупность указанных факторов делает **актуальным** исследование методов идентификации и локализации плоских ригидных объектов, направленное на устранение недостатков существующих решений.

Основные результаты диссертации были получены в процессе выполнения работ по следующим научным грантам РФФИ:

- 19-29-09066 — Методы анализа и обработки изображений для индексации видеоданных и их устойчивого поиска в режиме реального времени
- 18-29-26035 — Методы позиционирования и ориентирования изучаемого объекта путём анализа локальных геометрических особенностей отдельных проекций
- 17-29-03514 — Алгоритмы лимитированной сложности для поиска и локализации объектов интереса полностью автономными системами на базе беспилотных летательных аппаратов
- 17-29-03370 — Методы биометрической идентификации в реальном времени на мобильном устройстве по удостоверяющей фотографии
- 17-29-03161 — Каскадные алгоритмы локализации и прослеживания протяженных объектов на мобильных устройствах
- 16-07-00616 — Исследование возможности создания новых методов поиска нечетких дубликатов в видеопотоке
- 14-07-00730 — Математическое моделирование шумовых помех при распознавании
- 13-07-12173 — Исследование методов сегментации изображений документов на структурные блоки методами цветового и морфологического анализа
- 13-07-12172 — Распознавание документов удостоверяющих личность с помощью веб камер и камер мобильных устройств

Целью данной работы является повышение точности, скорости и уменьшение затрат памяти для семейства методов идентификации и локализации

плоских ригидных объектов, основанных на сопоставлении изображений в компактном представлении.

Для достижения поставленной цели необходимо было решить следующие **задачи**:

1. Исследовать методы выделения и использования признаков различной природы на изображениях для решения задач локализации и идентификации объектов.

2. Разработать быстрые методы выделения протяжённых признаков, устойчивые к искажениям изображений, порождёнными особенностями съёмки, для плоских ригидных объектов.

3. Разработать способ учёта протяжённых признаков для повышения точности и скорости классификации и локализации плоских ригидных объектов.

4. Исследовать возможность учёта краевых условий задачи при оценке параметров модели преобразования для повышения точности и скорости оценки.

Научная новизна:

1. Впервые предложен быстрый аналитический метод поиска машиночитаемых зон (МЧЗ), устойчивый к искажениям, порождаемым при съёмке в неконтролируемых условиях.

2. Разработан новый метод, использующий локальные и протяжённые признаки, и их комбинации для более точной оценки параметров проективного преобразования образа плоского ригидного объекта.

3. Выполнено оригинальное исследование влияния различных комбинаций протяжённых и локальных признаков на точность локализации и классификации плоского ригидного объекта на изображении.

4. Для семейства методов оценки параметров модели на основе случайных выборок *-SAC предложен обобщённый метод учёта краевых условий для широкого класса задач.

Практическая значимость Разработанные в рамках диссертации методы выделения четырёхугольников, поиска машиночитаемых зон, и классификации и локализации плоских ригидных объектов были реализованы в виде программных компонентов и внедрены в программное обеспечение «Smart IDReader» и «Smart Code Engine». Данные продукты интегрированы в информационную решения ряда коммерческих организаций, а также государственных

структур Российской Федерации. Значимость подтверждается полученными патентами в РФ и США, а также актами о внедрении.

Методология и методы исследования. В диссертационной работе использовались методы системного анализа, анализа изображений, теории графов, аналитической геометрии и линейной алгебры.

Основные положения, выносимые на защиту:

1. Предложенные методы поиска протяжённых текстовых признаков позволяют с высокой точностью обнаруживать структурированные тексты (на примере МЧЗ) за счёт анализа и сопоставления структур на изображениях с референтными данными.

2. Использование краевых условий в задаче оценки проективного преобразования образа плоского ригидного объекта позволяет существенно повысить точность и скорость решения.

3. Представление изображения с помощью комбинации локальных признаков (особые точки и дескрипторы) и протяжённых признаков (прямые, сегменты, четырёхугольники) позволяет:

- повысить точность и скорость локализации и классификации
- смягчить требования к объёму памяти, необходимому для хранения ресурсов и анализа изображений (с учётом затрат на выделение и хранение протяжённых признаков).

4. Для всех предложенных методов предложены реализации, эффективные по памяти и скорости для исполнения на мобильных устройствах в режиме реального времени.

Достоверность полученных результатов обеспечивается согласованностью теоретических ожиданий с полученными экспериментальными результатами, апробацией результатов на тематических научных международных конференциях, и внедрением в реальные системы распознавания. Результаты находятся в соответствии с результатами, полученными членами научного сообщества при использовании опубликованных методов, заявленных в диссертации, и сравнении с ними.

Апробация работы. Основные результаты работы были доложены и обсуждены на:

1. Международной конференции «International Conference on Document Analysis and Recognition» (ICDAR) в 2017 и 2019 годах;

2. Международной конференции «International Conference on Machine Vision» (ICMV) в 2018, 2019, 2020 годах;

Личный вклад. Результаты диссертационной работы получены соискателем. Постановка задач и обсуждение результатов проводились совместно с научным руководителем. В коллективных работах (публикациях) соискателем выполнены построение моделей и методов для решения задач, релевантных теме диссертации, а также анализ и оценка полученных результатов. Метод поиска МЧЗ [1] был разработан, исследован и опубликован без соавторства, в работе [2] автор предложил его улучшение с помощью оценки наклона и сепаратора. В [3] автору принадлежит метод поиска для существующего графа компонент и план оценки точности. В работе [4; 5] автору принадлежат метод фильтрации границ, методы обнаружения в пространстве Хафа нескольких линий-кандидатов, методы ранжирования и фильтрации гипотез, идеи восстановления и оценки 4й стороны и методы без оптимизации параметров. В работе [6] автору принадлежит метод выделения границ с фильтрацией, методы фильтрации гипотез по геометрии прообраза, восстановленного с использованием фокусного расстояния. В работе [7] автором выполнен анализ особенностей решения задачи локализации документов на снимках с камер. В работах [8; 9] использованы авторские методы для фильтрации точек и предложенное упрощение модели преобразования. В работах [10; 11] автору принадлежат идеи и методы использования геометрических примитивов при выделении локальных признаков в RANSAC, а так же отдельные критерии отбраковки моделей и гипотез; в [12] автором выполнена оценка влияния этих методов и критериев для разных методов дескрипции. В работах [13; 14] использованы авторские критерии отбраковки, автором выполнена оценка их влияния на скорость и точность оценки параметров. В работе [15] автору принадлежит идея фильтрация точек с нечётким классификатором объект-фон и способ оценки зоны объекта на отфильтрованных данных. Кроме того, автор участвовал в создании наборов данных, использованных в [8; 10; 14].

Соответствие работы паспорту научной специальности. Диссертационная работа соответствует направлениям исследований паспорта научной специальности 2.3.1 - «Системный анализ, управление и обработка информации, статистика» (п. 3 «Разработка критериев и моделей описания и оценки эффективности решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации», п. 4 «Разработка методов и ал-

горитмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации», п. 5 «Разработка специального математического и алгоритмического обеспечения систем анализа, оптимизации, управления, принятия решений и обработки информации»).

Публикации. Основные результаты по теме диссертации изложены в 15 работах, 3 из которых изданы в журналах, рекомендованных ВАК, включая 2 в изданиях категории K2, 3 — в научных журналах, индексируемых Springer, Web of Science, Scopus, и приравненных к категориям K1 и K2, 9 — в сборниках трудов конференций (индексируемых Web of Science и Scopus). Зарегистрированы 3 патента и 3 программы для ЭВМ.

Объем и структура работы. Диссертация состоит из введения, 3 глав, заключения и 2 приложений. Полный объём диссертации составляет 128 страниц, включая 27 рисунков и 13 таблиц. Список литературы содержит 159 наименований.

Глава 1. Признаки изображений, их типы и методы детекции.

Интерпретация и анализ признаков для локализации и классификации объектов на изображениях.

1.1 Введение: Особенности распознавания объектов на изображениях на сегодняшний день

Системы анализа и распознавания (обработки) изображений с развитием вычислительных мощностей и устройств регистрации изображений из единичных промышленных решений превратились в обычное дополнение к комплексным системам в различных областях [16]. Государственные и коммерческие предприятия, промышленность, финансовые, медицинские и прочие организации массово используют такие системы как для внутренних процессов (автоматический ввод и организация данных [17; 18], решение проблем информационной, производственной и государственной безопасности [19–21]), так и для упрощения взаимодействия с внешними контрагентами (удалённая регистрация, аутентификация, оплата и др. [22]).

При таком многообразии систем и запросов одно и то же изображение может требовать откликов различной степени детализации. Для задач классификации это хорошо отражено в системах и работах, посвящённых анализу изображений документов, живописи, фото- и видеоконтента [23–26]. Документ может быть отнесён к одной из широких категорий, таких как «страница журнала», «платёжный документ» или «удостоверение личности». Но также возможно определить более точный тип – «паспорт страны X назначения Y года Z», или «счёт-фактура от компании X». В случае картины может потребоваться как информация о стиле или возможном авторе для исследовательских целей, так и точное название, необходимое для мобильных гидов. При высокой полноте информации классы могут быть представлены частично или полностью зрительно сопоставимыми изображениями. В распознавании образов такие классы называются жёстко структурированными (риgidными). Таким образом, документы, удостоверяющие личность, в целом представляют собой неструктурированный класс. Однако биометрический паспорт РФ 2023 года выпуска частично структурирован.

Согласно [27] структуру объектов на изображении могут задавать достаточно простые (непроизводные) признаки, такие как пересечения, пятна, границы и линии, разрывы в яркостных переходах. Жёсткость связана с тем, насколько эквивалентны сами признаки и связи между ними внутри экземпляров класса. Совокупность признаков обеспечивает возможность использования локальной и глобальной информации, содержащейся в изображении, включая геометрические характеристики.

Преимущественно в задачах анализа и распознавания изображений [16; 27] рассматриваются двумерные растровые изображения. В качестве средств регистрации могут выступать как стационарные (промышленные) камеры и сканеры, так и малые цифровые камеры. Именно последние применяются при оснащении общеиспользуемых устройств – смартфонов, веб-камер, видеорегистраторов. В работе [28] рассмотрены особенности формирования изображений для различных устройств. Малые цифровые камеры, при более высокой скорости получения изображения, вносят больше искажений в процессе оцифровки сигнала. Это связано как со сложностью оптической системы, так и с особенностями процесса съёмки камерой. Сам процесс преобразования аналогового сигнала в цифровой является источником цифрового шума (из-за ошибок квантования и т.п.). Одним из факторов являются слабо контролируемые условия съёмки: недостаточное или неравномерное освещение, неизвестный поворот камеры относительно глобальной системы координат, неравномерное и непредсказуемое движение камеры либо объекта. В результате на изображениях могут возникать блики, перепады яркости и смазанные области. Кроме того, наблюдаемая сцена может быть геометрически искажена, если поза камеры не соответствует вектору наблюдения. На рисунке 1.1 показано, как одна и та же сцена может выглядеть при переменае освещения и ракурса.



Рисунок 1.1 — Вариации освещения и точки съёмки для сцены

Хотя объекты, представляющие интерес, могут быть произвольной формы и объёма, у преимущественной части создаваемых человеком конструкций про-

образом являются прямоугольники и параллелепипеды. Сцены с характерными признаками – прямолинейными границами, расположеннымими преимущественно по трём взаимно ортогональным направлениям – называют «манхэттенским миром» (по аналогии с тем, как расстояние городских кварталов Минковского называют манхэттенским). В работе [29] показано, что несмотря на первоначальную привязку к рукотворным сценам, на естественных сценах образующие направления также хорошо прослеживаются. При проецировании трёхмерных сцен «манхэттенского» мира на изображение появляются точки схода – точки пересечения проекций параллельных прямых, по одной на каждое направление, как показано на 1.2. Точки схода, согласно принципам, указанным в [27] являются признаком наравне с прямыми.

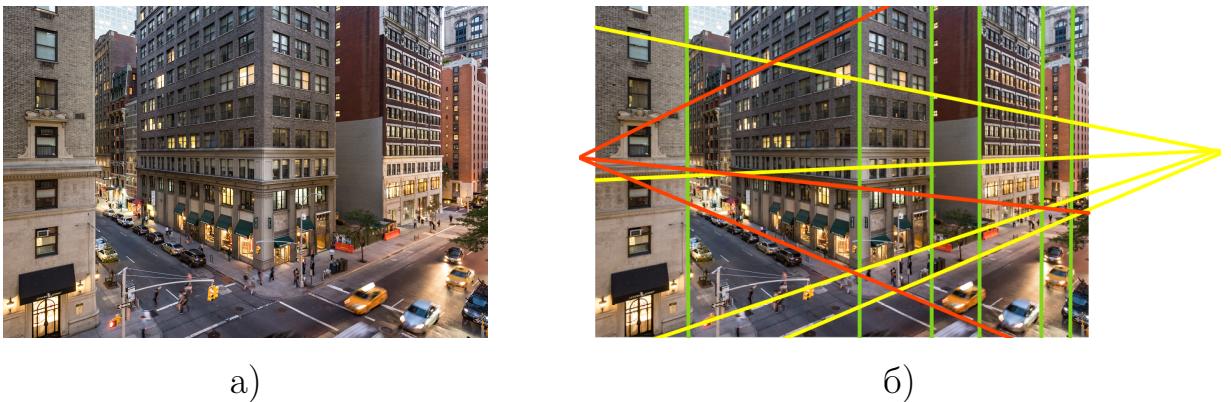


Рисунок 1.2 – Городская сцена как пример манхэттенского мира
а) исходное изображение; б) линии сцены, задающие 3 точки схода

1.2 Задача локализации и идентификации на изображениях

Формальная постановка задачи классификации (идентификации) объекта на изображении, часто встречающаяся в литературе, выглядит следующим образом: Требуется определить такой класс изображений \bar{c} из множества классов C , что

$$F(I, \bar{c}) < F(I, c'), \forall c' : c' \neq \bar{c}, c' \in C \quad (1.1)$$

где F – функция, определяющая близость изображения I к классу. Функция F зависит того, какое структурное представление выбрано для формирования классов. Класс может задавать как широкую категорию (документы, рисунки,

обложки), так и точный экземпляр объекта (знак «стоп» среди дорожных знаков).

Задача локализации объекта также может формулироваться с разной степенью точностью. Простая локализация требует приблизительное знание о позиции и размере объекта. В работах она чаще всего задаётся с помощью окаймляющего прямоугольника. Точная локализация подразумевает определение контура его внешних границ на изображении. Для определения точности локализации произвольных объектов существует ряд показателей, такие как коэффициент сходства Жаккара. Если истинный (размеченный) объект покрывает множество пикселей A , а метод обнаружил множество B , то точность локализации по Жаккару определяется следующим образом:

$$\text{IoU} = \frac{A \cap B}{A \cup B}, \text{IoU} \in [0, 1]. \quad (1.2)$$

Без привязки к глобальной системе координат объекты, точно локализованные по этой мере, нельзя соотнести между собой, поскольку множества пикселей не упорядочены. Привязку определяют параметрами преобразования. Наиболее общим преобразованием, подходящим для всех регистраторов изображений, является проективное преобразование.

В работах, посвящённых данным задачам, предлагаются как последовательные, так параллельные схемы решения:

- Последовательная схема: сначала локализация либо нормализация изображения, затем идентификация
- Параллельная схема: локализация и идентификация объекта выполняются одновременно

Проективная нормализация – это получение изображения, на котором образ объекта подобен прототипу, а границы окаймляющего прямоугольника параллельны осям изображения. За счёт этого для идентификации можно использовать методы, слабо устойчивые к проективным искажениям (например, классификаторы Виолы-Джонса). Финальные параметры локализации при этом складываются из параметров нормализации, сдвига на нормализованном изображении и установленном масштабе.

При параллельной схеме гипотеза преобразования H и оценка близости F рассчитываются одновременно. Так спроектированы наиболее точные на сегодняшний день решения: end-to-end модели, полученные методами сквозного

обучения, и аналитические решения, основанные на сопоставлении наборов локальных признаков. Как отмечено во введении, аналитический подход более применим в нашей задаче как с точки зрения ожидаемой вычислительной эффективности, так и ввиду требований к данным.

Рассмотрим далее в главе методы и алгоритмы, используемые для выделения на изображениях признаков разных типов и их последующего сопоставления.

1.3 Типы признаков и методы их выделения

1.3.1 Границы, края

Наблюдаемые прямолинейные границы, в зависимости от их протяжённости и непрерывности, могут быть описаны как прямые или сегменты. Аналитические методы детекции прямых, как правило, оперируют не исходным изображением, а картой границ. В обзорной работе [30] рассматриваются типы границ (край, хребет и т.п.) и набор базовых методов для выделения как прямолинейных границ, так и иных контуров. На рисунке 1.3 показаны изображения и графики интенсивности, соответствующие различным типам границ.

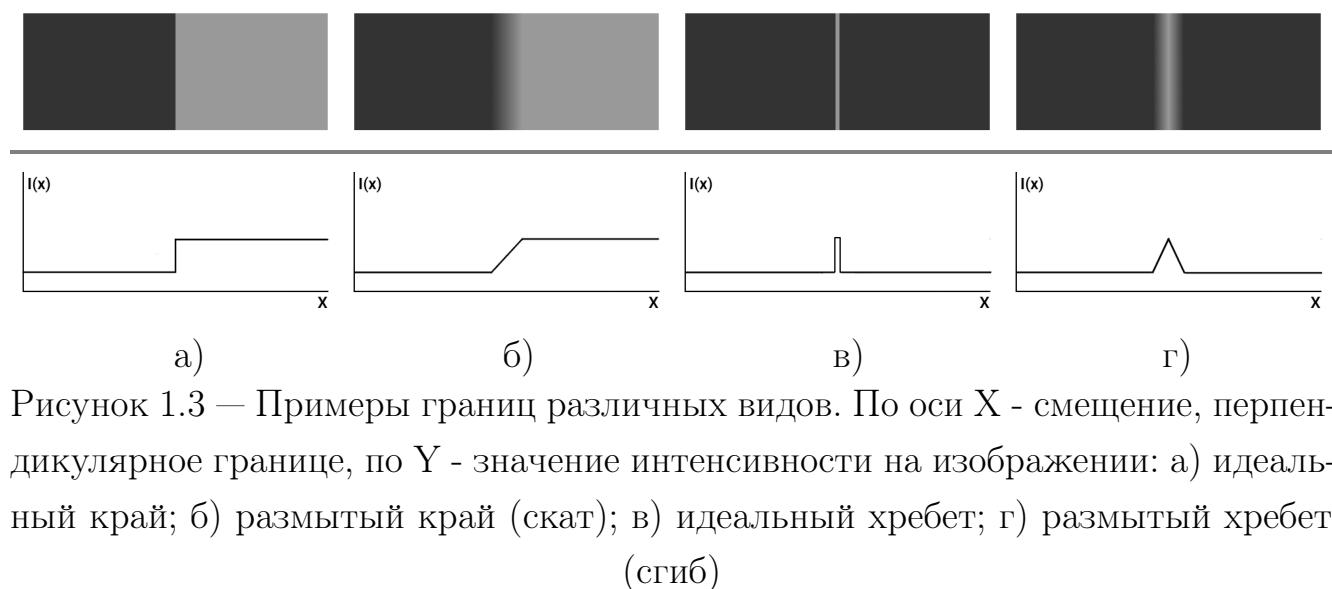


Рисунок 1.3 — Примеры границ различных видов. По оси X - смещение, перпендикулярное границе, по Y - значение интенсивности на изображении: а) идеальный край; б) размытый край (скат); в) идеальный хребет; г) размытый хребет (сгиб)

Пусть $I(x, y)$ здесь и далее обозначает яркость (интенсивность) изображения I в пикселе x, y . Для выделения границ могут быть использованы следующие базовые операторы:

- дискретный аналог производной по направлению на изображении

$$I'_y(x, y) = |I(x, y) - I(x, y + 1)| \quad (1.3)$$

Выражение (1.3) задаёт стандартный оператор производной для вертикального направления. Чем выше значение I' , тем более выражена граница между пикселями соседних строк.

- градиент по изображению

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \rightarrow \begin{aligned} G_1(x, y) &= I(x, y) - I(x + 1, y + 1) \\ G_2(x, y) &= I(x, y + 1) - I(x + 1, y) \end{aligned} \quad (1.4)$$

$$|G(x, y)| = |G_1(x, y)| + |G_2(x, y)| \quad (1.5)$$

$$\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \rightarrow \begin{aligned} G_x(x, y) &= I(x - 1, y - 1) + 2I(x - 1, y) + I(x - 1, y + 1) \\ &\quad - I(x + 1, y - 1) - 2I(x + 1, y) - I(x + 1, y + 1) \\ G_y(x, y) &= I(x - 1, y - 1) + 2I(x, y - 1) + I(x + 1, y - 1) \\ &\quad - I(x - 1, y + 1) - 2I(x, y + 1) - I(x + 1, y + 1) \end{aligned} \quad (1.6)$$

$$|G(x, y)| = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \quad \theta = \arctg \frac{G_x}{G_y} \quad (1.7)$$

Выражения (1.4) и (1.6) задают градиент через матричные операторы по направлениям. Приведённые варианты предложены Робертсон(по диагоналям) и Собелем(осям). Модуль градиента в формулах ((1.5) и (1.7)) соответствует выраженности границы, а угол θ – её наклону к ОХ. Для устойчивости к шуму любой из операторов можно расширить на большую область, либо модифицировать дополнительными фильтрами. Например, для (1.6) существует модификация, которая улучшает его вращательную симметрию. В обзорной работе подчёркнута необходимость дальнейшей фильтрации после применения базовых операторов.

Классическим среди несетевых детекторов границ считается метод, предложенный Канни [31]. В работе сформулированы критерии эффективности оптимальных операторов для границ типа «край»: высокая точность и полнота

детекции, истинность границы (координатная точность) и отсутствие избыточных откликов (одной реальной границе соответствует одна найденная). Показано, что первая производная фильтра Гаусса является хорошим приближением к оптимальным операторам. Для изображения, ввиду ассоциативности свёртки, можно выполнить сначала выполнить фильтрацию, а затем рассчитать градиент. После вычисления градиента используется фильтрация с двумя порогами, высоким и низким. На первом этапе выявляются надёжные, выраженные границы – пиксели, значения которых превышают высокий порог. После этого выполняется следующее правило: пиксель, соседний к граничному, также считается границей, если его значение превышает слабый порог.

Определение надёжных границ легло стало основой подхода на основе нечёткой логики, в котором два порога T_{max} и T_{min} определяют границу ($I(x,y) \geq T_{max}$) либо фон ($I(x,y) < T_{min}$), а для определения принадлежности оставшихся пикселей задают набор правил вида «если-то». Например, в работе [32] значения морфологического градиента цветного изображения [33] задают вероятность принадлежности пикселя границе. Нечёткий детектор [34] в работе использует одновременно нечёткую дивергенцию и минимизацию нечёткой энтропии при выделении начального «надёжного» множества на сером изображении.

Информативность многоканальных изображений может быть использована для существенного улучшения выделенных границ. Особенности расчёта градиента для цветного изображения рассмотрены в работе [35]. Вместо комбинаций независимых результатов операторов в каждом канале (максимум, среднеквадратичное), предлагается рассчитать тензорный градиент. Для этого многоканальное изображение рассматривается как векторное поле. В результате на границы влияет не только интенсивность в разных каналах, но и совпадение направления. Это порождает более точные границы, чем комбинация операторов, но и вычислительно сложней. В [36] градиент по яркости предлагают комбинировать с цветовым градиентом и текстурными признаками. По данной комбинации признаков по разметке обучен линейный классификатор с помощью SVM. Показано, что результат такого классификатора ближе к человеческому восприятию границ (разметке), чем детектор Канни и другие операторы.

Использование машинного обучения мотивировано в том числе сложностью с выбором порогов для аналитических методов [37]. В последних исследованиях много внимания уделяют построению карт границ с использованием

искусственных нейросетей (ИНС). Разработки в данном направлении тормозят сложность получения референсных данных в больших объёмах. К примеру, при рассмотрении работы [38] можно отметить, что повышение метрических показателей качества (*F-measure*) в сравнении с иными методами выглядит, как результат переобучения: на рисунках 1.4 в статье можно обнаружить, что как выделенные, так и размеченные границы не всегда соответствуют реально наблюдаемым. Тем не менее, ИНС, предложенная в [39], убедительно демон-

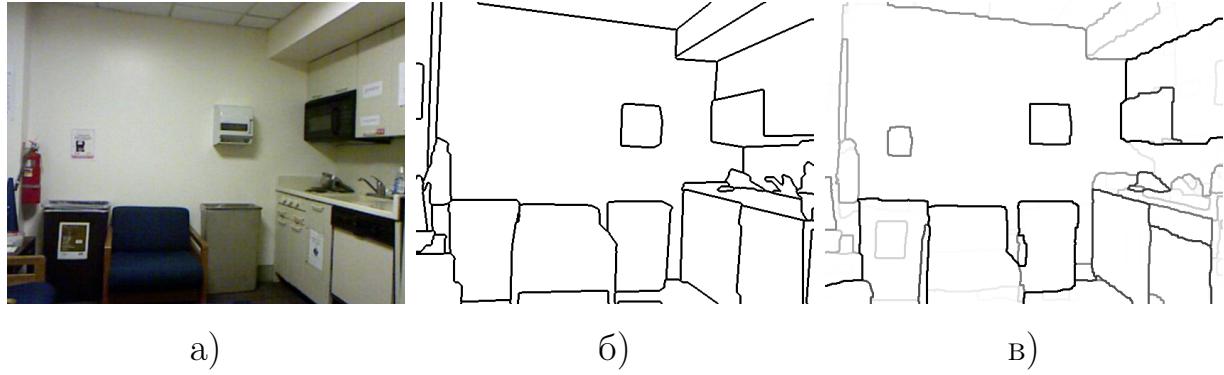


Рисунок 1.4 — Карта границ а) исходное изображение б) разметка в) результат сети.

Изображения из работы [38]

стрирует хороший результат при реимплементации, в т.ч. на новых наборах данных [40; 41]. Особенность данной ИНС в том, что для итоговой карты границ используется комбинация из выходов в т.ч. промежуточных слоёв, а не только выход последнего свёрточного слоя. При этом замедление вычислений (на CPU, относительно с аналитических методов) хотя ожидаемо, но слишком велико для случаев, когда поиск границ лишь один из промежуточных этапов. Детализация экспериментов (оборудование и результаты) показывает, что проблема сохраняется для многих ИНС [42], несмотря на ускорения.

1.3.2 Прямые линии и сегменты

Многофункциональным для задач поиска примитивов является двумерное преобразование Хафа (ПХ), при котором всему примитиву на xy плоскости соответствует точка в некотором пространстве параметров. Для детекции прямых устоявшимся решением является нормальная параметризация:

$$\rho = x \sin(\theta) + y \cos(\theta)$$

где ρ - расстояние от центра координат до прямой («радиус»), θ - угол между радиусом и осью Ox . Использовать ρ - θ параметризацию вместо k - b (угловой коэффициент и сдвиг) для ПХ предложили в работе [43] для того, чтобы а) избежать неограниченных значений параметров б) каждой прямой соответствовала единственная точка в пространстве параметров. Это достигается, поскольку $\rho \geq 0$ и ограничено сверху размерами изображения, а уникальность обеспечивает ограничение $\theta \in [0, \pi]$. Пример преобразования Хафа для прямой, заданной точками, по k - b параметризации приведён на рис. 1.5.

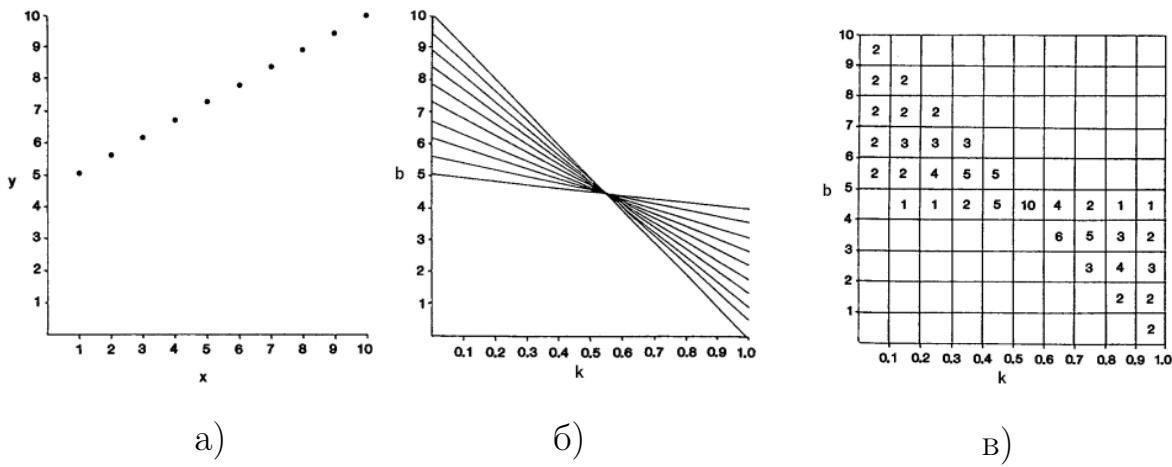


Рисунок 1.5 — а) бинарное изображение б) преобразование Хафа для бинарного изображения (а) в) аккумулятор преобразования Хафа (б)

Предполагалось, что преобразование выполняется на бинарном изображении границ. В работе [44] предложено расширение метода для работы с полутоновыми изображениями, где интенсивность пикселя учитывается при построении отображения в пространство Хафа. ПХ в тривиальных реализациях затратно как по числу операций (т.е. времени исполнения), так и по необходимой памяти. Поэтому большую практическую значимость имеют последующие работы, посвящённые оптимизации вычислений. Брейди в работе [45] сравнил существующие ускорения и представил способ вычисления аппроксимированного преобразования за $N \log N^2$ (против N^3 в лучших прошлых случаях). Был предложен новый метод объединения частичных результатов, благодаря которому получаемая аппроксимации вычисляется с доказуемой (и достаточно высокой) точностью. На данный момент существуют и более точные аппроксимации, необходимые для обработки медицинских изображений, однако вычислительно они более затратны [46]. Кроме того, алгоритм Брейди может быть реализован

нерекурсивно с ускорением за счёт архитектурных расширений [47], позволяющих обрабатывать за 1 операцию вектор данных (векторизация SIMD).

Методы контурного анализа позволяют вычленять из границ прямолинейные участки – отрезки. Каждый отрезок формируется с отдельного граничного пикселя и попиксельно наращивается, чтобы получить связный контур, удовлетворяющий критериям прямолинейности. В отличие от ПХ, учёт прерывистых границ в таких методах затруднён, поскольку на коротких участках недостаточно информации для проверки критерия. Один из базовых методов данного типа представлен в работе [48]. Края выделяют с помощью свёртки, а затем уточняют и фильтруют по пороговому значению. После этого определяют связность соседних пикселей по схожести ориентации градиента. Связный контур пошагово аппроксимируют сегментом. Если на контуре возникает ошибка отклонения, превышающая пороговое значение, сегмент «разламывают» в точке наибольшей ошибки. Метод был усовершенствован в [49]: с помощью компонент связности можно группировать соседние точки с одинаковой ориентацией градиента, что ускоряет дальнейшее построение. В методе LSD [50] итеративно наращиваются целые прямоугольные области. Рассчитывается градиент изображения по ур. (1.7), и создание новой области начинается с непросмотренного пикселя с наивысшей магнитудой. Каждая область начинается с одного пикселя, и угол области считается равным ориентации градиента в этом пикселе. Затем оцениваются ближайшие пиксели-соседи: если ориентация градиента в пикселе близка к углу области, то он присоединяется к области и маркируется как просмотренный. Значение угла области обновляется с каждым добавленным пикселем. Центр прямоугольника области определяет центр искомого сегмента, а основное направление соответствует главной оси инерции прямоугольника, где магнитуда задаёт вес. Границы области определяют концы сегмента.

Для достижения лучшей скорости метода EDLines [51] авторы заявили свой подход к порождению карты границ. Для градиента изображения вычисляют набор пикселей-якорей – пикселей, которые с большей вероятностью окажутся элементами границы. В качестве якорей предложено использовать пики градиента. Якоря соединяют, прорисовывая границы между ними. Начиная с одного якоря, метод пользуется величинами и направлениями градиента в соседних пикселях, перемещаясь к следующему якорю по максимумам. Для извлечения сегментов авторами используется метод наименьших квадратов (МНК). Алгоритм рекурсивно обрабатывает все оставшиеся пиксели в цепочке, вписы-

вая линии в имеющийся набор, до тех пор пока ошибка не превысит заданного порога, либо пока не будут обработаны все. Т.е. идея в том, чтобы сгенерировать минимальную длину начального отрезка, а затем итеративно удлинять отрезок. Заявлено, что EDLines в 10 раз быстрее, чем LSD, при идентичной детализации (рис. 1.6).

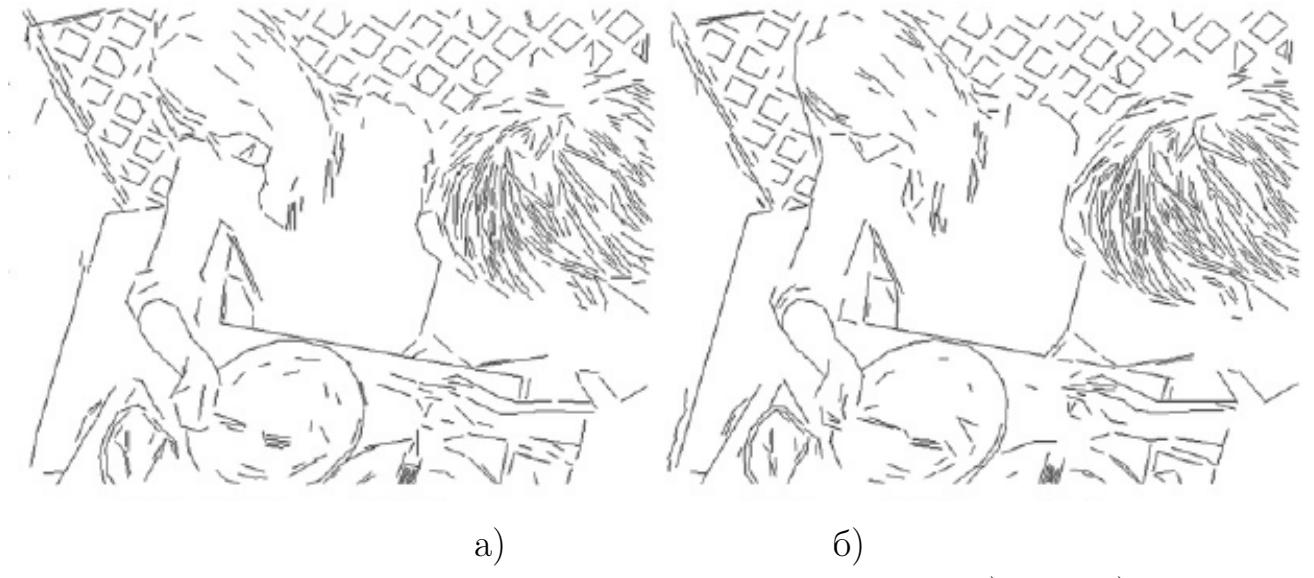


Рисунок 1.6 – Результат работы детекторов отрезков а) LSD б) EDLines
Изображения из работ [50] и [51]

Ускорение EDLines, предложенное в работе [52], достигается за счёт до-
страивания линии компонентой связности и использования её направления. Та-
кой ход также позволяет перепрыгивать разрывы градиента и обнаруживать
целые линии или отдельные сегменты без скачков. Метод AG3line [53] исполь-
зует стратегию активной группировки пикселей. Пиксель-кандидат к сегменту
оценивается с учётом геометрии линии, точность может достигать одного пик-
селя. Чтобы уменьшить разрывность, вызванную откинутыми нестабильными
пикселями, зона пикселей-кандидатов расширяется по линии, согласно ограни-
чениям геометрии.

Последние работы сосредоточены на исследованиях возможностей нейро-
сетей, в т.ч. для выделения подобных примитивов. В работе [54] для поиска
линий применяют архитектуру Transformer [55], в которой каркасный анализ
(с участием выделения признаков-границ) заменён на многомерное кодирова-
ние-декодирование. Для постепенного уточнения линий используются т.н. слои
«внутреннего внимания». Популярность метода LSD отражается на сетевых ме-
тодах. В работе [56] предлагается детектор LSDNet с локализованным приме-

нением ИНС. Лёгкая сеть встраивается в классический алгоритм LSD. Авторы заменяют первый шаг алгоритма - построение тепловой карты отрезков линий и поля касательных из необработанных градиентов изображения — на облегчённую ИНС, которая способна рассчитывать более сложные и ёмкие признаки. Второй этап - выделение сегментов по карте - несколько модифицирован. Карта границ бинаризуется комбинированным методом - с глобальным и локальным порогами. Построение базовым прямоугольников модифицировано с учётом изменений карты, введена иная мера сходства для пикселей. DeepLSD [57] обрабатывает изображения с помощью глубокой сети для создания поля притяжения линий, а затем преобразует его в суррогатную величину и угол градиента изображения, которые затем передаются в любой существующий линейный детектор ручной работы. Авторы также предлагают новый инструмент оптимизации для уточнения сегментов линий на основе поля притяжения и точек схода. Усовершенствование значительно повышает точность современных глубинных детекторов.

1.3.3 Точки схода и прямоугольники

Прямые линии и сегменты могут служить базисом для выделения более сложных признаков, таких как точки схода и четырёхугольники. Как отмечено в 1.1, основной интерес вызывают такие четырёхугольники, которые соответствуют прямоугольникам и параллелограммам на реальном прототипе и его изображении-эталоне. Методы поиска четырёхугольников с использованием сегментов и прямых устанавливают соответствия между его сторонами и найденными примитивами. После поиска примитивов значимыми этапами в подобных (*контурно-линейных*) методах являются генерация, оценка и фильтрация гипотез – комбинаций примитивов, образующих угол либо целый четырёхугольник.

Одной из базовых работ считается [58], в которой локализуют прямоугольную доску с произвольным заполнением. Авторами предложен алгоритм, основанный на контурно-линейном подходе, который включает следующие этапы: детекция границ типа «край», поиск прямых, формирование четырёхугольников (вариантов расположения документа), их фильтрация, ранжирование и, наконец, уточнение в большем разрешении. Четырёхугольник формируют из

прямых линий, найденных с использованием модифицированного БПХ. В модификации параметр угла θ находится в интервале $[-180^\circ, 180^\circ]$ и привязан к направлению градиента: $\theta = \text{atan2}(G_y, G_x)$. Определены ограничения на θ для прямых, что используется для ускорения перебора:

- а) прямые противоположных сторон имеют противоположные направления (разница в $180^\circ(\pm 30^\circ)$),
- б) угол между соседними сторонами составляет $90^\circ(\pm 30^\circ)$,
- в) направления линий непротиворечивы, образуют цикл по часовой стрелке либо обратный.

Кроме того, направление используется для оценки четырёхугольника. Вес прямой ПХ рассчитан на целом изображении, поэтому может возникнуть случай, когда нет границ вдоль какой-либо стороны, как на рисунке 1.7. Авторы пред-

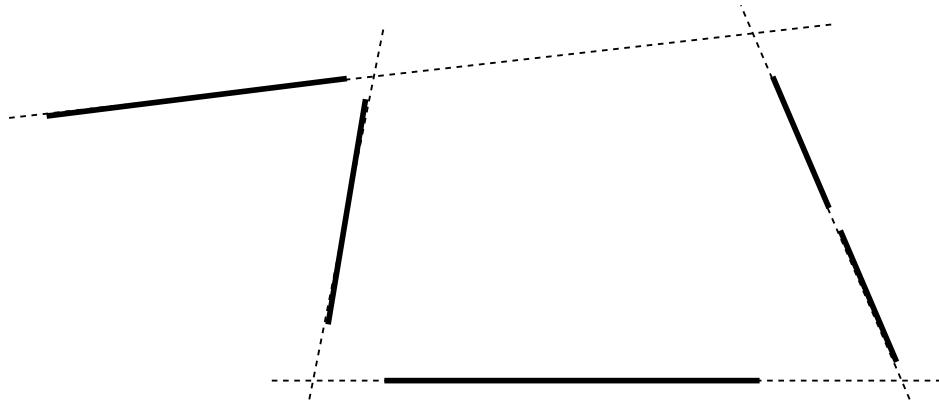


Рисунок 1.7 — Пример некорректного четырёхугольника. Реальные края показаны сплошным чёрным, прямые ПХ - прерывистыми линиями.

ложили проверять наличие краёв между углами четырёхугольника, при этом направление градиента должно соответствовать общему направлению вдоль образующей прямой. Так же в работе используется предположение, что угол прообраза четырёхугольника прямой, для восстановления реального соотношения сторон, а также оценки фокусного расстояния. Кроме того, для большей вычислительной эффективности авторы выполняют основной анализ при грубой дискретизации для ПХ и в масштабе, существенно меньшем реального. Затем полученное грубое решение уточняют на исходном масштабе с помощью МНК.

В более новых работах вопрос быстродействия методов поднимается, как правило, в контексте исполнения на мобильных устройствах. В публикации [59]

для уменьшения времени работы алгоритма авторы снижают разрешения входного изображения и аккумулятора ПХ, используют таблицу поиска и эвристики при вычислении карты границ, и вводят регион интереса для уменьшения зоны поиска документа. Авторы используют детектор границ Канни с адаптивным порогом, а затем фильтруют найденные границы в зоне текста – при сильном сжатии строки порождают ложные прямые на ПХ. Зоны текста определяют с помощью анализа компонент связности.

Локализация на изображении плоских прямоугольных объектов без внутренних текстур рассмотрена в [60]. Модель объекта задана совокупностью примитивов-сегментов, соответствующих прямолинейным границам. Авторы оперируют выделенными сегментами для генерации гипотез углов. Пара сегментов образует угол, если расстояние от точки пересечения образующих до концов сегмента меньше порога. Пары углов образуют 2 варианта корректных сочетаний:

- пара углов имеет общую сторону, тогда сегмент для 4-й стороны выбирается, как наиболее близкий;
- пара углов формирует четырёхугольник.

Гипотезы четырёхугольников ранжируются по суммарному отклонению сегментов модели от найденных при проецировании.

Несколько состыкованных ячеек-рамок комиксов детектируют в [61]. Рамки могут быть частично перекрыты текстовыми выносками. Предложено выделить сегменты и кластеризовать их, если пересекается окаймляющая зона. Кластер может быть разбит на несколько в зависимости от типа, который определяют по стыкам сегментов. Для формирования четырёхугольника секции из кластера берут по 2 горизонтальные и вертикальные прямые. Чтобы избежать избыточной генерации, сегменты фильтруют длине (не должно быть мелких), а гипотезы – по соотношению сторон, минимальному размеру и «типу соединения» сегментов (рис. 1.8). Как и в [58] оценивают «полноту» гипотезы, т.е. насколько покрыт периметр реальными границами.

Некоторые типы методов не требуют явного выделения базиса. В работе [62] предлагается дополнить нейросеть, извлекающую признаки линий, геометрическими отображениями – преобразованием Хафа и отображением Гаусса (сферическое). Вектора признаков, полученные сетью, отображаются в пространство Хафа, задавая линии. Ячейки аккумулятора Хафа с свою очередь отображаются на сферу, где прямые переходят в окружности, а точки схода, со-

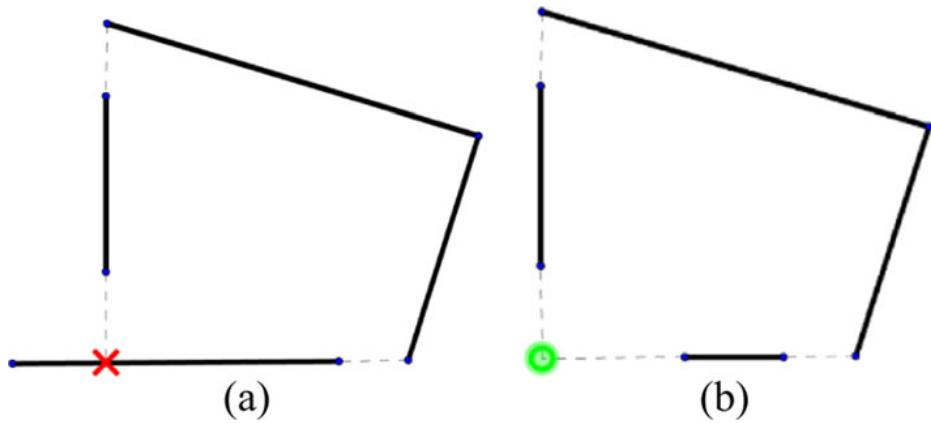


Рисунок 1.8 — Пример а) Т- и б) L-соединений сегментов. Разрешённым считается только L. Изображение из работы [61]

ответственно, расположены в точках пересечения окружностей. Преимущество использования окружностей в переходе из неограниченной плоскости изображения в ограниченную единичную сферу. Это облегчает обнаружение точки схода за пределами изображения. Четырёхугольники также могут быть выделены с использованием нейросетевых методов, например в [63; 64] используется оригинальная U-net нейронная сеть для сегментации изображений. В данных сетях, как правило, миллионы параметров, поэтому множество исследований направлено на достижение высокой точности с уменьшением их числа. В работе [65] авторы модифицировали U-net архитектуру сети, уменьшив число параметров более чем на 70%. В исследовании [66] достигнуто уменьшение числа параметров в 100 раз за счёт применения прямого и обратного преобразования Хафа.

1.3.4 Локальные признаки. Особые точки и дескрипторы

Непротяженные локальные признаки изображения условно типизовали на углы, пятна и точки. Такое разделение было достаточно условным, поскольку признаки визуально перетекают друг в друга при разных масштабах и шумах. В научных работах последних лет фигурируют преимущественно понятие особой (ключевой) точки. Предполагается, что у особой точки, помимо координат центра, могут быть дополнительные параметры - размер (от пятна), ориентация/маркер неориентированности (от угла) и вес. Особой точке ставят в соответствие функцию от её локальной окрестности - дескриптор. Каждой точке

соответствует небольшая часть изображения, соответствующая её окрестностям - патч (рис. 1.9).

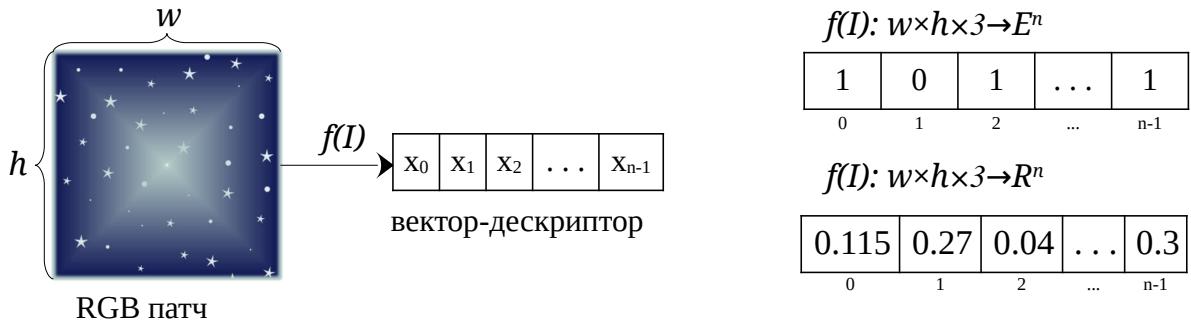


Рисунок 1.9 — Преобразования патча в дескриптор

В работе [67] Моравец представил детектор углов - признаков, которым характерен высокий контраст в ортогональных направлениях. Для квадратного окна рассчитывают суммы квадратов разностей интенсивностей пикселей для 8 смещений. Направление и оценка окна соответствуют смещению с минимальной суммой, а углами считаются центры тех окон, у которых минимальная оценка - локальный максимум среди соседних окон. Харрис в работе [68] доработал детектор углов, рассчитав структурный тензор для более точной оценки направлений. Если у Моравеца отклик анизотропен только по направлениям смещений, то у Харриса отклик анизотропен по всем направлениям при квадратном окне и изотропен при гауссовом окне. Помимо этого, в работе Харриса были сформулированы требования к особым точкам:

- *Повторяемость* – точка находится в том же месте сцены несмотря на изменения точки обзора и освещения
- *Значимость* – каждая точка имеет уникальное описание
- *Компактность* и *эффективность* – количество точек существенно меньше числа пикселей изображения
- *Локальность* – точка с размером занимает маленькую область изображения, поэтому работа с ней не чувствительна к перекрытиям

В дальнейшем для оператора Харриса было предложено несколько улучшений. Для устойчивости к изменениям масштаба в работе [69] предлагается использовать масштаб-нормированный оператор Лапласа на этапе выбора шага масштабирования. Ограничения задач в ранних работах позволяли использовать в качестве дескриптора яркость пикселей окрестности в явном виде и

сравнивать только близкие по координатам точки – для этого определяли во-круг точек окно поиска и окно корреляции. Мерой близости считали попик-сельно рассчитываемую корреляцию. В [70] авторы показывают, что метод расчёта матрицы с поиском особых точек работает на изображениях типов и разными типами преобразований (движение вбок и вперёд, вращение, отличие камер и др.). Рассматриваются вопросы множества сопоставленных точек-кандидатов, их проверки, ранжирования и согласованности.

Переходной работой к общему поиску является [71], в которой локальные признаки используются для поиска нечёткой копии изображения из набора – более 1000 изображений. Авторы используют для расчёта дескрипторов вектора гауссовых производных в точках. Устойчивость к масштабу достигается за счёт гауссовой пирамиды масштабов – набора изображений с масштабированием от точного к грубому соответствующей σ_G . Дескриптор рассчитывается в точке, проецируемой на каждый уровень пирамиды. Инвариантность и компактность полученного дескриптора позволила избавиться от условия геометрической близости у точек на сравниваемых изображениях, и, соответственно, ограничительных окон поиска. Для быстрого сравнения изображений, представленных наборами многомасштабных дескрипторов, использовали преобразование расстояния Махаланобиса в Евклидово. Евклидово расстояние позволило авторам повысить эффективность поиска за счёт индексирующих структур. Сопоставление с базой изображений проходит по схеме голосования – каждому дескриптору изображения-запроса ставится в соответствие дескрипторы из базы, расстояние до которых меньше порога. Дескриптор голосует за то изображение, из которого был извлечён. В результат идёт изображение, набравшее максимум голосов. Может существовать несколько альтернатив с одинаковым или близким количеством голосов. В этом случае предлагается:

1. у каждой точки изображения в базе есть p соседей, 50% из них должны быть сопоставлены,
2. углы векторов к между сопоставленными точками-соседями должны быть согласованы .

На рис. 1.10 показан упрощённый пример такого сопоставления точек.

Такой поиск оказался устойчив к преобразованиям подобия, небольшим проективным искажениям и частичным заслонениям. Последующие работы в области направлены на уменьшение чувствительности детектором к перепадам масштаба и устойчивости дескрипторов к аффинным преобразованиям. Одним

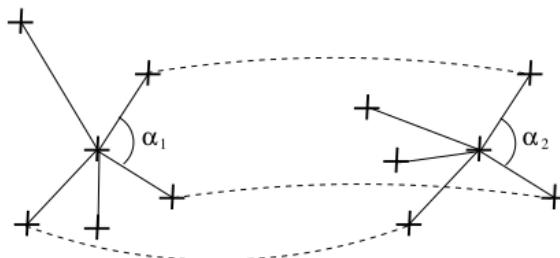


Рисунок 1.10 — Сопоставление точек. Угол α_1 согласован с α_2 .

из ключевых алгоритмов на этом пути является SIFT, представленный в работе [72]. SIFT одновременно локализует особые точки и рассчитывает для их дескриптор. Рассматриваются пирамида гауссиан и их разности. Точка считается особой, если она локальный экстремум разности гауссиан. В отличие от точек Харриса, такие точки считают «пятном» и они имеют «размер», определяемый через уровень пирамиды и коэффициент размытия. Помимо этого, для точки рассчитывают направление (ориентацию) (1.9) и магнитуду (1.8) исходя из направлений градиентов соседних точек.

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (1.8)$$

$$\theta(x, y) = \operatorname{tg}^{-1} \frac{(L(x, y+1) - L(x, y-1))}{(L(x+1, y) - L(x-1, y))} \quad (1.9)$$

В формулах (1.8) и (1.9) $L(x, y)$ является свёрткой $I(x, y)$ с фильтром Гаусса с σ_G определённого уровня пирамиды. Дескриптор рассчитывается на уровне пирамиды, где точка является экстремумом. Окно поворачивается на направление точки, за счёт чего достигается инвариантность к повороту. Дескриптор рассчитывается как гистограмма градиентов в 4 регионах окна вокруг точки, а затем домножается на σ_G соответствующего уровня. Таким образом получается 128-мерный вещественный дескриптор.

Алгоритм SURF [73] разработан как ускоренная замена SIFT. Для детекции предлагается ускорить метод детекции за счёт простой аппроксимации при расчёте матрицы Гессе – вторая производная Гаусса приближается прямоугольным фильтром, что по качеству близко к результатам обычных расчётов после дискретизации. Для дескриптора используют распределение откликов вейвлета Хаара в окрестности точки. Ключевым для ускорения и детекции и дескрипции является использование интегральных изображений [74]. Также размерность вектора признаков может быть уменьшена до 64, для ускорения сопоставления и повышения устойчивости, т.к. в этом случае дескриптор всё ещё выигрывает у SIFT по информативности.

SURF и SIFT до сегодняшнего дня являются актуальными для сопоставления. Вместе с уменьшением размерности для оптимизации вычислительных затрат было предложено квантование дескрипторов из вещественных к коротким целым, либо даже к бинарному виду [75] со сменой метрики на расстояние Хэмминга, для которого существуют высокопроизводительные реализации. Однако бинарные дескрипторы можно и специально проектировать. Однако бинарные дескрипторы можно и специально проектировать. В работе [76] предлагается эвристический дескриптор BRIEF, вектор признаков которого формируется из сравнений яркостей пикселей внутри размытого по Гауссу патча:

$$f(p_1, p_2) = \begin{cases} 1 : I(p_1) < I(p_2); \\ 0 : I(p_1) \geq I(p_2), \end{cases} \quad (1.10)$$

где I - изображение, p_1, p_2 - координаты пикселей. Выбор точек для сравнения происходит случайным образом Гауссовским распределением вокруг центра особой точки, последовательность выбирается и фиксируется. Такой дескриптор оказался устойчив к смене освещения и искажениям, связанным со смещением точки наблюдения (сдвиг, поворот по вертикальной оси), но крайне неустойчив к вращению плоскости изображения. Для метода BRIEF предложено множество улучшений, которые стали классическими и актуальны в использовании. В работе [77](ORB) вернулись к использованию ориентации точки для устойчивости, однако вместо патча изображения поворачивают последовательность точек. Помимо этого, авторы предлагают брать для формирования последовательности не все пары точек подряд, а добавлять по очереди так, чтобы их корреляция была меньше порога. В то же время (на той же конференции) представлена работа [78], где последовательность точек для бинарных тестов изменена со случайной на последовательно-концентрическую. Пары точек делят на 2 типа - «длинные» - для расчёта ориентации точки, и «короткие» - для построения дескриптора. Наконец, дескриптор FREAK [79] сочетает в себе наработки [77; 78] и представление о модели сетчатки (плотность точек в последовательности уменьшается по мере удаления от центра).

Однако аналитические наработки на базе тривиальных признаков (1.10) не позволили получить дескриптор, выигрывающий по различительной способности у дескрипторов SURF и SIFT. В то же время методы машинного обучения позволили продвинуться в получении более точных вещественных дескрипторов. В [80] обращают внимание, что многие существующие являются

комбинацией блоков, которые можно разбить на явные группы – сглаживание, преобразование (разница гауссиан и т.п.) и объединение (усреднение). Поэтому можно построить оптимальную комбинацию из блоков, а также подобрать для них оптимальные параметры. В результате удалось получить дескрипторы, превосходящие SIFT по точности более чем в 2 раза (по метрике TPR95) и в 4 раза более компактные. Единственным вопросом остаётся скорость вычисления самих дескрипторов – авторы отмечают, что снижение размерности это отдельный шаг.

В работе [81] авторы предлагают использовать машинное обучение, в частности AdaBoost [82], для формирования финального классификатора как линейной комбинации слабых классификаторов. В качестве слабых классификаторов предложено использовать градиентные признаки. Каждый такой признак на патче X параметризуется прямоугольной областью R , ориентацией градиента e и порогом T :

$$h(X; R, e, T) = \begin{cases} 1, \varphi(R, e) \leq T; \\ -1, \varphi(R, e) > T \end{cases} \quad (1.11)$$

где $\varphi(R, e)$ – функция выраженности градиента направления e в каждой точке патча. В силу дискретности изображения можно задать q направлений чтобы квантованное значение e принимало значение из множества $\left\{0, \frac{2\pi}{q}, \dots, (q-1)\frac{2\pi}{q}\right\}$. В таком представлении признаки легко считать с помощью всё тех же интегральных изображений. Классификаторы, получаемые с помощью такого обучения, существенно превосходят все прошлые наработки: дескриптор BinBoost размерностью всего 8 байт точнее SIFT не менее чем в 2 раза. При этом время вычисления такого дескриптора, по словам авторов, эквивалентно времени расчёта SIFT. Подход был улучшен с участием авторов в RFD [83], где бустинг заменили на жадный алгоритм выбора признаков, попарно наименее скореллированных. Обучение для признаков RFD проходит на уровне выбора порога T , что требует меньше данных для обучения, чем бустинг.

Современные обучаемые дескрипторы также используют схему бустинга. Дескриптор BEBLID [84] сначала обучается на слабых классификаторах с AdaBoost как вещественный, а затем упрощается до бинарного. Вычислительная эффективность достигается слабыми классификаторами формата парных тестов на квадратных областях (как в случае ORB). Ограниченность открытых датасетов приводит к перенасыщению и переобучению. Для тестирования вычислительной эффективности, точности и универсальности дескрипторов пред-

ложен набор критериев и дополнительный датасет непосредственно патчей изображений HPatches [85]. Критериями будут считать полноту и точность на следующих задачах – проверка похожести пары дескрипторов, сопоставление изображений, поиск дескриптора в большом наборе. Кроме того, показано, что нормализация аналитических дескрипторов может повысить их точность до обучаемых в сценариях реального использования.

Машинное обучение также породило множество нейросетевых дескрипторов. Основными архитектурами ИНС в данном случае являются сиамские и триплетные сети. При расчёте сетевых дескрипторов может быть добавлен полно связный слой, реализующий обучаемую метрику, вместо использования стандартных метрик Евклида и Хэмминга. Сиамская сеть MatchNet [86] технически состоит из пары сетей: патч преобразуется в вектор с помощью малопараметрической версии сети AlexNet, а метрическая часть состоит из 3 полно связных своём с нелинейностью и логистической функцией (softmax) на последнем слое. Полученная сеть показывает лучшие результаты чем SIFT, его сетевые аппроксимации и дескрипторы из работ [78–80], и достаточно близок по точности к BinBoost. В силу вещественной реализации и особенностей метрики, подобные дескрипторы проигрывают тому же BinBoost по вычислительной эффективности. Триплеты – тройки данных в обучающей выборке, которые состоят из основного патча «якоря»), его негатива (непохожего патча), и позитива (похожего патча). Триплетная функция потерь максимизирует расстояние от якоря к негативу, и минимизирует расстояние от якоря к позитиву. Для локальных дескрипторов такую функцию предлагается использовать, например, в работе с описанием TFeat [87]. Он реализован с помощью мелких свёрточных сетей и стратегии быстрого жёсткого отрицательного анализа. Сегодня попытки улучшить результаты дескрипторов на базе BRIEF продолжаются. В работе [88] предложена модификация ORB, в которой при построении дескриптора используются не бинарные, а тернарные тесты. Если в базовом решении на патче есть n пар «точек»:

$$(A_i, B_i), i \in [0..n - 1] : A_i > B_i \rightarrow 1 \text{ в } i\text{-м бите, иначе } 0 \quad (1.12)$$

то в новой работе авторы предлагают использовать n троек вида:

$$(A_i, B_i, C_i) i \in [0..n - 1] : A_i > B_i \& A_i > C_i \rightarrow 1 \text{ в } i\text{-м бите, иначе } 0. \quad (1.13)$$

Хотя в заключении авторы заявляют, что из версия эффективнее других дескрипторов в т.ч. SURF и SIFT, приведённые ими же графики не позволяют

согласиться. ORB-TPLCG не является самым эффективным ни по скорости (рис. 1.11), ни по соотношению точность/полнота(см. [88]).

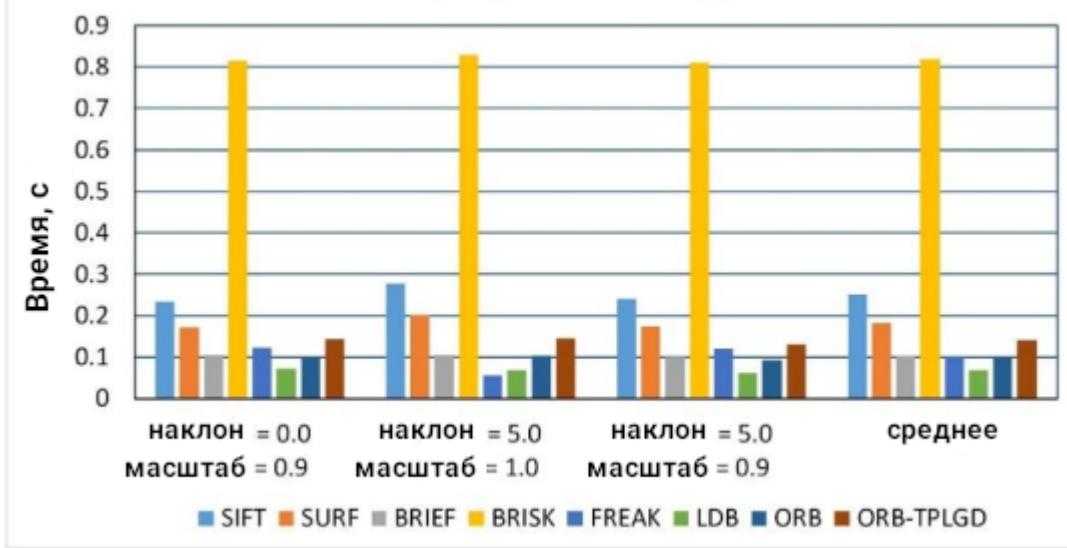


Рисунок 1.11 — Сравнение времени вычисления разных дескрипторов. Изображение из работы [88].

Детекторы, разработанные вручную, можно обучить путём случайной выборки в пространстве свёрточных фильтров. В работе [89] попытались найти оптимальный фильтр, используя стратегию обучения с учетом ограничений частотной области. Однако обучение использовалось только для надежного выбора признаков посредством обучения классификаторов, а не для их извлечения непосредственно из изображений. В работе [90] Key.Net объединяет ручные и обучаемые фильтры внутри неглубокой сети и предлагает лёгкий/эффективный обучаемый детектор. Ручные фильтры предоставляют якорные структуры для локализации, оценки и ранжирования повторяющихся признаков, которые подаются в обучаемые фильтры. В [91] свёрточная нейронная сеть D2-Net, которая также одновременно является дескриптором признаков и детектором признаков. Большая стабильность точек достигается за счёт переноса оценки координат на более поздний этап вместо начального. Модель обучали с использованием пиксельных соответствий, извлеченных из доступных реконструкций SfM, без дополнительной разметки. В [92] концепция D2-Net совершенствуется для повышения точности. ASFeat предложили исследовать информацию о локальной форме характерных точек. Вместо ожидания сложной деформации, заложенной в комбинации свёрток, предлагается использовать простые преобразования – проективное, аффинное и подобия.

1.4 Сравнение представлений. Поиск ближайших соседей

Результирующее количество векторов признаков зависит как от представления, так и от области приложения задачи классификации. В распознавании живописи количество классов можно ограничить текущей экспозицией (до 3500 полотен одновременно выставлено в Лувре), а вот поиск видеоконтента может требовать ориентирования в миллиардах признаков (коллекция Youtube ещё в 2006 году превышала 80 миллионов роликов). Если в первом случае перебор стабильно неэффективен, то во втором затраты трудоёмкости огромны и будут непрерывно увеличиваться. Использование индексирующих структур позволяет избегать полного перебора всех классов, с разменом времени на память. С помощью таких структур и методов их обхода можно получить усечённый список из кандидатов, ближайших к запросу по выбранной метрике.

Поиск с помощью древовидных структур – полноценное направление в методах поиска ближайших соседей. Одним из первых методов являются kd-деревья [93] – многомерная версия бинарных деревьев поиска. Широко используемая библиотека FLANN [94] – это улучшенная модификация именно kd-дерева и поиска по нему. Авторы предлагают рандомизировать выбор начальных вершин и использовать иерархическую кластеризацию, т.е. элементы кластеризуются по k-средних на каждом уровне дерева. В работе [95] предлагается использовать R-деревья. Дерево разделяет n-мерное пространство на n-мерными параллелепипедами, которым допустимо пересекаться между собой. С помощью этого обеспечивают размещение близких элементов в одну листовую вершину. В частности, новый объект попадёт в тот лист, параллелепипеду которого потребуется наименьшее расширение при добавлении.

В работе [96] предлагается использовать для поиска ближайших соседей локально-чувствительное хеширование (LSH, locality-sensitive hashing). LSH по сути инвертировано обычному хешированию, т.е. вместо «раскидывания» похожих элементов сильно отличающимися хэшами, хеш-функции подбирают таким образом, чтобы похожие объекты с высокой степенью вероятности попадали в один кластер. Если LSH может быть использован для дескрипторов с любой метрикой, то хеш множества индексов (МИН, multi-index hashing) в применении к изображениям [97] подразумевает работу с бинарными дескрипторами. Хешами в этом случае являются подстроки дескриптора. Деление дескрипто-

ра на участки, из которых берутся подстроки, определяет число хеш-таблиц и точность поиска. Заметим, что интересным свойством такого индекса является возможность точного быстрого поиска до определённого расстояния.

Той же командой авторов в 2011 предложен поиск IVFADC [98], заявленный как единственный метод для поиска по миллиардам векторов признаков. В пространстве поиска по всем дескрипторам рассчитывают диаграмму Вороного, т.е. разбиение на непересекающиеся регионы. Для разбиения предлагают использовать кластеризацию методом K-средних. Ключевыми для поиска являются центроиды регионов. Для каждого центроида хранится список векторов признаков, принадлежащих его кластеру (региону Вороного). Авторы предлагают для компактности хранить номер дескриптора и его разницу («невязку») с центроидом. При поиске соседей для конкретного запроса сначала фильтруются центроиды – выбирается несколько ближайших при сравнении квантованных значений. Затем списки векторов, принадлежащих соответствующим кластерам, объединяются и сравниваются с запросом в квантованном или полном (при неопределенности) виде. Эксперимент авторов, проведённый на открытой коллекции в 128 млрд. векторов показывает, что их метод точнее и быстрее FLANN. Позднее команда авторов, предложившая IVFADC, представила [99] ряд методов для поиска k-ближайших (как точного, так и приближенного) с оптимизированной схемой для эффективного параллельного вычисления.

В [100] приведён пример сетевого преселектора решения для поиска ближайших. Предложена свёрточная нейронная сеть со сквозным обучением для устранения неоднозначности совпадений признаков, которую в классическом случае решают использованием полулокальных ограничений. Сеть идентифицирует наборы пространственно согласованных совпадений путём анализа моделей консенсуса соседей в четырёхмерном пространстве всех возможных соответствий между парой изображений без необходимости использования глобальной геометрической модели. Показано, что модель можно эффективно обучать с помощью слабого контроля – в форме совпадающих и несовпадающих пар изображений – без необходимости ручного аннотирования соответствий между точками. В работе [101] авторы предложили совместную оптимизацию кодирования запросов и квантования продукта, чтобы сохранить эффективность моделей нейронного ранжирования при сжатии размеров индексов. Кодировщик запросов и индекс PQ обучаются совместно сквозным способом на основе трёх стратегий оптимизации, а именно: потери, ориентированной на ранжиро-

вание, оптимизации центроида PQ и сквозной отрицательной выборки. Авторы показали, что такая стратегия превосходит обучение на основе ошибки реконструкции для многих методов сжатия.

1.5 Пространственная верификация сопоставлений

Для оценки модели преобразования простым решением уравнения по минимально-достаточному количеству сопоставленных точек требуется фильтрация ошибок сопоставления. В общем случае допускается, что из-за локальностии признаков и маломерности пространства дескрипторов на сложных сценах и объектах возникнет нерелевантных, шумных точек, которые образуют ложные сопоставления. Подходы к оценке модели на зашумленных данных зависят ожидаемого характера шума. МНК и метод полных квадратов (МК) применимы для нормального и гауссова распределения шума соответственно. Для неизвестных распределений предлагаются подходы на базе робастных М-оценок, схем голосования и ограничения выборок [102].

Робастные М-оценки учитывают, что влияние выбросов следует уменьшать тем больше, чем сильней его невязка с моделью. Хьюбер [103] предложил обобщённое решение как минимизацию функционала:

$$\sum_i \rho(e_i(\theta), \sigma), \quad (1.14)$$

где σ - параметр масштаба (шума), $e_i(\theta)$ значение ошибки i -го сопоставления в модели с параметрами θ , и ρ функция невязки, которая отвечает требованиям:

$$\begin{aligned} \rho(x) &= \rho(-x); \quad \rho(0) = 0; \quad \forall x \rho(x) \geq 0; \\ |x_1| &\geq |x_2|, \rho(x_1) \geq \rho(x_2); \quad \forall x \exists \rho'(x) = \frac{\partial \rho(x)}{\partial x} \end{aligned} \quad (1.15)$$

Для ряда определённых распределений – Коши, Лапласа, нормального с зашумлениями [104] – предложены специализированные функции ρ , для которых на рассчитаны параметры, с которыми оценки достигают 95% точности. В случаях неизвестного распределения могут быть использованы либо более устойчивые в среднем оценки [105], либо последовательное итеративное приближение. Тогда при расчёте σ для (1.14) вводится набор весов w_i для $e_i(\theta)$ с некоторым

начальным приближением. Наиболее частым решением является МНК с итеративным пересчётом весов. Ограничение использования [102] при работе с сопоставлениями локальных признаков изображений обосновывается следующими моментами:

- конкурентность устойчивости и точности – чем устойчивей оценка, тем менее точна
- необходимость хорошего первого приближения

Кроме того, статистические методы подразумевают не более 50% выбросов, что не всегда выполняется в случае сопоставлений.

Использование схемы голосования для выбора геометрически согласованных признаков представлено в [72]. Автор кластеризовал с помощью преобразования Хафа сопоставленные точки, для которых кроме координат определены размер и угол-ориентация. Кластеры при таких параметрах задают варианты только для преобразования подобия как грубой аппроксимации реального преобразования. Точное преобразование строится с помощью МНК по точкам из выигравшего кластера. Преимуществом подобного подхода является высокая устойчивость к шуму – автор заявляет, что удалось достигнуть обнаружения для объектов, представленных 1% от всех локальных признаков. К недостаткам относится во-первых необходимость использовать алгоритмы детекции точек, дающие ёмкие данные – недостаточно только координат и относительного размера, и во-вторых пересчёт в точное решение из достаточно грубого приближения.

Базовый метод на основе неполных данных RANSAC (RANdom SAmple Consensus) [106] предлагает итерационную процедуру оценки параметров и вводит следующие термины:

- гипотеза – параметры модели, полученные на одной из итераций;
- консенсус – оставшиеся данные (не из выборки) тестируются на соответствие гипотезе с заданной точностью и, в итоге, делятся на те, которые подтверждают модель (далее - инлаеры (inliers)), и те, которые опровергают её (выбросы);
- остановка – прекращение итерационной процедуры по внутренним критериям (число итераций либо достаточное число подтверждающих данных для модели).

Общая процедура алгоритма RANSAC:

1. Сформировать случайную выборку из входного набора данных.

2. По выборке вычислить параметры текущей гипотезы.
3. Для рассчитанной гипотезы проверить оставшиеся данные с модельной функцией потерь, порождая наборы инлаеров и выбросов.
4. Проверить условия остановки и вернуться к 1 либо выдать набор-кансус.

Для i -го сопоставления, соответствие модели оценивается с помощью функционала ошибки ρ и порога \mathcal{T} :

$$\rho(e_i^2(\theta)) = \begin{cases} 0, & \text{если } e_i^2(\theta) < \mathcal{T}^2, \\ const, & \text{иначе,} \end{cases} \quad (1.16)$$

где $e_i(\theta)$ значение ошибки i -го сопоставления в модели с параметрами θ .

Значение функционала \mathcal{C} рассчитывается как сумма ошибок:

$$\mathcal{C}(\theta) = \sum_i \rho(e_i^2(\theta)). \quad (1.17)$$

Пусть \mathcal{P} - это вероятность появления выборки без выбросов. Обозначив как ε вероятность выбора инлаера из входного набора и \mathcal{M} минимальный размер выборки, без ограничений для неё и допуская её вырожденность, получим вероятность независимой выборки из инлаеров, $\varepsilon^{\mathcal{M}}$. Вероятность не получить нужную выборку за \mathcal{I} итераций получается $(1 - \varepsilon^{\mathcal{M}})^{\mathcal{I}}$, или $1 - \mathcal{P}$. Таким образом теоретическая минимальная оценка \mathcal{I} числа необходимых итераций для получения с вероятностью \mathcal{P} выборки из инлаеров:

$$\mathcal{I}^* \geq \frac{\log(1 - \mathcal{P})}{\log(1 - \varepsilon^{\mathcal{M}})}. \quad (1.18)$$

С момента публикации оригинального метода было предложено множество различных модификаций для улучшения точности, скорости и устойчивости, а также варианты их обобщения.

1.5.1 Модификации RANSAC с улучшениями отдельных этапов

RANSAC может быть чувствительным к выбору правильного порога шума, который отделяет инлаеры от выбросов. Если порог слишком велик, то все

гипотезы оцениваются одинаково (хорошо). С другой стороны, если порог слишком мал, инлаеров может быть недостаточно для получения устойчивых параметров модели. В работе [107] предложены модификации MSAC (М-оценка) и MLESAC (максимальное правдоподобие), которые уменьшают эту чувствительность. Авторы предложили оценивать качество консенсуса по вероятности (в исходной работе [106] оценивают только мощность набора).

В работе [108] с описанием LO-RANSAC (локально оптимизированный), ключевое наблюдение состоит в том, что минимальные выборки могут усилить основной шум и привести к появлению гипотез, далёких от истины. Эта проблема решается путём введения процедуры локальной оптимизации при достижении наилучшей модели. В исходной статье локальная оптимизация реализована как итерационный процесс аппроксимации методом наименьших квадратов с уменьшающимся порогом выброса внутри RANSAC.

Модификация Graph-Cut [109] локальной оптимизации включает новую оценку модели. Оценка основана на идее формулирования задачи выбора инлаеров как задачи минимизации энергии с учётом расстояния между точками. Авторы предлагают на этапе локальной оптимизации (для лучшей в моменте гипотезы) пересчитывать инлаеры с помощью алгоритма рассечения графа. Под графом понимают граф соседей, с рёбрами между точками, расположеными на расстоянии не больше порога. Авторы предлагают для эффективного использовать методы поиска ближайших соседей, описанные ранее.

В работе [110] (MAGSAC) предложен подход к избавлению от пользовательского порога на инлаеры, названный σ -консенсус. Рассматривая шум, как случайную величину с функцией плотности, предлагается не оценивать его напрямую, а маргинализировать по его же диапазону масштабов. Авторы заявляют, что даже единичный расчёт σ -консенсуса на финальном этапе всегда даёт более точную модель для широкого спектра задач без замедления.

PROSAC [111] устанавливает ранг «ожидания» сопоставленных пар точек и гипотез в соответствии с некоторой мерой качества данных. По мере выполнения процедуры проверки гипотез, уверенность в адекватности показателей качества снижается, и стратегия порождения выборок смещается в сторону исходного RANSAC. Более ожидаемые выборки вылезают на более ранних этапах, но далее постепенно включаются данные с более низкими оценками. Авторы заявляют, что достигают значительного ускорения по сравнению с RANSAC

(порядка сотен раз), благодаря тому, что хорошие гипотезы генерируются на ранних этапах.

В [112] приведена сетевая реализация (NG-RANSAC), который использует априорную информацию для поиска гипотез, чтобы повысить вероятность гипотезы без выбросов или с малым количеством. В несетевых вариантах априорная информация получается эвристическими методами, по рукотворным дескрипторам, разработанным с учётом знаний исследователя. Самоконтроль процесса достигается с использованием доли инлаеров как части обучающих данных, а добавление дифференцируемой версии RANSAC позволяет осуществлять дальнейшие улучшения. Эксперименты по оценке фундаментальной матрицы, перемещения камеры и оценке линии горизонта дают самые современные результаты.

1.5.2 Модификации RANSAC с обобщением улучшений

К 2012 было предложено множество независимых улучшений разного рода. В работе [113] их объединили в единый модульный подход, который включал накопленные практические соображения и вычислительные оптимизации. Структура USAC состоит из 5 шагов:

1. префильтрация
2. генерация и проверка сэмпла
3. генерация и проверка гипотезы
4. оценка гипотезы
5. уточнение гипотезы (локальная оптимизация)

Последующие работы преимущественно рассматривают именно обобщённые решения как базовые, модифицируя одновременно в несколько модулей для достижения улучшений по нескольким критериям.

В работе [114] (MAGSAC++) предлагается ускорение расчёта σ -консенсуса, выполняя маргинализацию распределения шума с помощью введения метода итеративно перевзвешиваемых квадратов. Помимо этого, предложен генератор выборок P-NAPSAC. Предложенные улучшения при тестировании объединены авторами с возможностями USAC.

В [115] (VSAC) предложено улучшение структуры USAC с учётом новых исследований. Авторы внедряют концепцию независимых инлаеров для более эффективной фильтрации ложных гипотез. Независимые инлаеры – это выборка, порождающая параметры модели, плюс подвыборка над всеми инлаерами в контексте пространственной независимости – точка помечается независимой, если достаточно удалена от уже существующих независимых. Таким образом сохраняется точность MAGSAC++, а скорость повышается в 2 раза.

Для модификации Graph-Cut в работе [116] проведены расширенные эксперименты, на базе MAGSAC++/USAC.

В работе [117] предложен генератор выборок с нейроподдержкой, в котором вероятности принадлежности данных к инлаерам определяются с помощью сети и пересчитываются при каждой неудачной итерации. Кроме того, предложено использовать дополнительные параметры локальных признаков (масштаб и угол) для повышения надёжности на сложных сценах.

Последние обзоры показывают [118], что исследования SAC-методов продолжаются – как общего направления, так и касающиеся только анализа изображений, либо даже отдельных структур и типов объектов.

1.6 Иные методы сопоставления

Понятие инлаера существует не только в контексте консенсусных методов и гипотезы модели. В ряде методов авторы формируют саму гипотезу как набор инлаеров – строгих соответствий между точкам – для дальнейшей оценки параметров по ним. Так, в работе [119] предложено выделять такие строгие соответствия между двумя наборами локальных признаков (особых точек и дескрипторов любого типа). Соответствия оцениваются путём решения дифференцируемой задачи оптимального транспорта, стоимость которой прогнозируется с помощью графовой нейронной сети. В [120] предлагается строить инлаеры без явного этапа выделения и дескрипции точек. Авторы определяют с помощью архитектуры Transformer глобальное рецептивное поле, благодаря которому совпадения можно искать в т.ч. между областями с невыраженной структурой. В работе [121] отмечают, что сегменты – мощный признак, дополняющий точки. Однако создание устойчивых дескрипторных описаний и их сопоставление в

случае сегментов существенно более сложно, чем для точек. Проблемы возникают из-за частичного заслонения, слабо выраженной либо периодичной текстуры вокруг сегмента. Авторы предлагают графовую нейросеть, которая извлекает комплексное представление из изображений. При сопоставлении точки и прямые обрабатываются независимо друг от друга. В работе [122] представлена система поиска печатных документов, которая использует как текстовые функции, так и локальные признаки (малобитные). Текст обнаруживается с помощью алгоритма, основанного на максимальном стабильных экстремальных областях с расширенными возможностями. Патч изображения текста заголовка исправляется с использованием алгоритма на основе градиента и распознаётся с помощью оптического распознавания символов. Локальные признаки извлекаются из изображения запроса. Текст заголовка используется для поиска, а характеристики используются для сравнения изображений. Совместное использование текста и признаков позволило значительно уменьшить размер общего представления. В работе [123] показывается, что сочетание текстовых и яркостных признаков значительно выигрывает у поиска по отдельно яркостных признакам. Авторы справедливо отмечают, что текст – высокоуровневый признак, который улучшает понимание содержимого изображения. Также авторами показано, что в этом случае важнее полнота детекции тексте, нежели f-score. В работах [124; 125] предлагается сочетать текстовые и яркостные признаки для улучшения классификации. Вектор признаков изображения извлекают с помощью нейросетей типа AlexNet, слова извлекают с помощью сети, после чего кодируют векторным кодированием Фишера, которое фиксирует морфологию текста.

1.7 Выводы. Постановка задачи

В первой главе рассмотрены существующие методы выделения, представления, комбинирования и сопоставления разных типов признаков, разработанные для решения задачи выделения и классификации объектов на изображении. Для каждого направления методов и каждого типа признаков прослеживается переход: от ранних аналитических методов, параметризуемых вручную, к нейросетевым методам, которыми преимущественно представлены современ-

ные решения. Несмотря на высокие показатели точности на отдельных наборах данных, нельзя сказать, что задачу классификации и локализации можно считать решённой даже в применении к группе плоских ригидных объектов (хотя часто заявляются более общие цели).

Аналитические методы отличаются скоростью и точны в контексте пиксельной точности. Такое выделение границ и точек устойчиво к точке съёмки и перепаду яркостей. В то же время точность таких дескрипторов существенно меньше, чем у обученных версий, а при выделении велико влияние шума и сцены. В процессе решения задач, как правило, оперируют признаками (как геометрическими, так и дескрипторными) одного типа. В то же время многократно отмечено преимущество мультимодальной классификации (совмещение текста и дескрипторов изображения) над одномодальной любого типа.

В свою очередь нейросетевые модели и методы могут достигать превосходного качества. Однако, в большинстве случаев, их исполнение требует больших ресурсов, чем может предоставить даже самое современное мобильное устройство. Таки образом пользовательское решение, основанное на таких методах, будет ограничено в автономности использования из-за зависимости от соединения с исполняющим сервером. Помимо этого, следует учитывать и проблемы обучения. Касательно классификации и локализации ригидных объектов на изображении, наблюдаются следующие проблемы:

1. Необходимые объёмы данных. Для точной классификации репрезентативные данные нужно собрать, что не всегда возможно. (В отличие от классификаций по категориям, где изображение недоступного должно и может быть аппроксимировано изображениями доступных.)
2. Обновление классов. Каждое добавление класса требует дообучения, при ряде удалений оно также желательно для сохранения точности.
3. Число классов. Реальные задачи точной классификации могут оперировать сотнями и тысячами классов. Большинство архитектур не способно сохранять точность при таком расширении, либо расширение сопряжено с ростом вычислительных и временных затрат.

В завершении можно отметить, что разрабатываемые методы имеют весьма общий характер. Это расширяет сферу их применимости, но не даёт основы для поиска и использования специфических свойств идентифицируемых объектов. Целью данной работы является улучшение классификации и локализации изображений плоских четырёхугольных объектов. В качестве модельных объектов

будем рассматривать документы (ID-документы в частности) и произведения 2D-искусства (картины разных жанров).

Для достижения поставленной цели решим следующие задачи:

1. Разработать методы быстрого устойчивого выделения глобальных признаков плоского ригидного объекта.
2. Разработать метод понижения размерности при классификации с учётом признаков разного типа (глобальных и локальных, дескрипторы и текст).
3. Разработать метод классификации и локализации плоских ригидных объектов с учётом природы объектов и способов съёмки.

Глава 2. Выделение признаков с учётом структуры объекта и модели формирования изображения

2.1 Формирование изображения. Камера обскура.

Базовой моделью формирования изображения при разработке методов анализа изображений является модель камеры-обскуры. В данной модели точка x на плоскости изображения образуется из точки X пространства по преобразованию $x = KAX$. K – калибровочная матрица камеры для случая квадратного пикселя, A – матрица параметров позы камеры:

$$K = \begin{pmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{pmatrix} \quad A = R|c, \quad (2.1)$$

где f - фокусное расстояние, p - принципиальная точка, R - матрица поворота 3×3 , c - вектор сдвига 3×1 .

На рисунке 2.1 показан пример формирования изображения документа, где O - центр системы координат, f - фокусное расстояние, I - изображение, P - принципиальная точка, проекция O на плоскость изображения. Углы документа $A_0B_0C_0D_0$, заданные в 3-мерных координатах, переходят в 2-мерные точки $ABCD$ изображения.

Такая модель определяет только проективное искажение, исключая из рассмотрения аберрации сложных оптических систем (например, радиальную дисторсию, порождаемую кривизной линзы). Геометрические аберрации могут быть компенсированы как предварительной калибровкой камеры, так и автоматически на уровне программируемых решений, вариант которого рассмотрен в [126]. Поэтому камера-обскура достаточно хорошо аппроксимирует сканеры и малые цифровые камеры и достаточна для решения задач, не связанных с изображениями ультра-высокого разрешения.

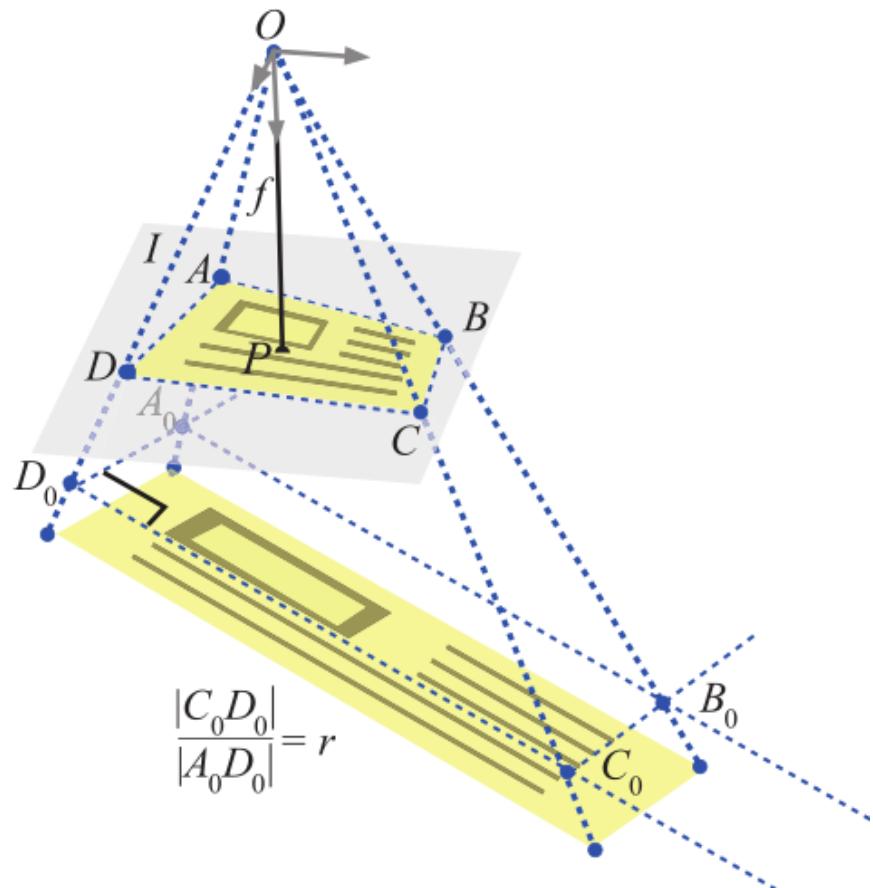


Рисунок 2.1 — Формирование изображения в модели камера-обскура на примере документа.

2.2 Методы выделения четырёхугольников

В отличие от работ, посвящённых задачам выделения примитивов на произвольных изображениях, нас интересуют не все допустимые линии и четырёхугольники, а соответствующие границам объекта либо близкие к ним. Наиболее релевантными по сути являются работы [58; 59] – авторы так же опираются на модели изображений и объектов. При построении методов будем учитывать следующие моменты:

- границы объекта могут быть частично заслонены;
- существуют квази-четырёхугольные объекты, которые так же являются объектом интереса.

При построении решения это означает, что нельзя полагаться на детекцию углов как базисных примитивов, поскольку потеря детектором 1 угла может быть критичной для метода.

Выделим некоторые варианты условий, при которых может решаться задача:

1. известные и строгие ограничения
 - весь объект располагается внутри кадра
 - объект занимает значительную часть кадра
 - известны линейные размеры либо соотношение сторон объекта
2. неизвестные / нестрогие ограничения (размеры объекта и положение внутри кадра заданы широким диапазоном значений либо неопределены)

Исходя из данных ограничений были разработаны 2 метода, оба включают следующие общие этапы:

- построение полутоновой карты границ
- использование преобразования Хафа для выделения линий
- пересечение линий, маркировка полученных углов и построение четырёхугольников-кандидатов
- отбрасывание кандидатов по геометрическим критериям

2.2.1 Поиск четырёхугольников в строгих ограничениях

Описанные ограничения на размещение объекта в кадре вполне естественны для задач распознавания и легко реализуемы при взаимодействии с пользователем: на 2.2 они выражены в формате областей интереса (далее - регионы, regions of interest, ROI), которые на кадре видны пользователю. Каждый регион должен содержать по одной стороне объекта. Одновременно, регулируя только размеры ROI, можно выразить дополнительное ограничение на наклон и сдвиг сторон.

Воспользуемся допущенными ограничениями. В предложенном методе используются ROI, представляющие собой прямоугольные зоны изображения со сторонами, параллельными осям координат. Каждый ROI содержит только одну сторону объекта целиком, углы лежат внутри пересечений регионов. Ограничение наклона, обеспечиваемое ROI, означает, что искомые границы можно считать ортотропными, т.е. имеющими сравнительно небольшое отклонение от горизонтального или вертикального направления. Совокупность таких ограни-



Рисунок 2.2 – Варианты ROI и их отображения в кадре.

чений позволяет искать каждую сторону независимо внутри соответствующего региона, и упрощает некоторые этапы этого поиска.

Этапы метода ниже описаны для ROI, содержащего верхнюю часть объекта. Для остальных трёх ROI процесс идентичен, с точностью до поворота на 90 градусов.

1. Расчёт начальной карты границ. Рассчитаем производную изображения в регионе, предварительно подавив шум с помощью высокочастотного фильтра (например, фильтра Гаусса). Для поиска горизонтально-ортотропных границ нам нужно только изменение вдоль оси OY. Такую производную, в отличие от полного градиента Di Zenzo [35], можно достаточно просто рассчитать на цветном изображении:

$$G(x,y) = \frac{1}{3}(I'_R(x,y) + I'_G(x,y) + I'_B(x,y)) \quad (2.2)$$

- производная для цветного изображения (RGB), где $I'(x,y)$ - производная для 1 канала, рассчитывается по формуле (1.3).

На рисунке 2.3 показано, как граница, выраженная на цветном изображении, исчезает на изображении, преобразованном в полутонаовое.

2. Фильтрация карты границ. Анализ реальных данных выявил огромный класс случаев, когда фон сцены изображения плотно прорисован и имеет множество краёв. Эти края часто острее, чем настоящие края объекта. Эту особенность назвали «эффектом ковра», имея в виду первые проанализированные примеры. Такие случайные края возле границы объекта в основном встречаются на фоновой части изображения. Для каждого пикселя границы предложено рассчитать коэффициент близости границ $d(x,y)$ как расстояние до ближайших

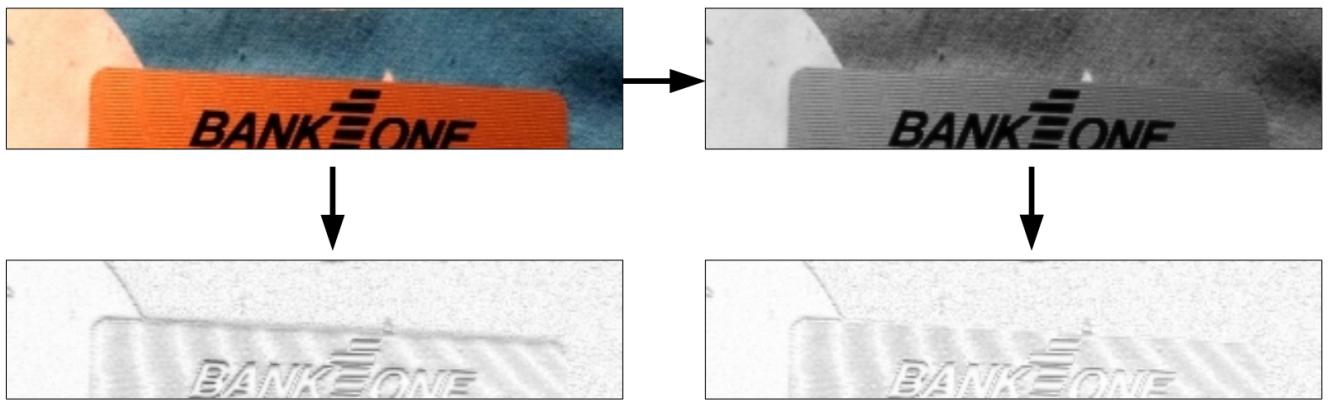


Рисунок 2.3 — Отличие градиента, рассчитанного на цветном и сером изображении.

границ, расположенных выше и ниже (для границ, вытянутых вдоль оси OX).

$$\forall G(x, y) > 0 : \quad d(x, y) = \frac{(y_2 - y_1)}{h_r} \quad (2.3)$$

$$G(x, y_1), \quad G(x, y_2) > 0, \quad y_1 < y, \quad y_2 > y;$$

Если сверху или снизу нет границ, используется расстояние до верхней или нижней границы области интереса. При нормировании на высоту ROI h_r получается коэффициент от $(0, 1.0]$ для корректирования значения каждого граничного пикселя. Поскольку сцена с «ковром» расположена только с одной стороны от границы объекта, данный коэффициент будет в среднем выше для истинной границы, чем для побочных границ, обусловленных фоном. Изменения при фильтрации показаны на рисунке 2.4.



Рисунок 2.4 — Выделение границ в регионе.

3. Фильтрация связных границ. Собираем граничные пиксели в связанные компоненты, а затем применяем пороговую обработку к этим компонентам вместо отдельных пикселей. Ортотропия границ позволяет выполнять поиск связанных компонентов за один проход через область интереса вдоль предпочтительного направления границы [127]. По сравнению с отдельными пикселями, компоненты предоставляют больше свойств для отсечений, например яркость (минимальная/средняя/максимальная) и длина. В моей реализации используется фильтрация по длине компоненты и средней яркости.

В результате получается полутоновая карта границ, где интенсивность каждого пикселя в компоненте одинакова и представляет собой средний контраст соответствующего края.

4. Поиск прямых линий. Чтобы найти прямую линию, соответствующую границе объекта, карта границ преобразуется БПХ [45]. Алгоритм БПХ по [45] обрабатывает независимо четыре угловых диапазонах по 45 градусов для расчёта весов всех линий. В случае региона с ортотропной границей рабочим будет только диапазон $[-45, 45]$ градусов, и исключение неиспользуемого диапазона позволяет в 2 раза снизить вычислительную сложность. Более того, для изображения $w \times h$, где $w > h$, требуется меньший диапазон углов. Предлагается, угол объекта лежит в пересечении двух ROI. Тогда граница объекта пересекает зону угла размером $h \times h$ слева и справа. При сжатии отфильтрованного изображения карты границ по ОХ так, чтобы $(w - 2h) \times h \rightarrow h \times h$, диапазон углов касания целевой линии составит $[-45, 45]$.

В итоге сложность преобразования будет $O(w \cdot h + h^2 + 2h^2 \log(h))$ операций для изображения $w \times h$, ($w > h$) с учётом масштабирования (вместо $O(4w \cdot h \log(h))$).

Недостаточно выбрать одну лучшую прямую в пространстве Хафа, поскольку допустимо, что даже на отфильтрованной карте границ самая сильная линия соответствует внутренней части документа или фона сцены. Для преодоления этого предложена следующая итерационная процедура:

1. Карта границ преобразуется БПХ, и наиболее сильный кандидат границы объекта выбирается как линия с наивысшей оценкой Хафа.
2. Подавляем окрестность линии на отфильтрованной карте.
3. Если нужно больше линий-кандидатов, возвращаюсь к шагу 1.

Следуя этой процедуре, можно получить достаточно линий-кандидатов, которые будут достаточно разными за счёт этапа подавления, т.е. «стирания» учтён-

ных границ с карты. Оценка Хафа присваивается кандидату как вес, который будет использоваться на следующих этапах.

5. Составление четырёхугольника. На данном этапе для каждого ROI образован набор линий, составляющих множество $L = \{l : a, b, c, w\}$, где w - вес прямой. Составление всех возможных четырёхугольников попарным пересечением прямых требует n^4 операций для $|L| = n$. Пусть $L_q \subset L$ образуют четырёхугольник $q = \{q_i\}_{i=1}^4$. Тогда в качестве базового веса w_q четырёхугольника принимается сумма составляющих весов линий:

$$w_q = \sum_{l \in L_q} w_l \quad (2.4)$$

Линии истинных границ объекта не должны выходить за пределы четырёхугольника, поэтому такие комбинации линий следует подавлять. Штраф p рассчитывается для угла четырёхугольника $q_i : l_a \cap l_b \rightarrow q_i$ как сумма значений K ближайших пикселей вне точек пересечения сторон:

$$\begin{aligned} p(q_i) = p(x, y) = \sum_{j=1}^K G((x, y) + j \cdot \vec{l}_a) + G((x, y) + j \cdot \vec{l}_b), \\ |\vec{l}_a| = |\vec{l}_b| = 1 \end{aligned} \quad (2.5)$$

При расчёте штрафа по (2.5) единичные вектора \vec{l}_a, \vec{l}_b вдоль прямых направлены из центра относительно q_i .

6. Восстановление четвёртой стороны по трём известным. Предположим, что истинная прямая (либо) сегмент для одной из сторон не обнаружен – край объекта выходит за кадр, либо недостаточно выражен на карте границ. Тогда пусть четырёхугольник $ABCD$ является проекцией прямоугольника R с соотношением сторон r . Известные стороны заданы отрезками AD, AB' и DC' в трёхмерной системе координат камеры с фокусным расстоянием f . Известно, что точки B и C лежат в той же полуплоскости что B' и C' относительно прямой, заданной отрезком AD . Необходимо восстановить положение точек B и C .

Для решения этой задачи необходимо, для начала, восстановить направления на обе точки схода (см. 2.5). Для этого вычислим положение первой точки схода: $V_1 = AB' \cap DC'$. Для второй точки схода V_2 известно: $V_2 \in AD$, $OV_1 \perp OV_2$ (поскольку объект прямоугольный).

Поэтому: $V_2 = AD \cap \beta$, $\beta \perp OV_1$ (в общем случае, когда $D \notin \beta$ – иначе V_2 в бесконечности на AD).

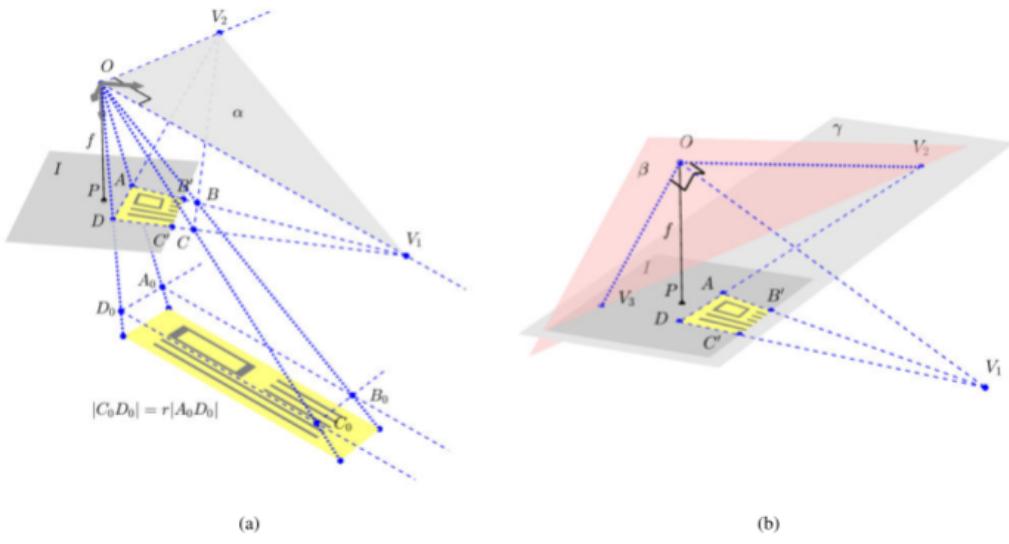


Рисунок 2.5 — Восстановление точек схода и стороны.

Вектора OV_1 и OV_2 не только задают плоскость α , параллельную плоскости прообраза, но и формируют ортогональный базис объекта, т.е. параллельны его сторонам. Воспользуемся этим для восстановления четырёхугольника $A_0B_0C_0D_0$, подобного прообразу R . Зафиксируем точку D_0 : на луче $D_0 \in OD$, так что $D_0 > D > O$. $A_0 : A_0D_0 \in D_0OV_2$, $A_0D_0 \parallel OV_2$

$A_0 = OA \cap A_0D_0$ (т.е. A_0 на прямой, проходящей через точку D_0 в направлении OV_2).

$$C_0 : C_0D_0 \in D_0OV_1, C_0D_0 \parallel OV_1$$

$$C_0D_0 \perp A_0D_0$$

$|A_0C_0| = r \cdot |A_0D_0|$ (т.е. C_0 на прямой, проходящей через точку D_0 в направлении OV_1 и в одном полупространстве относительно плоскости OAD , что и точка C'). Положение точки B_0 определяется аналогичным образом. Искомые точки B и C являются проекциями точек B_0 и C_0 на плоскость изображения I . Если найденная сторона лежит внутри изображения, то можно «честно» оценить её вес по карте границ. В ином случае считаем, что её вес меньше, чем вес худшей из прямых в данном ROI (для методов, описанных далее – в вертикальной либо горизонтальной ориентации).

7. Фильтрация по согласованию параметров геометрии. Не все четырёхугольники, полученные алгоритмом, могут оказаться проекцией прямоугольника R с фиксированным соотношением сторон r . Установим для каждого прообраз и соответствие параметров прообраза идеалу включим в оценку кандидатов. Известно, что любой тетраэдрический угол имеет только одно (с точностью до гомотетии) сечение $A'B'C'D'$, имеющее вид параллелограмма [128] и

оно всегда существует. Множество выпуклых четырёхугольников $\{Q\}$ на изображении будем рассматривать в модели камеры-обскуры для камеры с известными внутренними параметрами - фокусным расстоянием и принципиальной точкой (см. рис. 2.1). В центрально-симметричной модели [129] изображения образуется тетраэдрический угол с вершиной O в центре объектива камеры и сечением $ABCD$, образуемым плоскостью изображения и пересекающим катеты угла в углах полученного четырёхугольника. Таким образом построенного параллелограмма $A_0B_0C_0D_0$ (см. 2.5) достаточно для сопоставления с прообразом. Обратите внимание, что для тестируемой четырёхугольной альтернативы может не быть прямоугольного прообраза. Поэтому наряду с соотношением сторон учитываются также отклонение углов:

$$\begin{aligned} r' &= \max \left\{ \frac{A_0B_0}{B_0C_0}, \frac{B_0C_0}{A_0B_0} \right\} \quad r \cdot t_l < r' < r \cdot t_r \\ \alpha &= \angle A_0B_0C_0 = \angle V_1OV_2 \quad |\alpha - 90^\circ| < t_a, \end{aligned} \quad (2.6)$$

где r — известное соотношение сторон объекта; t_l, t_r, t_a — параметры алгоритма.

Помимо фильтрации нереализуемых четырёхугольников, веса оставшихся корректируются с учётом отклонения от прообраза. Коэффициент несоответствия прообразу определён как:

$$P_r = \frac{|r' - r|}{\max\{r, r'\}} \quad (2.7)$$

Таким образом, с учётом (2.5), финальный вес четырёхугольника q :

$$\begin{aligned} C(q) &= \frac{w_q \cdot (1 - P_r)}{1 + \sum_i (1 - c(q_i, q_{i+1}))} - \sum_i p(q_i) \\ c(q_i, q_{i+1}) &= \frac{1}{|q_i - q_{i+1}|} \sum_{(x,y) \in [q_i, q_{i+1}]} G(x,y) \end{aligned} \quad (2.8)$$

Метод обозначен в сравнении и далее как **АЧ1**.

2.2.2 Модификация 2.2.1 в более слабых ограничениях

Можно предположить, что на положение объекта наложены более слабые ограничения: ROI отсутствуют, но противоположные стороны либо «преимуще-

ственno горизонтальныe» (диапазон тангенса угла наклона $[-1; 1]$), либо «преимущественно вертикальныe». Такое допущение учитывается на всех этапах расчёта четырёхугольников. Метод модифицируется следующим образом:

1. Карта границ. В результате извлечения рёбер получаются две карты границ: с «преимущественно горизонтальными» и «преимущественно вертикальными» краями соответственно. Затем оба изображение размываются фильтром Гаусса вдоль направления градиента.

2. Выделение прямых. Изображение с вертикальными границами (условно те, которые вдоль длинной стороны изображения) перед поиском линий делится на три равные части по горизонтали. Таким образом поддерживается качество обнаружения прямых на коротких отрезках. Для поиска линий-кандидатов также используется БПХ. Затирание с пересчётом БПХ на полном изображении слишком затратно, поэтому в данном случае имеет смысл вычислить множество прямых за один проход. Для этого на каждом изображении пространства Хафа:

1. Определяются точки - локальные максимумы, затем сортируются по весу
 2. Начиная с наибольшего веса, точка определяет прямую-кандидат, если она не лежит в окрестности ранее добавленных точек (т.е. более значимых)
- В результате было получается K прямых для горизонтального БПХ-изображения и $3K$ для вертикального БПХ-изображения (по K с каждой части).

3. Построение четырёхугольников. Выполняется построение пар линий (вертикальных и горизонтальных), а затем их пересечения для формирования четырёхугольников. В данном случае нет предварительной информации, какую из сторон объекта формирует прямая, поэтому четырёхугольник-кандидат дополнительно проверяется на выпуклость.

Остальные проверки и ранжирование выполняются идентично заявленному в разделе 2.2.1.

2.2.3 Поиск четырёхугольников в неизвестных ограничениях

Для неизвестных ограничений предлагается следующий метод:

1. Предварительная обработка изображения. Изображение масштабируется, а мелкие элементы, такие как буквы и шум, удаляются использованием морфологического закрытия.

2. Выделение сегментов. Карта границ строится для всего изображения, как предложено в части 2.2.2. На карте границ нужно выделить прямолинейные сегменты. При этом будет сначала использоваться её бинарное представление, т.е. только информацию о принадлежности пикселя границе [130]. Рассмотрим последовательный набор точек контура, в окрестности каждой точки которого гипотеза о прямолинейности контура признана верной. Чтобы проверить гипотезу, нужно вычислить ковариационную матрицу для каждой точки контура $\{(x_i, y_i)\}, i \in [0, N]$ в окрестности фиксированного размера l :

$$\begin{aligned} X &= \{x_k\}, \quad Y = \{y_k\}, \quad i = [i - l, i + l] \\ \Sigma(X, Y) &= \begin{pmatrix} DX & cov(X, Y) \\ cov(X, Y) & DY \end{pmatrix} \\ cov(X, Y) &= M[XY] - MX \cdot MY \end{aligned} \tag{2.9}$$

Собственные векторы и собственные значения позволяют оценить распределение точек на каждом участке контура путем его аппроксимации эллипсом e . Гипотезу о прямолинейности текущего участка контура будет считаться верной, если соотношение длин осей эллипса больше порога: $e_x/e_y > tr$

Поскольку использование только бинарного изображения может привести к пропуску некоторых частей правой границы, дополнительно [131] используется БПХ на изображении с серыми границами и детектор линейных сегментов [50]. Сегменты, полученные в результате преобразования Хафа и анализа контуров, используются вместе. Важно отметить, что они настроены не на то, чтобы работать с максимальной эффективностью по отдельности, а на то, чтобы дополнять друг друга.

Каждому сегменту присваивается вес, равный его длине, и сегменты, лежащие на одной линии, объединяются. Два отрезка считаются лежащими на одной прямой, если максимальное расстояние от конечных точек одного отрезка до линии, содержащей другой отрезок, меньше порогового значения. Объединённый сегмент имеет концы в крайних концах этих сегментов, а его вес равен сумме их весов. Использование суммы весов позволяет уменьшить вес объединённого сегмента, состоящего из двух удалённых непересекающихся частей, и увеличить вес сегмента, полученного при одновременном использовании

двух алгоритмов. Не более чем K наибольших по весу отрезков остаются для построения четырёхугольника объекта.

3. Построение графа пересечений. Для построения четырёхугольника по найденным отрезкам строится граф (граф возможных пересечений). Вершины данного графа соответствуют найденным отрезкам, ребра — точкам пересечения прямых, содержащих эти отрезки. Для построения графа все сегменты делятся на преимущественно вертикальные и преимущественно горизонтальные в зависимости от углов, которые они составляют с осями изображения. Если отрезок можно отнести к обоим классам (диагональный), то вместо него добавляются два отрезка: горизонтальный и вертикальный (например, левый отрезок D на рис. 2.6а соответствует двум вершинам DH и DV на рис. 2.6б). Также каждой вершине присвоен вес w_v , равный весу соответствующего сегмента.

Ребра графа строятся как пересечения всех горизонтальных отрезков со всеми вертикальными. Если предположить, что граф построен на N сегментах, то весь граф содержит $O(N^2)$ рёбер. Каждое ребро в графе классифицируется в соответствии с типом угла документа, который может быть соответствующей точкой пересечения (левый верхний LT, правый нижний RB и т.п.). Ребра классифицируются относительно положения точки пересечения сегментов, которые ее создали. Когда точка является точкой пересечения самих сегментов, а не только линий, на которых они лежат, она может соответствовать произвольному углу (LTRB) или, например, когда она лежит слева от горизонтального сегмента и на вертикальном сегменте это может быть левый верхний или нижний левый (BLT). Каждому ребру присваивается вес w_e , равный минимальному расстоянию между концами отрезков, образующих соответствующую точку пересечения.

Построенный таким образом граф можно разложить в четырёхдольный ориентированный граф, так что каждое ребро имеет определенный и уникальный тип (рис. 2.6в)). В таком графе каждому циклу, состоящему из четырех ребер и не проходящему через одни и те же вершины, будет соответствовать возможный четырехугольник на изображении. Обнаружение циклов можно организовать, например, радиальными обходами с максимальной глубиной обхода, равной четырем, которые начинаются в одной из частей графа и могут неоднократно заходить в одну вершину. Сложность этого обхода составит $O(N^4)$, что накладывает дополнительные ограничения на количество начальных сег-

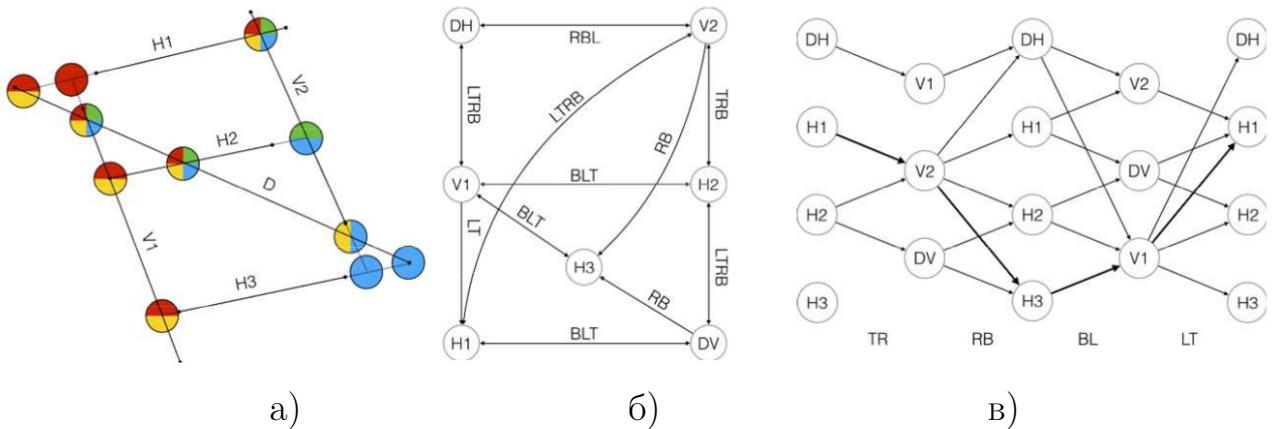


Рисунок 2.6 – Пример графа для 6 отрезков: а) точки пересечения сегментов и назначение типов; б) граф пересечений, каждое ребро соответствует одной точке пересечения; в) расширенный граф пересечений, каждое ребро соответствует возможной стороне четырехугольника.

ментов. Каждому четырехугольнику присваивается вес, равный разности сумм весов всех ребер и сумм весов всех вершин:

$$w_q = A \cdot \sum(w_v) - B \cdot \sum(w_e), \quad (2.10)$$

где A, B - параметры взвешивания. В конце алгоритма выбирается необходимое количество циклов в графе, наибольших по весу, и для каждого из них строится четырёхугольник-кандидат. Проверка геометрии и дальнейшее ранжирование выполняется по методу из раздела 2.2.1.

2.3 Методы выделения текстовых признаков

Помимо признаков, формируемых из произвольных текстов на объекте, существуют тексты, которые размещены именно в целях автоматической / упрощённой детекции/идентификации объекта. Для таких признаков задан стандарт (к примеру, число и тип символов, шрифт, контрольная сумма и алгоритм её вычисления), по которому можно отличить их от произвольных текстов и подлога. К известным примерам можно отнести номера автомобилей, вагонов, банковских карт. Документы с жёсткой структурой являются характерными представителями интересующих нас объектов. Отдельно среди них можно выделить документы, удостоверяющие личность, содержащие МЧЗ. МЧЗ внедрён

ICAO [132] для автоматизации ввода данных пассажиров, а его стандарт спроектирован таким образом, чтобы минимизировать помехи для распознавания текста в зоне. Машиночитаемые зоны, МЧЗ или MRZ в международном варианте, размещаются на документах, удостоверяющих личность. Существует ряд стандартов формирования МЧЗ, таких как ICAO и национальные стандарты для паспортов, разработка которых была направлена на массовое внедрение систем распознавания на контрольно-пропускных пунктах.

Согласно ICAO, фон (бланк документа в зоне) должен быть белым в ИК-спектре, используемый шрифт – OCR-B, размер символов и высота междустрочного интервала фиксированы. Допустимые паттерны МЧЗ:

- TD1 (ID карты) - 3 строки по 30 символов
 - TD2 (ID карты), MRVB (визы) - 2 строки по 36 символов
 - TD3 (паспорта), MRVA (визы) - 2 строки по 44 символа

Примеры приведены на рисунке 2.7.

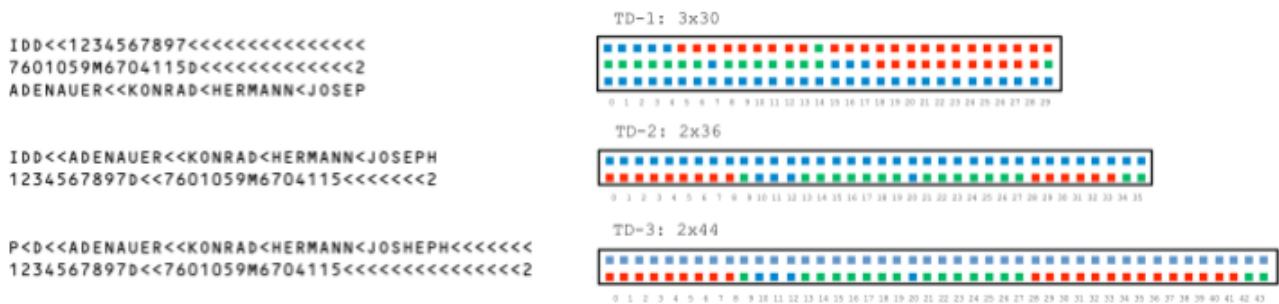


Рисунок 2.7 – Примеры МЧЗ.

Заметим, что по стандарту расположение зоны МЧЗ в структуре документа зафиксировано и известно. Решение задачи локализации всего документа задает и положение МЧЗ на изображении. Однако, локализация самой МЧЗ менее ресурсозатратна. Поэтому в случае ограниченной вычислительной мощности устройства, на котором выполняется распознавание, целесообразно рассматривать исключительно задачу локализации и распознавания МЧЗ, так как эта зона зачастую содержит все основные персональные данные, дублированные на документе.

Базовым регистратором изображений в таких системах был специализированный паспортный сканер. Они позволяют получать изображение за существенно меньшее время в сравнении с обычными сканерами, но качество изображения может быть существенно ниже. В зависимости от типа сканера, получаемые изображения содержат либо непосредственно машиночитаемую зону, либо

по типу обычного сканера документ целиком без элементов сцены. Методы для таких изображения позволяют достигать очень высокой точности за счёт учёта стандартов МЧЗ.

Так, в работе [133] предложено двухэтапное решение по локализации зоны. Первый этап – грубая оценка – основан на анализе гистограмм интенсивности и модуля градиента из части изображения. Оценка выполняется на части изображения – вертикальной полосе, проходящей через центр документа. Второй – уточнение – использует преобразование Хафа над выбранной грубой зоной, в т.ч. для оценки наклона. При этом низким считается разрешение порядка 150 дпі, и известно что документ повернут в диапазоне $-10 - +10$ градусов либо $+170 - +190$. В работе [134] предложен метод для документов с незначительно малым наклоном (1-2 градуса). Для поиска зоны используют проекцию на вертикальную ось в красном канале. Аналогично работе [133], для ускорения вычисления проекции используют только центральную полосу документа. Уменьшение выбросов на графике проекции достигается с помощью фильтра скользящей средней (как быстрая замена для фильтра Гаусса на всем изображении). В случае расширенного изображения, когда документ занимает не всю площадь изображения, подходы на базе вертикальный проекций дополняют аналогичной обрезкой по пикам на горизонтальной проекции [135].

Ограничение вычислительной сложности – не единственная проблема, возникающая в задаче распознавания МЧЗ с помощью смартфона. Стандарт требует, чтобы в инфракрасном диапазоне фон МЧЗ был однородным и светлым. Это облегчает распознавание МЧЗ специализированными устройствами, работающими в ИК-диапазоне. При этом камеры мобильного устройства снимают в видимом диапазоне, и наличие неоднородного фона на некоторых типах машиночитаемых документов служит помехой для автоматической обработки изображений. Существует также ряд более общих проблем [28], возникающих при попытке распознать документ, снятый камерой мобильного телефона. К ним относятся: блики, проективное искажение, изгиб документа и слабое освещение, часто приводящее к размытости снимков.

В статьях [28; 136] авторы подробно разобрали проблемы, возникающие при съёмке машиночитаемых документов с малоформатных цифровых камер, такие как:

- проявления «цифрового шума» и артефактов алгоритмов сжатия
- перепады яркости, блики и цветовые искажения

- поворот документа и проективное искажение
- изгиб строк документа

Первым предложенным подходом к решению задачи локализации МЧЗ была адаптация методов, основанных на проекциях и разработанных для сканированных изображений. Предполагалось, что модель искажений, появляющихся при съемке камерой, может быть редуцирована до поворота, который оценивается с помощью преобразования Хафа и Фурье [137]. Однако такой подход имеет значительные ограничения, так как в нем не учитываются перепады освещения и наличие сложной сцены на изображении.

Поиск МЧЗ с помощью выделения особых точек и сопоставления дескрипторов предложен в работе [138]. Хотя авторы демонстрируют высокую точность поиска, предложенные опорные зоны ограничивают применимость алгоритма одним типом и одной страной. Подходов для универсализации данного метода авторами не было предложено.

Работа [139] с выделением текстовых блоков морфологическим фильтром и вычленением из них строк МЧЗ выполнена в предположении всех реальных проблем. Однако покадровая точность метода весьма низкая даже на синтезированных данных, в которые не были включены такие искажения как сгибы документов и перепады освещения.

При разработке методов случаи были разграничены по предпочтаемым устройствам и условиям регистрации изображения:

- метод для видеокадров (в т.ч. полученных в условиях недостаточной освещённости)
- метод для смешанного потока данных (сканы, фото, видеокадры с преимущественно равномерным освещением)

Оба метода включают следующие общие этапы:

- выделение примитивов, соответствующих символам
- фильтрация примитивов
- построение графа с вершинами-примитивами
- построение минимального остовного дерева графа
- выделение кластеров на остовном дереве
- отбрасывание кандидатов по геометрическим критериям

2.3.1 Поиск МЧЗ на видеокадрах

Разработанный метод локализации машиночитаемой зоны на изображении основан на выделении и кластеризации особых точек с последующим анализом, преобразованием и отсечением кластеров по геометрическим признакам.

1. Выделение точек. Помимо базовых требований к методам выделения особых точек 1.3.4, для решения задачи были предъявлены следующие:

- быстродействие
- высокая плотность покрытия печатного текста
- игнорирование участков с прямолинейными границами

Предложено использовать метод YAPE [140], подходящий под все требования. Входное изображение предварительно масштабируем к фиксированному размеру (800 px по ширине), переводим в серое и сглаживаем с помощью фильтра Гаусса. Масштабирование, помимо дополнительного подавления шумов, связано с одним из параметров YAPE – максимальным радиусом области анализа для каждой точки. Результат работы детектора YAPE приведён на рис. 2.8.

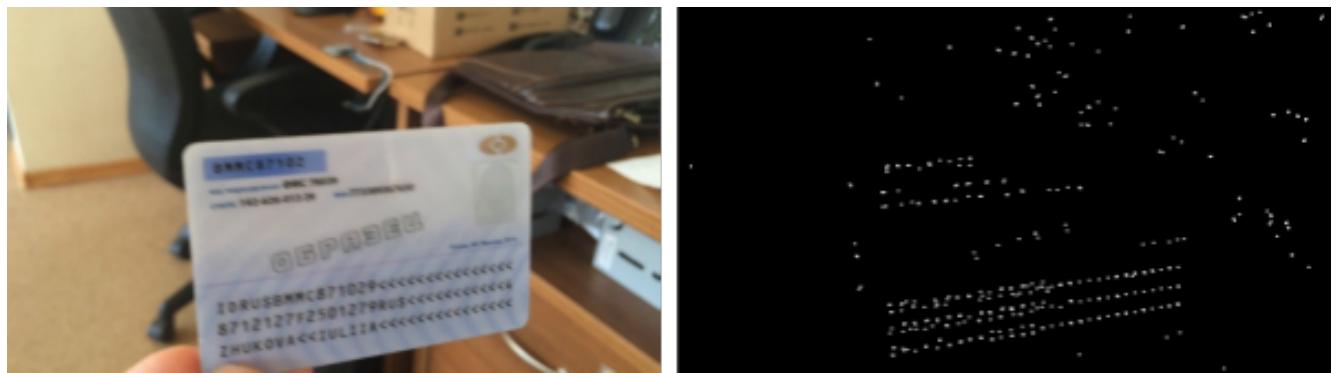


Рисунок 2.8 – а) исходное изображение б) точки, выделенные YAPE

2. Фильтрация точек. Точки, заведомо не принадлежащие искомой зоне, следует отбросить. Для этого используем следующую информацию:

- строки машиночитаемой зоны - наиболее протяжённый текстовый объект на документе
- выбранный детектор особых точек не выделяет точки, принадлежащие непрерывным границам (прямым на изображении)

Тогда применима следующая схема:

1. Вычислим на бинарной карте точек прямые-кандидаты, используя БПХ.
2. Для каждой точки определяем ближайшую прямую среди кандидатов.
3. Точка отбрасывается, если расстояние до ближайшей прямой больше порога.

Поскольку ожидаемые прямые проходят по строке, порог было предложено привязать к высоте символа h_{sym} на изображении (максимальной на изображении зафиксированного размера). В данной работе используется $0.5h_{sym}$.

3. Кластеризация точек. Точки объединяются в кластеры по схеме:

1. Построим полный граф, в котором точки будут вершинами, а вес ребра между точками p_i и p_j будет рассчитываться по следующей формуле:

$$w_{ij} = g(a)|p_i - p_j|, \quad (2.11)$$

где $g(x)$ - некоторая монотонная функция, a - угол между ближайшими прямыми для p_i и p_j . Функция $g(a)$ штрафует ребро за непараллельность ближайших прямых у его вершин, т.к. вероятно, что одна из точек не соответствует МЧЗ.

2. Определим на графике минимальное остовное дерево
3. Разобьём дерево на несколько частей (кластеров), выкинув ребра, вес которых больше порога (например $t = 2 w_{sym}$, где w_{sym} - максимально возможная ширина символа МЧЗ на изображении зафиксированного размера).

4. Анализ кластеров, выбор результирующего прямоугольника.

Если размер кластера (количество точек в нём) меньше, чем минимально допустимое количество символов МЧЗ в строке, то отбрасываем его. Для каждого из полученных кластеров находится угол наклона к сторонам изображения как среднее из углов прямых, соответствующих точкам, входящим в кластер. После этого находится описанный вокруг точек кластера прямоугольник, расположенный под этим углом. Из полученных прямоугольников выбирается тот, который лучше всего соответствует известным геометрическим ограничениям какого-либо из типов МЧЗ.

В случае, когда несколько кластеров структурно идентичны, анализируем растр внутри каждого кластера. Символ «<» является специфичным для МЧЗ разделителем, т. е. в подавляющем большинстве случаев он встречается как минимум 2 раза и не представлен в других полях ID документов. Сепаратор

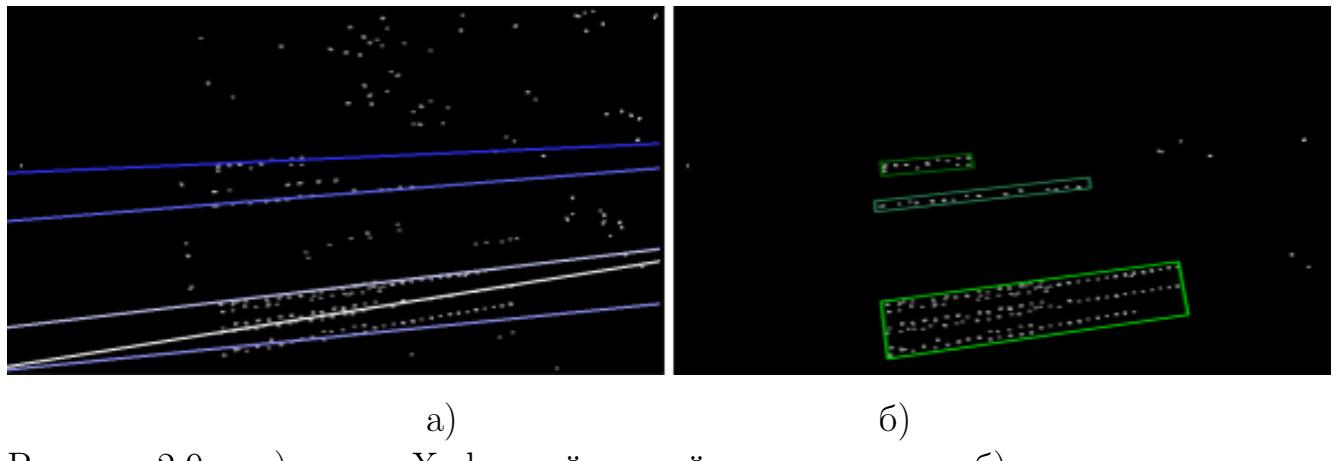


Рисунок 2.9 — а) линии Хафа, найденный на карте точек б) кластеры-кандидаты для выбора результата

стабильно распознается как характерная точка, поэтому достаточно проанализировать локальные окрестности точек в начале и конце кластера. Для этого мы представляем эталонное изображение символа «<» как локальный дескриптор $f_<$. Метод дескрипции в данном случае не требует крайней скорости и устойчивости, поскольку масштаб и поворот определены прямоугольником, а число сопоставлений невелико. Остальные 36 символов алфавита МЧЗ не могут использоваться в качестве локальных дескрипторов – шрифт OCR-B может встречаться не только в МЧЗ, но и в заполняемых полях документов, что вместе с увеличением числа вычислений может привести к коллизиям выбора кластера.

Патчи-окрестности изображения вокруг нужных точек рассчитываются с учётом размера и поворота кластера. Т.к. известны для сепаратора известны места обязательного размещения, то не требуется анализировать окрестности всех точек кластера - достаточно рассчитать дескрипторы $\{f_i\}$ для ограниченного интервала ($1/4$ конец и начало, в случае перевёртыша). Для каждого такого дескриптора вычисляется расстояние с эталонным дескриптором $f_<$. Обнаружение символа успешно, если расстояние меньше порога. Вес кластера рассчитывается следующим образом:

$$w = \left(1 - \frac{N_m}{N_c}\right) \cdot \sum_{i=0}^{N_c} D(f_i, f_<), \quad (2.12)$$

где N_m - число расстояний меньше порога, N_c - число сравнений, $D(a, b)$ - метрика расстояния между парой дескрипторов a и b .

2.3.2 Поиск МЧЗ на произвольных изображениях

1. Поиск компонент символов. Символ рассматривается как связная компонента, тёмная на светлом фоне, за положение символы примем центр массы компоненты. Для их поиска входное изображение преобразуется в бинарное изображение $I: \forall(x,y) : I(x,y) \in [0,1]$, где 0 соответствует чёрному. На практике, требования к использованному при этом алгоритму бинаризации оказываются минимальными: на бинаризованном изображении 2.10 одному печатному символу должна соответствовать одна компонента связности, а ее площадь должна быть сопоставимой с площадью символа. Последнее нужно для фильтрации компонент связности и увеличения устойчивости к шуму типа «соль и перец». В реализации метода поиска МЧЗ используется метод Niblack, поскольку он незначительно превосходил аналоги (NICK и использующийся в [139] toggle mapping) по качеству и скорости. Построим начальный граф связности пикселей $G_0 = \langle V_0, E_0 \rangle$, где $V_0 = \{(x,y) : I(x,y) = 0\}$, $E_0 = \langle (x_0, y_0), (x_1, y_1) \rangle : |x_0 - x_1| + |y_0 - y_1| = 1$.

Обозначим C_0 множество компонент связности графа G_0 . Координаты центра масс для компоненты связности $c \in C_0$:

$$p(c) = \langle p_x, p_y \rangle : p_x = \frac{1}{|c|} \sum_{(x,y) \in c} x, \quad p_y = \frac{1}{|c|} \sum_{(x,y) \in c} y \quad (2.13)$$

На рис. 2.10 показаны центры компонент для бинаризованного изображения.

2. Построение основного графа. Определим множество вершин основного графа $G_1 = \langle V_1, E_1 \rangle$ через C_0 :

$$V_1 = \{c \in C_0 : |c| > tr_V\}, \quad (2.14)$$

т.е. компоненты связности, размер которых превосходит пороговое значение tr_V . Тогда множество рёбер основного графа можно определить с ограничением длины:

$$E_1 = \langle (c_i, c_j) : \sqrt{(p(c_i) - p(c_j))^2} < tr_E, c_i, c_j \in V_1 \rangle \quad (2.15)$$

При построении графа для каждого ребра вычисляется соответствующий ему угол наклона. Рёбра не направленные, поэтому этот угол может принимать значения от 0 до 180° . Этот граф состоит из небольшого числа вершин и ребер, что

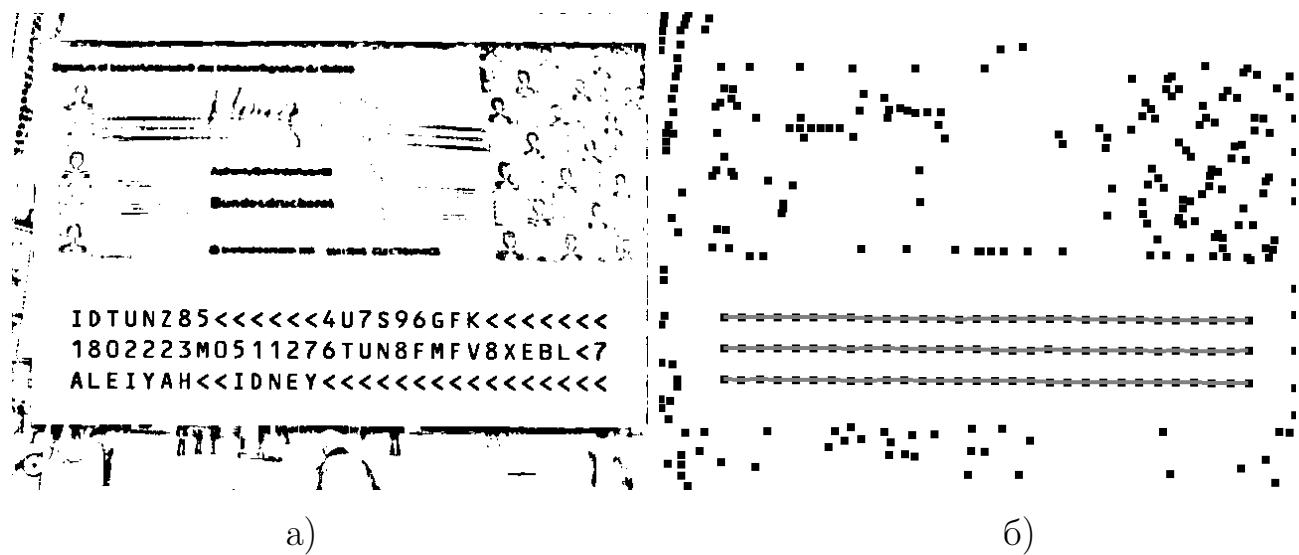


Рисунок 2.10 — а) бинаризованное входное изображение б) отображение центров его компонент

позволит в дальнейшем анализировать его, используя минимальное количество вычислительных ресурсов. Пример графа на рис. 2.11а.

3. Определение ориентации текста. Для определения ориентации текста строится минимальный оствовный лес. Поскольку расстояние между соседними символами одного слова меньше, чем расстояние между строками, большинство рёбер в оствовном лесу – рёбра, соединяющие соседние буквы одного слова. На этом наблюдении основан алгоритм поиска направления строк текста на изображении. Строится круговая гистограмма, показывающая распределение рёбер оствового леса по углам наклона. Мода полученного распределения соответствует углу наклона линий текста. Для устойчивости вместо поиска интервала гистограммы с наибольшим числом элементов выполняется поиск пары соседних интервалов с наибольшим суммарным числом элементов. Мода распределения вычисляется как средневзвешенное значение углов, соответствующих этим соседним интервалам.

Для большинства документов распределение получается унимодальным. Однако в некоторых случаях на кадре изображен разворот документа, причем на одной странице ориентация текста альбомная, а на другой – портретная. В этом случае распределение бимодально, и найденная по гистограмме мода может соответствовать как направлению строк МЧЗ, так и перпендикулярному направлению. Таким образом, в общем случае предложенный метод находит направление строк МЧЗ лишь с точностью до поворота на угол 90° . Для обхода этой проблемы используется следующее простое правило: если к концу обра-

ботки не найдено текстовых структур, напоминающих по свойствам известные типы МЧЗ, предполагаемое направление текста меняется на перпендикулярное и все следующие шаги алгоритма, начиная с фильтрации рёбер графа по направлению, повторяются.

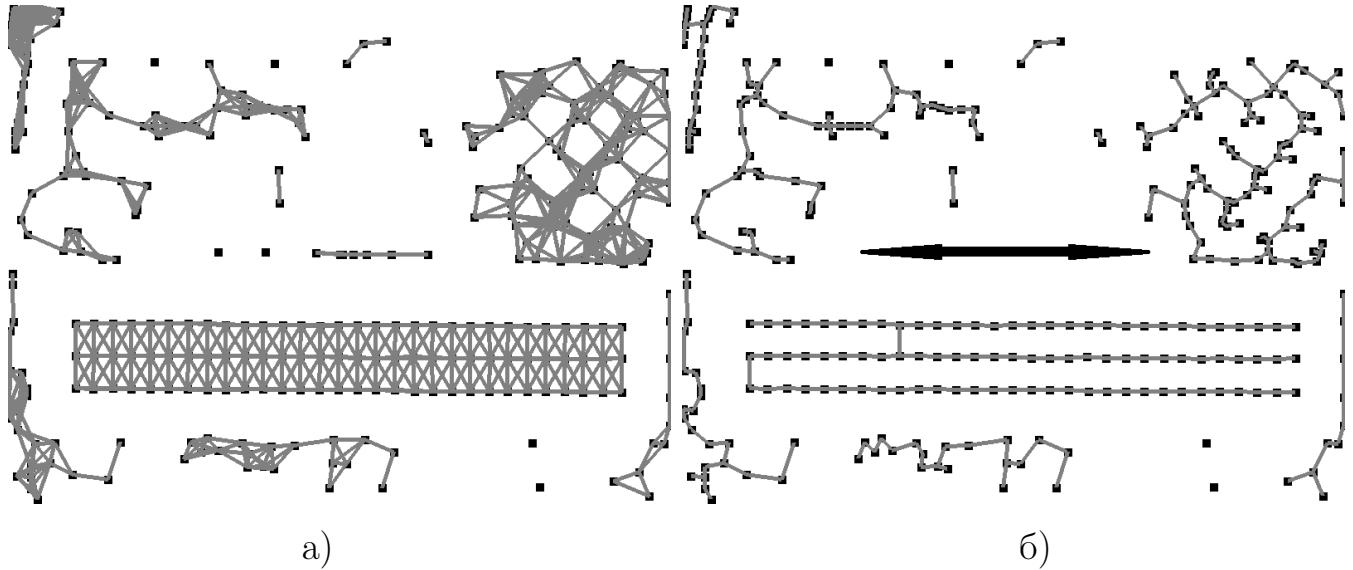


Рисунок 2.11 — а) граф связности компонент G_1 , б) его оствное дерево и направление текста

4. Поиск путей в графе, соответствующих строкам МЧЗ. Из множества путей в графе нужно выделить те, которые соответствуют строкам МЧЗ. Сначала из исходного графа удаляются все рёбра, отклонение которых от найденного направления текста превосходит пороговую величину. Минимальный оствный лес строится заново. В полученном подграфе остаётся найти пути, которые соответствуют строкам МЧЗ. Опишем эвристику, использующуюся для их поиска. Искомый путь должен удовлетворять следующим условиям:

1. Степень всех его вершин, кроме, возможно, крайних, должна быть равна 2.
2. Количество рёбер в пути должно превосходить m . Для устойчивости к ошибкам бинаризации m должно быть меньше, чем предполагаемое число символов в строке МЧЗ. При этом оно должно оставаться достаточно большим, чтобы минимизировать количество строк обычного текста, попадающего под данные критерии. На практике максимальное качество достигается при $m=10$.
3. Длины любых двух соседних рёбер должны отличаться не более чем в k раз.

4. Углы наклона двух соседних ребер должны отличаться не более чем на π градусов.
5. Путь максимален по включению.

Для удобства будем называть пути, удовлетворяющие перечисленным условиям, цепями. На рис. 2.11б изображён минимальный оставный лес, в котором оставлены только рёбра, входящие в цепи. В силу малого размера графа, поиск цепей может быть выполнен при помощи тривиальной модификации алгоритма обхода графа в глубину.

5. Кластеризация цепей. Это завершающий этап обработки, здесь знания о структуре МЧЗ используются для поиска наиболее консистентного подмножества цепей. На вход подается совокупность цепей, выходом служит окаймляющий прямоугольник. После завершения предыдущего этапа одной строке МЧЗ может соответствовать несколько цепей. Это может произойти, например, если символ строки МЧЗ некорректно бинаризовался из-за наличия блика и из графа исчезла соответствующая вершина. Для корректной обработки таких случаев вводится шаг объединения коллинеарных цепей. Цепи объединяются, если они, во-первых, коллинеарны с учётом некоторой погрешности, во-вторых, средние длины их рёбер схожи, в-третьих, расстояние между цепями не превосходит пороговую величину.

6. Анализ кластеров. Последний шаг – поиск среди цепей кластеров, соответствующих геометрическим свойствам МЧЗ. Цепи, объединённые в один кластер, должны быть приближённо равны по углу наклона, длине и средней длине входящих в них рёбер, а расстояние между двумя соседними цепями одного кластера не должно превосходить пороговую величину. Окаймляющий прямоугольник для большего по площади кластера является результатом работы алгоритма.

2.4 Эксперимент

2.4.1 Наборы данных

- **SmartDoc 2015**[141] - содержит изображения документов формата А4. Состоит из 150 10-секундных видеороликов в разрешении 1920x1080, с документами (в раскадровке образуют порядка 25000 изображений). Для создания набора использовался Google Nexus 7. Ролики содержат 5 вариаций сцены, в пакете документов 6 типов, по 5 документов каждого типа. В кадре только 1 документ, границы которого не выходят за границу изображения. На многих кадрах присутствуют перепады освещения, дефокус, смаз, блики и тени. Данные явления характерны реальному сценарию съемки. Присутствуют как фоны сливающиеся с документом, так и визуально перегруженные. Для всех изображений есть идеальная разметка, содержащая настоящие координаты углов на изображении.

- **MIDV-500**[142] - содержит изображения документов, удостоверяющих личность (ID). Состоит из 500 видеороликов в разрешении 1920x1080px (в раскадровке образуют 15000 изображений). В MIDV-500 представлено 50 различных типов документов, по одному экземпляру на тип. Видеоролики отсняты с использованием двух мобильных устройств (Apple iPhone 5, Samsung Galaxy S3) в 5 разных условиях, по 30 кадров в каждом. В кадре не более одного документа, документ может выходить за границы и быть неразличимым (сильный смаз). Также набор содержит 50 изображений-эталонов документов. Для всех изображений есть идеальная разметка, содержащая тип документа и настоящие координаты углов на изображении либо отметки, соответствующие неразличимым изображениям.

- **MRZ** [139] - содержит полностью синтезированные изображения документов с МЧЗ. Всего 456 000 изображений, для 422338 есть разметка, содержащая данные МЧЗ – текст и координаты окаймляющего прямоугольника.

Таблица 1 – Сравнение методов поиска четырёхугольников на SmartDoc 2015

Метод	IoU	Время на кадр	Устройство
[АЧ1]	0.9100	~80 мс	iPhone 6
[АЧ2]	0.9548	~120-150 мс	iPhone 6
LRDE	0.9716	~ > 1 мин	iPhone 4
ISPL-CVML	0.9658	~130-210 мс	AMD 1055T
NetEase	0.8820	-	-
A2iA	0.8090	-	-
RPPDI-UPE	0.7408	-	-
SEECS-NUST	0.7393	-	-

2.4.2 Поиск четырёхугольника

При оценке скорости и точности предложенных методов был использован фиксированный набор параметров. Общие параметры:

Пороги для фильтрации четырёхугольников, несоответствующих прообразу: $tl = 0.97$, $tr = 1.04$ и $ta = 2.5^\circ$. Для внешней оценки фокусного расстояния использовано приближение $0.705\sqrt{W^2 + H^2}$. Параметры метода АЧ1: Масштабирование: не более 240 пикселей по меньшей стороне. $N_h = 15$ «горизонтальных» прямых, $N_v = 45$ «вертикальных». Параметры метода АЧ2: Масштабирование: не более 300 пикселей по меньшей стороне.

Для замеров скорости предложенных методов использовался iPhone 6, за оценку времени работы принято среднее время по 100 случайным изображениям.

В таблице 1 приведено сравнение с другими методами, заявленными на соревнование методов в рамках конференции ICDAR [143] на одноимённом наборе. В данном эксперименте этап восстановления одной стороны не использован. Качество локализации оценивается по IoU (1.2). К сожалению, не все методы из табл. 1 опубликованы независимо от конкурса, а в для опубликованных не отражены данные о скорости. Тем не менее, можно проанализировать имеющуюся информацию о методах, превзошедших авторский. Для ISPL-CVML заявлено, что он основан на LSD, а при выборе кандидата авторы минимизируют оценку, в которую входят характеристики цвета и интенсивности. Можно ожидать, что его вычислительная сложность того же порядка, что и предложенный метод,

Таблица 2 — Влияние восстановления четвёртой стороны при поиске четырёхугольников на MIDV-500

Метод	4 угла (9791)	3 угла (2200)	весь (15000)	время, мс
[АЧ1] без восстановления	0.9124	0.6432	0.7055	54.2
[АЧ1] с восстановлением	0.9466	0.8377	0.8050	67.8

что косвенно подтверждается более общей публикацией авторов [144], где метод с такими же ключевыми особенностями оценивается в 130-210 мс в зависимости от типа плоского объекта на AMD Phenom II X6 1055T (вычислительная эффективность в 2 раза выше Apple A8 в iPhone 6). Метод LRDE описан, как использующий дерево форм (близости), для которого строят образы изображений в другом цветовом пространстве. Его скорость показана в более поздней работе с участием соавторов [145] и превышает 1 минуту для iPhone 4. Таким образом, предложенный метод обладает лучшей скоростью при достаточно высокой точности (3 место в общем зачёте).

Дополнительно в табл. 2 приведено влияние этапа восстановления стороны на точность и скорость. Для исследования выбран MIDV-500, поскольку там встречаются кадры, в которых частично или полностью скрыты границы объекта. Отдельно оценены подвыборки, на которых документ виден целиком (9791 изображений) либо видно 3 из 4 углов (2200 изображений). Таблица показывает, что при замедлении 25% достигается снижение доли ошибок в 1.5-2.1 раз в зависимости от того, какая доля объекта за кадром или заслонена, а так же от сложности сцены. Т.о. внедрение данного этапа сохраняет возможность вычислений на мобильных устройствах в реальном времени.

2.4.3 Поиск МЧЗ

Результатом работы предложенных методов является окаймляющий четырёхугольник. В работе [139] понятие окаймляющего четырёхугольника отсутствует, а под качеством детектирования подразумевается «качество сегментации»: детектирование считается успешным, если количество строк и символов найденной структуры, а также её геометрические характеристики согласуются с априорной информацией о типе МЧЗ. Для достоверного сравнения внутри най-

денного прямоугольника реализована морфологическая сегментация, используемая в [139]. Для замеров скорости использован Iphone 5s (для всех методов табл.3). Как и в разделе 2.4.2, отражено среднее по набору время. Показано,

Таблица 3 – Сравнение точности и скорость методов поиска МЧЗ на синтезированном наборе **MRZ**

Метод поиска	Качество сегментации	Время детекции, мс
Hartl [139] (1 кадр)	0.561	~ 5
Hartl [139] (5 кадров)	0.881	~ 25
[AT1]	0.942	~ 40
[AT2]	0.978	~ 20 – 40

что предложенные методы существенно точней обнаруживают МЧЗ по 1 изображению, чем метод [139] при интеграции 5 кадров. Метод вычислительно более затратен (требуется больше времени на анализ 1 кадра), но вне зависимости от этого соответствует возможностям мобильных устройств.

2.5 Выводы по главе

В данной главе исследуются наблюдаемые примитивы «линия» и «прямоугольник», а также характерные текстовые признаки в контексте принадлежности плоскому ригидному объекту.

Для формирования изображения в модели камеры-обскуры показано, что в случае прообраза-прямоугольника с известными геометрическими характеристиками для наблюдаемого четырёхугольника сторона, отсутствующая по наблюдению, может быть восстановлена аналитически по трём наблюдаемым (найденным).

Приведено уточнение теоремы о существовании у четырёхгранного угла сечения – параллелограмма для центрально-проективной модели формирования изображений: объект-параллелограмм порождает только образы-четырёхугольники, углы которых лежат а) в плоскости на расстоянии равном фокусному от центра координат б) на катетах четырёхгранного угла, образованного параллелограммом и центром координат. Это позволяет оценить наблюдаемый четырёхугольник с точки зрения соответствия реальному прообразу.

Предложен метод локализации прямоугольного объекта в реальном времени при заданных зонах для его сторон в кадре. Для большей устойчивости метода к шумному фону предложена фильтрация граничных компонент. Скорость достигается за счёт зонирования.

Предложен быстрый метод локализации прямоугольного объекта, произвольно размещённого на кадре или скане. Для повышения точности используются границы разных детекторов. Скорость достигается за счёт ограничений на графе генерации кандидатов при построении и обходе.

Оба метода используют оценку прообраза и возможность восстановления стороны, описанные выше, для повышения устойчивости к проективным искажениям и заслонениям. Оба метода допускают возможность отказа при отсутствии четырёхугольника, соответствующего допустимому прямоугольнику.

МЧЗ рассматривается как текстовый признак с известным ограниченным набором паттернов. Предложен быстрый метод поиска МЧЗ для кадров мобильных устройств при ограниченном повороте. Поиск основан на выделении строк на карте таких особых точек, которые отображают символы как их скопление. Предложено выделять с помощью преобразования Хафа кластеры точек, образующие прямые-строки, и фильтровать точки сцены.

Предложен быстрый метод поиска МЧЗ для кадров и сканов с произвольным поворотом. Символы рассматриваются как точки - центры компонент бинаризованного изображения. Точки сцены фильтруются по размеру, взаимной удалённости и количеству.

Оба метода допускают отказ при отсутствии кластера, достаточно соответствующего одному из паттернов. Показано, что известные паттерны текста в зоне позволяют выделить зону среди прочих без анализа и распознавания всех символов.

Глава 3. Метод идентификации и локализации объекта на изображении

В данной главе определены параметры задачи классификации и локализации плоских ригидных объектов в системах распознавания и представлен метод, удовлетворяющий реальным требованиям. Метод основан на сопоставлении входного изображения с изображением - эталоном.

3.1 Параметры задачи

Определим параметры задачи для двух основных вариантов применения систем распознавания. Как упомянуто во введении, в первом случае изображение (скан или фотография) объекта передаётся для анализа на сервер. Во втором случае распознавание выполняется непосредственно на мобильном устройстве, а входными изображениями главным образом выступают кадры с камеры этого устройства.

Различия во входных данных и требованиях подробно описаны в таблице 4.

Таблица 4 – Сравнение параметров мобильного и серверного подходов к распознаванию

Устройство	Мобильное (смартфон / планшет)	Серверное
Входное изображение	Видеокадр	HD-фото с мобильного устройства; скан
Положение объекта	Все границы объекта внутри кадра	Две границы могут отсутствовать (если объект был в углу сканера)
Характерное число пикселей	от 0.5МП до 8МП (Ultra HD 4K)	Скан: от 150 до 600 PPI, фото от 2МП тдо 15МП
Максимальное время работы	0.5 секунд (типовые устройства: Exynos 7420, Apple A8)	1 секунда (типовые устройства: AMD FX-8350, Intel Core i7 4770S)

Требования к скорости исходят из следующих соображений:

- на сервере время распознавания изображения не должно превышать время сканирования. Скорость работы современных потоковых сканеров составляет от 20-30 страниц в минуту (Canon imageFORMULA DR-C, Kodak i1) до 200+ у высокопроизводительных моделей (Canon imageFORMULA DR-G, Kodak Alaris i5 series), в среднем - 60 стр/мин.
- мобильные системы распознавания оперируют понятием «реальное время» начиная с 1 кадра в секунду при анализе для полного кадра [146] и 6-7 кадров в секунду при анализе отдельных зон интереса [147]; распознавание паспорта на мобильных сканерах должно укладываться в 1 секунду [134].

3.2 Модель изображения

Для N типов объектов определим классы изображений $C = \{C_i\}_{i \in [0,N]}$ следующего вида:

C_i – класс изображений i -го типа объекта, при $i \in [1,N]$

C_0 – класс прочих изображений

Для каждого класса C_i задано изображение-эталон T_i , $i \in [1,N]$.

Представление изображения I в модели «созвездия признаков» можно записать как:

$$\omega = W(I) = \{\langle p_i, f_i \rangle\}_{i \in [1,M]}, \quad (3.1)$$

где $p_i = \langle x_i, y_i \rangle$ – координаты i -й точки, f_i – дескриптор окрестности i -й точки, M – число точек в представлении.

Требуется определить класс C_i входного изображения Q . В случае, если $i \in [1,N]$ необходимо вычислить преобразование $H : Q \rightarrow T_i$ переводящее точки объекта на запросе в точки соответствующего эталона. В нашей работе будем рассматривать только проективные преобразования, порождаемые камерой, параметры которых заданы матрицей 3×3 .

3.3 Создание набора моделей

Далее для простоты представление изображения $W(T_i)$ эталона будет обозначено как *шаблон*.

3.3.1 Создание шаблона

Эталоном, с учётом 3.2, будет выступать изображение, полученное в контролируемых условиях. Считаются предпочтительными следующие условия:

- изображение высокого разрешения, полученное при равномерном нормальном освещении
- плоскость сенсора (камеры или сканера) параллельна плоскости объекта
- изображение обрезано по границам объекта

В ряде случаев объекты либо изображения объектов при ригидной структуре имеют заполнение – признаки одного типа, значения которых меняются от экземпляра к экземпляру. Так, в задаче классификации ID документов, бланк паспорта задаёт структуру, а персональные данные являются заполнением. В силу неустойчивости такие признаки скорее являются отвлекающим фактором для методов, оперирующих стабильной структурой, а значит избыточны для шаблона. Поэтому каждому эталону будет соответствовать разметка – набор координат прямоугольных зон с маркировкой признаков внутри зоны. Маркировка указывает, являются ли признаки внутри зоны опциональными (на заполнении), обычными или необходимыми. Отображение маркировки представлено на рисунке 3.1. Опциональные признаки отбрасываются при построении компактного представления $w_i = W(T_i)$ для эталона T_i .

Ряд близких решений для фильтрации заполнений был предложен ранее. Так, в работе [148] авторы вводят маскировку - заполнения перекрываются монотонными прямоугольными областями. Для ряда алгоритмов выделения признаков это соответствует «слепой зоне». В работе [26] авторы предлагают использовать несколько эталонов, сохраняя только признаки, присутствующие на каждом из них.



Рисунок 3.1 — Изображение-эталон с маркированными зонами. Красным отмечены опциональные зоны, зелёным – необходимые.

Изображения для шаблонов предварительно нормализуем следующим образом: сначала масштабируем – длина изображения приводится к стандартному значению при сохранении соотношения сторон, а затем сглаживаем фильтром, сохраняющим границы (билиатеральный). Такая предобработка упрощает вычисление масштаба для входного изображения, поскольку не допускает пары шаблонов разных размеров при одинаковых пропорциях.

На нормализованном эталоне нужно выделить особые точки и их дескрипторы. В реализации метода для детекции точек, как и в разделе 2.3.1, выбран алгоритм YAPE. Каждой точке соответствуют координаты и оценка, привязанная к перепаду интенсивности в окрестности. После фильтрации по зонам оставшиеся точки сортируются по оценке, чтобы оставить не более M лучших. Выбранный алгоритм не оценивает направление в точке, а по сегментам и прямым на входном изображении поворот объекта можно определить с точностью до поворота на 90° , что следует учитывать при выборе дескрипторов и дальнейшем построении описания. Так, быстрые бинарные дескрипторы, рассмотренные в части 1.3, устойчивы в пределах 15° - 30° . В случае выбора таких дескрипторов требуется рассмотреть повороты на 0 , 90 , 180 и 270 градусов для обеспечения классификации изображений с повёрнутыми объектами. Вычислительно менее затратно учитывать это на этапе работы с эталонами, которая проходит офлайн. В итоге, каждому эталону будет соответствовать 4 описания.

3.3.2 Создание поискового индекса

Тривиальное сопоставление дескрипторов «каждого с каждым» ставит скорость метода в линейную зависимость от числа шаблонов. Этого можно избежать, воспользовавшись следующими соображениями:

- как указано в 1.4, метрические дескрипторы позволяют заместить полное сравнение использованием алгоритмов аппроксимированного поиска без значимой потери в точности;
- эталоны известны заранее, а значит время их предварительной обработки и построения индексов не имеют существенного значения – только результирующий объём памяти индексов и время поиска в них.

Для реализации в методе предложено использовать иерархическую кластеризацию, подобную FLANN [94]. Такая структура может быть реализована для произвольной метрики. Для построения индекса обработаем каждое из полученных описаний $W(T_i)$ следующим образом: для каждой точки $p_j \in [1, |W(T_i)|]$ в описании занесём запись вида $\langle i, j, f_j^i \rangle$. В таком индексе для 1 дескриптора поиск ближайших выполняется за $O(\log \sum_{i=0}^N |T_i|)$. При этом для целочисленных метрик, как метрика Хэмминга, формат поиска в индексе заменяется: вместо « k -ближайших» используется поиск «в радиусе», т.к. в первом случае, при обнаружении нескольких равноблизких признаков, нужный признак может быть обрублен по принципу количества, а не расстояния.

3.4 Анализ входного изображения

3.4.1 Выделение глобальных признаков

На 1 этапе выполняется поиск глобальных признаков, поскольку эта информация позволяет улучшить скорость поиска локальных признаков и получить более точные сопоставления с шаблоном правильного типа. Выбор методов выделения прямых и четырёхугольников зависит от типа рассматриваемой системы. Методы, разработанные в 2.2 и 2.3, спроектированы так, чтобы ис-

пользоваться в описанных случаях. Так, метод поиска четырёхугольников на видеокадрах удовлетворяет требованиям для мобильной системы. Результатом их применения будет набор Q четырёхугольников-кандидатов и набор L линий, их образующих.

Параметры обработки изображения при выделении точек и расчёте дескрипторов привязаны к оценкам прямых и четырёхугольников, найденных на предыдущем этапе. Выделим следующие случаи:

- а) четырёхугольник найден и помечен валидным (accepted) – вес четырёхугольника выше порога,
- б) четырёхугольник найден и помечен невалидным (rejected) – вес четырёхугольника ниже порога,
- в) четырёхугольник не найден – недостаточно прямых, либо ни один из кандидатов не мог быть получен из прямоугольника в результате проективного преобразования, порождаемого камерой.

В первом случае лучший четырёхугольник с высокой точностью определяет масштаб и поворот объекта на входном изображении. Во втором случае доверять лучшему четырёхугольнику нельзя, поэтому будем определять только поворот изображения. Для этого выделим все образующие прямые четырёхугольников-кандидатов, удовлетворяющих геометрическим ограничениям. Полученные прямые отсортируем по углу и выберем ту прямую, для которой наибольшее число других прямых вмещается в угловое окно размером α . Будем считать α – углом поворота объекта. В случае, когда полученные геометрические признаки не позволяют уверенно определить один или оба параметров, вместо них используют стандартные значения.

3.4.2 Выделение локальных признаков

На входном изображении, как ранее на эталонах, выделяются особые точки. Размеры и ориентация областей входного изображения при расчёте дескрипторов корректируются исходя из значений, полученных выше. Для алгоритмов на базе особых точек могут представлять проблему случаи, когда фон содержит существенно больше откликов детектора, чем объект. Расчёт дескрипторов для фоновых точек и их сопоставление будут занимать значимое время. Предложе-

но отфильтровать полученные точки перед этапом расчёта дескрипторов, т.е. отделить фоновые точки от точек объекта. Если выделенные ранее четырёхугольники признаны валидными, то осмысленно сохранить только их внутренние точки. Если же четырёхугольник невалиден или ненайден, то требуется дополнительный анализ.

Для этого была обучена и использована облегчённая модель сети глубокого обучения для классификации точек на объект/фон. Используется свёрточная нейронная сеть с одноканальным входом размером 100×100 . Архитектура, размер входных данных, глубина сети и количество фильтров выбраны таким образом, чтобы, во-первых, в поле зрения CNN попадало достаточно информации для принятия точных решений, во-вторых, количество вычислительных операций, а, следовательно, и время обработки, были приемлемы. Создание конкретного экземпляра классифицирующей сети не относится к результатам диссертации, её параметры детально описаны в [15].

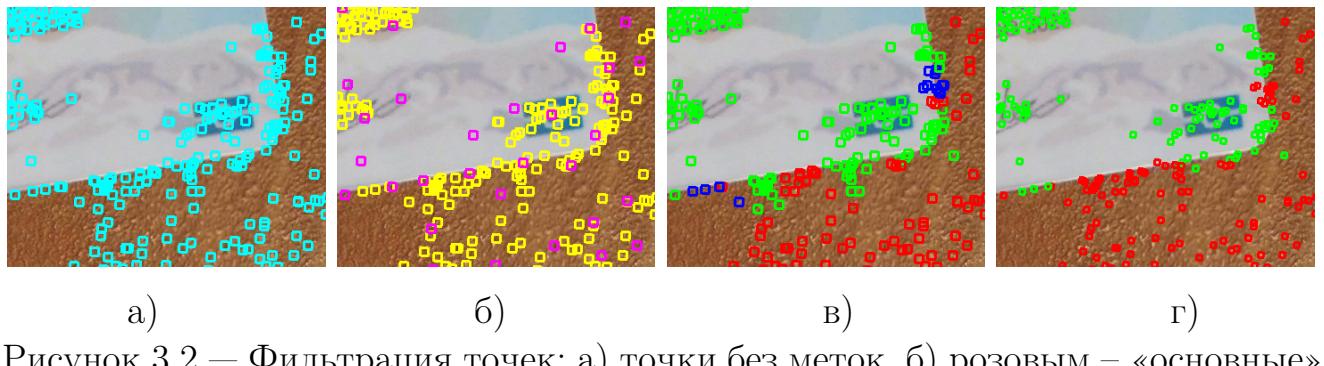


Рисунок 3.2 — Фильтрация точек: а) точки без меток, б) розовым – «основные» точки, жёлтым – «побочные», в) красным - «фон», синим - «конфликт», зелёным - «объект», г) решение конфликта

Полутоновый участок с центром в ключевой точке подаётся в ИНС в качестве входных данных. Размер патча принимается равным размеру ключевой точки, а затем масштабируется до входного размера ИНС. Для ускорения фильтрации предлагается применять сеть не ко всем ключевым точкам изображения, а к определённому их подмножеству. Рассмотрим вектор p , содержащий все ключевые точки изображения, упорядоченные в порядке невозрастания оценки ключевой точки. Проходя по этому вектору от начала до конца, мы отнесём текущую ключевую точку к классу «основная» или «побочная». Если рассматриваемая точка ещё не помечена, ей присваивается класс «основная», а все точки в некоторой δ -окрестности помечаются как «побочные». После выполнения этой операции над всеми элементами вектора ключевых точек «основные» точки из

разбиения рассматриваются как вход для ИНС (рис. 3.2б) Чем больше δ , тем меньше точек подаётся на вход сети, и тем быстрее будет алгоритм фильтрации. С другой стороны, выбор большого значения δ приводит к менее точной фильтрации, поскольку, во-первых, могут возникнуть ошибки на границе объект-фон и, во-вторых, возможные ошибки сети будут более значимы. Сначала «основные» точки классифицируются с помощью ИНС, а затем на основе полученного результата, присваиваются метки «побочным» точкам. Предлагается следующий подход для разметки «побочных» точек. Выполняется проход по вектору r , и в процессе прохода точки в δ -окрестности «основных» точек помечаются их же метками: если «основная» точка - фон, то все в её окрестности тоже фон. Если какая-либо из точек находится в пределах пересечения окрестностей с разными метками, то такой точке присваивается метка «конфликт» (рис. 3.2в). Если текущая рассматриваемая точка - «конфликтная», то она отправляется на вход CNN для уточнения метки, после чего полученная метка пытается распространиться по всей её δ -окрестности (рис. 3.2г).

Все точки, оставшиеся «конфликтными» после завершения прохода по вектору, помечаются как точки объекта, поскольку гораздо важнее сохранить ключевые точки, которые могут принадлежать объекту, чем удалять точки, которые, вероятно, принадлежат фону.

Рассчитаем зону покрытия объекта локальными признаками как окаймляющий прямоугольник оставшихся точек. Если линейные размеры прямоугольника значимо отличаются от диапазона для шаблонов, то масштаб следует скорректировать. Вычислив дескриптор $f(p_i)$ для каждой из оставшихся точек p_i , получаем представление для входного изображения вида $W(Q) = \{p, f\}$

3.5 Сопоставление изображений в модели

3.5.1 Отсечение и ранжирование классов по локальным признакам

Если объект на входном изображении визуально соотносится с каким-либо из эталонов, ожидается, что для некоторых особых точек с входного изображения будут существовать близкие (похожие) особые точки из эталонов. Пару де-

скрипторов будем считать близкой, если расстояние между ними не превышает заданный порог r_{max} :

$$|f_i - f_k| < r_{max} \quad (3.2)$$

Таким образом, для каждой особой точки $\langle p, f \rangle \in W(Q)$ нужно найти похожие среди особых точек эталонов. Для этого воспользуемся построенным индексом. Заметим следующее: наши предположения о возможной вариативности признаков ограничены, т.к. изображение, соответствующее конкретному классу, может быть в единственном экземпляре. К найденным дескрипторам применяется процедура голосования: дескриптор $f_j^i \in W(T_i)$ добавляет голос эталону T_i . Как правило, оценка при ранжировании выполняется без фильтрации – рассчитывается общее число близких дескрипторов либо оно же, нормированное к числу точек шаблона. Но можно выделить следующие случаи, возникающие при поиске ближайших:

1. дескриптор f_k входного изображения имеет много близких дескрипторов в целом
2. дескриптор f_k имеет много близких дескрипторов на конкретном шаблоне
3. дескриптор шаблона f_j^i является близким для многих дескрипторов входного изображения

Такие случаи обрабатываем следующим образом:

1. если для f_k число ближайших в радиусе больше порога, то локальный признак малоинформативен и не участвует в расчёте оценок
2. f_k участвует в оценке конкретного шаблона только 1 раз
3. дескриптор f_j^i шаблона может быть учтён в его оценке ограниченное число раз (не превышая порог)

Для выполнения такой обработки для каждой точки близкие дескрипторы сортируем в порядке увеличения расстояния. Расстояние между близкими по (3.2) f_k и f_j^i учитывается при расчёте оценок шаблонов:

$$S(T_i) = \sum_{k=0}^{|W(Q)|} \frac{r_{max} - |f_k - f_j^i|}{r_{max}}, \quad f_k \in W(Q), f_j^i \in W(T_i) \quad (3.3)$$

Шаблоны классов ранжируются по оценке (выше – лучше), затем отбираются K лучших кандидатов.

3.5.2 Отсечение классов по структурным текстам

Данный этап предложено выполнять до отсечения по локальным признакам в целях лучшей общей вычислительной эффективности. Отсечение рассматривается на примере МЧЗ при классификации ID документов.

Текст МЧЗ содержит большую часть информации о типе документа, в частности его вид (паспорт, виза и т.п.), страну и дату выдачи. Локализация зоны при знании геометрии типа позволяет восстановить локализацию документа. Тем не менее, в одно и то же время в обращении могут находиться насколько бланков, а сама зона может смещаться при погрешности печати, поэтому визуальное сопоставление остаётся необходимым.

Страна и вид документа закодированы в первых 5 символах верхней строки. На этапе построения поискового индекса добавляются записи соответствия вида $\langle XXXXX, \{..., T_i, ...\} \rangle$. Локализация зоны позволяет выделить точное расположение данных символов, что позволяет выполнить распознавание очень быстро. После выполнения OCR возможны 2 варианта:

1. символы задают код, которому соответствуют K_M шаблонов;
2. нет соответствия между символами и возможными кодами

Если $K_M < K$, то выполняется переход к геометрической валидации, иначе выполняется отсечение по локальным признакам среди K_M возможных типов. Если МЧЗ не обнаружен, то отсечение по локальным признакам проходит среди $N - K_M$ типов. В случае, когда среди классов все типы документов всех стран, содержащие МЧЗ, достигается более чем в 100-кратное снижение числа рассматриваемых классов на следующих этапах.

3.5.3 Геометрическая валидация

Локальные признаки истинного кандидата должны быть геометрически согласованы с локальными признаками входного изображения, т.е. образовывать «созвездия», похожие с точностью до преобразования камеры H . Для каждого из K отобранных кандидатов вычислим преобразование H , которое переводит точки эталона в точки входного изображения в пределах заданной

погрешности. Для гипотезы H пары точек с близкими дескрипторами считается верным сопоставлением (инлаером), если:

$$|H(p) - p'| < \delta, \quad p \in W(Q), \quad p' \in W(T). \quad (3.4)$$

В общем случае :

- входное изображение и шаблон связаны проективным (центрально-проективным) преобразованием, параметры которого могут быть вычислены по 4-м сопоставленным точкам.
- распределение шума, в данном случае - должно сопоставленных точек, неизвестно.

В отсутствие дополнительных данных для фильтрации сопоставлений, оценить параметры преобразования можно с помощью методов семейства консенсусных выборок (-SAC, RANSAC и модификации), рассмотренные в 1.5.

Как показано в [26], базовых параметров RANSAC – числа итераций и порога инлаеров – недостаточно для отсеивания ложных гипотез в подобных задачах и моделях. Рассмотрим типы ошибок и причины их возникновения в нашей задаче для построения решения:

- решение плохо обусловлено - точки, порождающие гипотезу, недостаточно удалены друг от друга (на входном изображении, шаблоне или в обоих случаях)
- решение недопустимо либо не имеет смысла - например, преобразование невыпуклое, вырожденное, либо не соответствует ни одному из возможных поворотов камеры

Данные проблемы можно решить, введя ограничения на выборки и допустимые параметры модели. Скорость оценки также может быть улучшена. Классический алгоритм RANSAC подразумевает, что для каждой гипотезы подсчитывается число инлаеров. Таким образом, отсеивая невалидные выборки на раннем уровне, можно не только повысить качество метода, но и ускорить его. Помимо этого, современные платформы позволяют ускорения за счёт архитектуры. Например, при поддержке расширения SIMD, CPU может обрабатывать несколько элементов данных одновременно. Данные помещаются в векторы шириной 128, 256 или 512 бит и обрабатываются специализированными встроенными командами (интринсики x86 SSE, ARM NEON и т.д.), что позволяет ускорить вычисления в несколько раз. Большинство современных процессоров имеют возможности векторной обработки.

Еще одной важной особенностью процессоров является конструкция подсистемы памяти. Доступ к данным из оперативной памяти и их загрузка могут занимать сотни тактов, тогда как арифметические операции в десятки и сотни раз быстрее. Для решения этой проблемы используется иерархическая система кэширования, когда локальные блоки данных загружаются в кэш быстрого доступа.

Для части проблем независимо в ряде работ уже были предложены решения. Модульные фреймворки, такие как USAC (VSAC) позволяют объединить достижения различных SAC-методов в одном решении. Некоторые проверки выборок и моделей включены в USAC внутри модулей выборки минимального подмножества и генерации модели. Хотя USAC учитывает ряд важных практических и вычислительных соображений, тем не менее, он не использует векторизацию SIMD и обеспечивает лишь ограниченную локальность памяти.

С учтём вышеперечисленного, предложено решение в формате фреймворка для семейства методов на основе RANSAC, которая подходит для процессоров SIMD и сохраняет локальность памяти без ограничений для конкретного метода. Фреймворк параллельного эффективного выборочного консенсуса (PESAC) имеет блочную структуру, основанную на структуре USAC [113]. Структура PESAC расширена блоками-ограничителями, которые значительно сокращают количество выборок, передаваемых для расчета модели, и гипотез модели, подлежащих проверке, что значительно ускоряет время выполнения. Считаем такие проверки важным инструментом повышения точности и времени алгоритма, поэтому включаем их в PESAC отдельными блоками и демонстрируем их эффективность экспериментально.

PESAC состоит из следующих блоков:

- блоки выборки и ограничения выборки
- блоки расчета модели и ограничителя модели
- блоки проверки модели, раннего прекращения и уточнения модели

Схема структуры показана на рис. 3.3. Блоки, выделенные голубым фоном, являются необязательными и не включены в исходный RANSAC.

Особенностью PESAC является особый способ хранения входных данных, подходящий для удобной SIMD-векторизации. Чтобы сохранить локальность памяти, обрабатываем $\mathcal{K} \gg 1$ выборку за одну итерацию основного цикла. Следует отметить, что это может быть недостатком в случае, когда теоретически необходимое число \mathcal{I} итераций, определяемое уравнением (1.18), сравнимо с \mathcal{K}

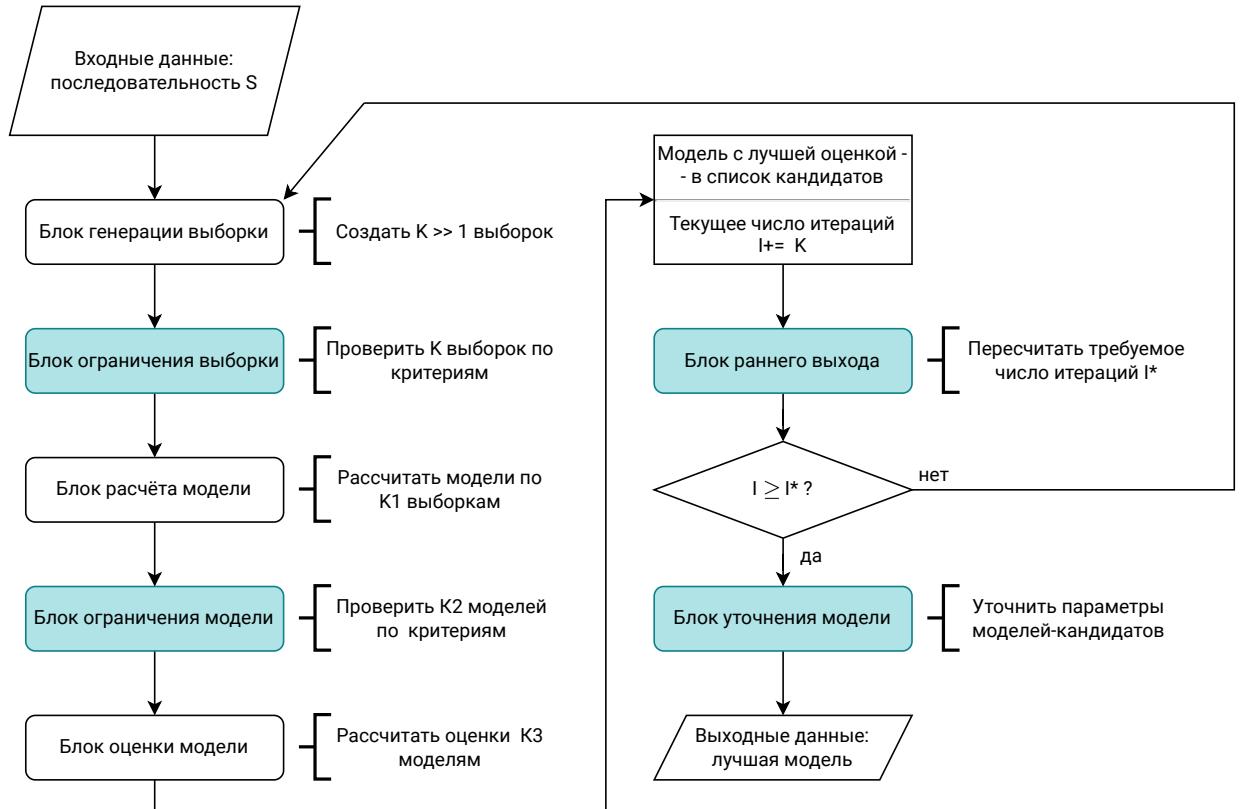


Рисунок 3.3 – PESAC – принципиальная схема.

и не кратно \mathcal{K} . В этом случае можно просто использовать больше выборок и дополнительные итерации. На практике, из-за зашумленности инлаеров, выборок требуется достаточно много (т.е. такой случай не наступает).

Входные данные. Последовательность \mathcal{S} состоит из сопоставлений. Сопоставление – пара точек $\langle p, p' \rangle$, где p – со входного изображения, p' – с шаблона, и данные – характеристики соответствия между ними. \mathcal{S} может храниться в памяти разными способами. Здесь предложен метод, обеспечивающий локальность в памяти и возможность щадящей обработки векторных данных:

$$\mathcal{S} = \{\mathcal{R}, \mathcal{Q}, \mathcal{SP}, \mathcal{EW}\}, \quad (3.5)$$

где $\mathcal{R} = \{r_1, \dots, r_{\mathcal{N}}\}$, $\mathcal{Q} = \{q_1, \dots, q_{\mathcal{N}}\}$, \mathcal{N} – длина последовательности, а точки из пары (r_i, q_i) должны преобразоваться друг в друга и образовать ребро в графе совпадений. Массивы \mathcal{SP} и \mathcal{EW} размера \mathcal{N} хранят вероятности и пространственные характеристики соответствий. Их использование не является обязательным.

Заметим, что этот метод хранения позволяет повторять точки в \mathcal{R} и \mathcal{Q} . Кроме того, точки могут быть разных типов (например, для задачи оценки положения камеры по соответствиям точек 2D–3D).

Выходные данные. Результатом SAC-методов является одна или несколько моделей и их характеристик, которые могут быть полезны для дальнейших приложений. Чаще всего это будут коэффициенты модели H и оценка C , но может быть и внешняя маска, индексы баллов, используемые для построения модели, и т.д.

Блок выборки. Принимает на вход последовательность \mathcal{S} соответствий и предназначен для генерации случайных n минимальных подмножеств совпадений, которые будут использоваться для дальнейшего расчёта модели. В наших обозначениях выборка представляет собой конечное множество $\{i_1, \dots, i_M\}$, $i_j \in \overline{1, N}, j \in \overline{1, M}$, индексов из последовательности \mathcal{S} , где M – минимальное количество совпадений, необходимое для вычисления гипотезы модели.

Выборка может быть разной:

- Исчерпывающая, когда выполняется полный перебор всех минимальных подмножеств совпадений в заданном порядке; этот вид актуален для небольшого количества соответствий, когда можно перебрать все подмножества;
- Равномерная, когда совпадения в выборке получены путём равномерной генерации M случайных чисел, т.е. индексов соответствий, от 1 до N ; этот вид используется, когда дополнительная информация о соответствиях отсутствует, а также использовался в исходном подходе RANSAC [106];
- Неоднородная, когда при выборе совпадений учитываются дополнительные характеристики соответствий, чтобы раньше генерировать более предпочтительные выборки; это самая современная стратегия отбора проб, за которой последовали модификации PROSAC [111], P-NAPSAC [114] и других.

Заметим, что блок генерирует K выборок за одну итерацию основного цикла. Использование $K \gg 1$ позволяет векторизовать генерацию случайных чисел непосредственно в этом блоке и далее задействовать расширения SIMD.

Блок ограничения выборки. Расчёт модели может быть трудоёмким. Если конкретная выборка малоосмысlena в контексте решаемой проблемы, то едва ли по ней можно рассчитать точную модель. Мы можем избежать ненуж-

ного расчёта параметров модели, добавив несколько предварительных проверок для образцов. Подборка ограничителей выборки направлена на отбраковку «плохих» выборок с минимальными вычислительными затратами. Из \mathcal{K} начальных выборок формируется $\mathcal{K}_1 < \mathcal{K}$ валидных. Проверки должны включать относительно простые критерии валидности, в сравнении с расчётом модели и оптимизациями, если таковые выполняются.

В существующих работах также предлагались геометрические ограничения на следующие свойства:

- относительный порядок (ориентация) точек выборки для оценки аффинного преобразования и 2D-гомографии [149; 150];
- площадь фигур, образованных точками образца: треугольники в аффинном преобразовании [151], четырёхугольники в оценке 2D-гомографии [152];

Блок расчёта модели. Для \mathcal{K}_1 выборок строится \mathcal{K}_2 моделей. Отмечу, что \mathcal{K}_2 может быть не равно \mathcal{K}_1 , если какие-то выборки не порождают модель, либо наоборот - порождают больше чем 1 (как в задаче расчёта фундаментальной и матриц по методам с 7 или 5 точками). Для $\mathcal{K}_1 \geq 1$ можно использовать векторизацию алгоритма расчёта параметров модели. Простота использования SIMD во многом зависит от конкретного преобразования - в нашем случае это 2D гомография. Благодаря совпадающей индексации в \mathcal{R} и \mathcal{Q} можно одновременно обрабатывать t элементов этих последовательностей. Значение t представляет собой количество значений, хранящихся в регистре SIMD, в зависимости от размера регистра и типа используемых данных и обычно варьируется от 2 до 16.

Блок ограничения модели. Отсев на этом уровне позволяет пропустить дорогостоящую проверку физически бессмысленных преобразований. Блок отбраковывает «плохие» с точки зрения решаемой задачи модели. Из \mathcal{K}_2 полученных моделей остаются \mathcal{K}_3 «хороших» перед этапом проверки модели. Геометрические ограничения зависят от конкретной задачи, в связанных работах были предложения для расчётов:

- 2D-гомографии: сохранение выпуклости путём проверки того, что эллипс внутри выпуклого четырёхугольника по-прежнему остаётся эллипсом при гомографии [153], хиральное неравенство [154].

Блок валидации модели. После получения \mathcal{K}_3 моделей из блока ограничения моделей, необходимо вычислить для них оценки и другую статистику.

Для каждого преобразования блок вычисляет последовательность N невязок, оценку модели $C(\theta)$ и вектор-маску инлаеров.

В исходной формулировке RANSAC функционал ошибки ρ берётся как в уравнении (1.16). Были предложены различные варианты ρ : увеличить вклад отдельного соответствия в общий балл ([107]); применяются вероятностные подходы (MLESAC [107]), моделирующие распределение вероятности ошибки по выбросам и выбросам для оценки гипотезы модели, а также методы максимального правдоподобия или оценки с использованием апостериорного максимума; чтобы исключить необходимость определения пользователем порога выбросов ([110]); и т. д. Независимо от этого, вычисление оценок и невязки также допускает векторизацию с помощью предложенного представления для \mathcal{R} и \mathcal{Q} . Поскольку объем входных данных может быть довольно большим, а оценку необходимо выполнять для каждого преобразования, проверка модели обычно является более дорогим и трудоёмким блоком, чем выборка и расчёт модели. В глобальном случае оценка модели C накапливается из значений некоторого функционала в каждом соответствии (например, функционал ошибки ρ из уравнения (1.16) и оценка из уравнения (1.17)). Используя SIMD-параллелизм благодаря совпадающей индексации в \mathcal{R} и \mathcal{Q} , можно одновременно обрабатывать t соответствий и обновлять оценку по t значениям.

Оптимизировать верификацию модели можно с помощью приёмов, направленных на досрочное отбрасывание якобы «плохих» гипотез модели (с сохранением гипотез $\mathcal{K}_4 < \mathcal{K}_3$) и при верификации определённой модели отбрасывание соответствий, которые гарантированно несовместимы с ней. В литературе описаны следующие подходы:

- тесты по подвыборке из d случайных сопоставлений;
- тест на базе последовательного критерия отношения вероятностей: для каждого соответствия рассчитывается отношение правдоподобия, и если оно становится больше определённого порога, то модель отбрасывается; и т.д.

Блок раннего выхода. Как правило, необходимое число итераций определяют экспериментально и дополняют методами раннего выхода, т.к. для точной оценки требуется знать долю выбросов, что редко случается на практике. Условие остановки зависит от используемой модификации RANSAC и может быть:

- адаптивным числом итераций, что их число \mathcal{I} переоценивается с помощью ур. (1.18), с учётом числа инлаеров лучшей в моменте модели;
- критерий случайности Вальда-Вольфовица или тесты, как в прошлом блоке;
- критерии из PROSAC и др. модификаций

Для реализации раннего выхода мы собираем некоторую статистику, с которой работает критерий (например, количество образцов, полностью сгенерированных и отклонённых ограничителем выборки, оценку лучшей на данный момент модели, ее поддержки и т. д.) во время выполнения основного цикла.

Блок уточнения модели. Этот блок предназначен для уточнения полученных \mathcal{K}_4 моделей-кандидатов. Мы выбираем определённое количество «лучших» моделей с точки зрения оценки и применяем к ним неминимальный оптимизатор. Таким образом, мы оптимизируем модель на основе большего подмножества входных данных, чем минимальная выборка, и уменьшаем влияние случайного шума на окончательную гипотезу. Здесь можно использовать различные стратегии, например:

- многомерная нелинейная оптимизация (метод Гаусса-Ньютона) и т.п.;
- анализ главных компонентов (PCA) для неминимальной оценки.

3.5.4 Реализация PESAC для проективного преобразования

Рассмотрим возможную конфигурацию фреймворка для решаемой задачи, в особенности - критерии в блоках ограничений. Выборка составляется на основе вероятностей с помощью методов генерации чисел по произвольному дискретному распределению. Т.к. требуется сгенерировать много чисел, необходим метод с низкой сложностью вычисления. Будем использовать метод псевдонимов со сложностью $O(1)$. Так же из соображений скорости и простоты векторизации для генерации псевдослучайных чисел использован генератор xoshiro128+.

Проективное преобразование в задаче локализации не может быть вырожденным или близко к тому, невыпуклым. Решение должно быть реализуемо камерой и устойчивым. Для отсечения на уровне выборок рассмотрим попарные

расстояния:

$$d_{ij}^r = |r_i - r_j|, \quad d_{ij}^q = |q_i - q_j|, \quad i, j = \overline{1, \mathcal{M}} \quad (3.6)$$

d_{\min} - порог для невырожденности и устойчивого решения, тогда:

$$d_{ij}^r > d_{\min}, \quad d_{ij}^q > d_{\min}, \quad i, j = \overline{1, \mathcal{M}}. \quad (3.7)$$

$$s_{ij} = \frac{d_{ij}^q}{d_{ij}^r}, \quad i, j = \overline{1, \mathcal{M}}. \quad (3.8)$$

s_{ij} - коэффициент масштабирования, тогда рассогласованность масштабов:

$$\left| \frac{s_{ij}}{\bar{s}} - 1 \right| < s_{\max}, \quad i, j = \overline{1, \mathcal{M}}, \quad (3.9)$$

где $\bar{s} = \binom{\mathcal{M}}{2}^{-1} \sum s_{ij}$. Визуализация выборок, которые будут отброшены по данным критериям, приведена на рис. 3.4. Параметры преобразования можно рассчитать по 4 соответствиям через единичный квадрат, как предложено в работе [11]. Для проверки выпуклости будем проверять точку пересечения диагоналей. Кроме того, исходя из ожидаемой устойчивости дескрипторов к масштабу, можно ввести порог на относительную площадь объекта на изображении.

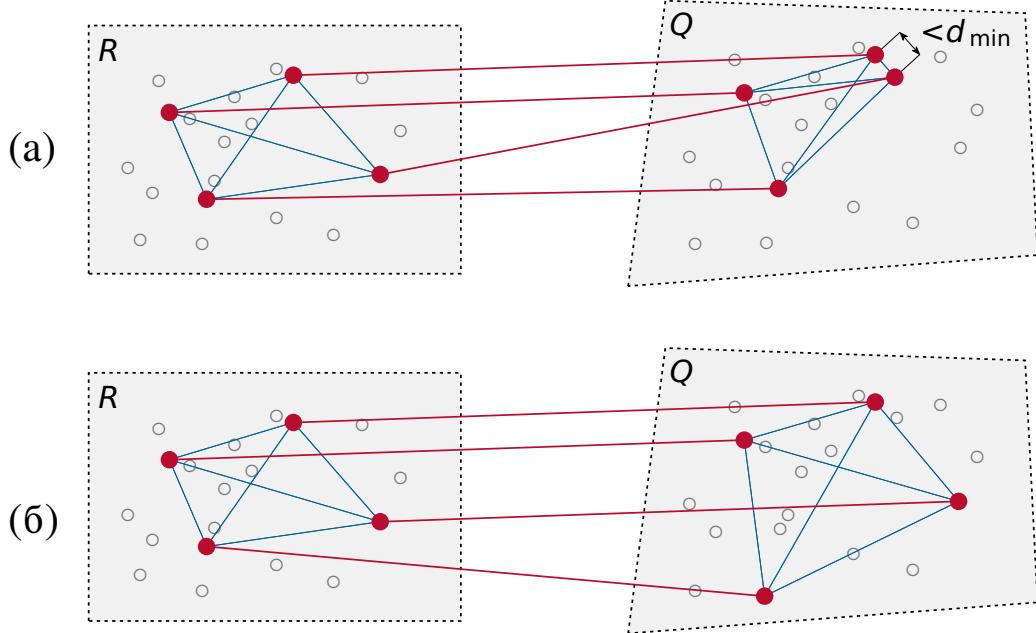


Рисунок 3.4 – Примеры выборок, которые будут отсеяны а) расстояния между точками меньше порога б) изменение масштаба рассогласовано между сторонами

Кроме того, в экспериментальной части будем использовать оценку MSAC для гипотез, ранний выход по адаптивному порогу, а так же РСА с неминимальным решателем для уточнения параметров модели.

3.5.5 Использование комбинированных входных данных

В случае, когда найденные в разделе 3.4.1 прямые-кандидаты или четырёхугольники задают корректный базис для нормализации или точную локализацию, преобразованием между точками запроса и точками эталона является изменение масштаба и сдвиг. Использование такой геометрической модели позволяет:

- ускорить алгоритм оценки преобразования за счёт увеличения скорости сходимости (поскольку для вычисления H достаточно 2 точек против 4 для проективного)
- уменьшить количество корректно сопоставленных запросов, отброшенных по порогу, т.к. дополнительные 2 точки из базиса проективного преобразования образуют корректные сопоставления.

Исходя из вышесказанного, будем рассчитывать H таким образом:

- если оценка четырёхугольника выше порога, то H – преобразование сдвиг + масштаб
- иначе, H – проективное преобразование

Дополним классическую итеративную схему RANSAC для точек с помощью выделенных четырёхугольников и прямых:

1. Каждый из геометрически корректных найденных четырёхугольников порождает дополнительную гипотезу, углы четырёхугольника сопоставляются с углами шаблона.
2. Из выделенных прямых составим пучки, определяющие точки схода.

Пара ортогональных точек схода определяют такое преобразование H_0

по [155], что $H = H_0 \times H_1$, где $H_1 = \begin{vmatrix} m_x & 0 & s_x \\ 0 & m_y & s_y \\ 0 & 0 & 1 \end{vmatrix}$ – остаточное преобразование сдвига (s_x, s_y) и масштаба (m_x, m_y) . С помощью RANSAC оценим H_1 – для построения гипотезы в данном случае достаточно 2 точек.

Обозначим как $G_i(H)$ число инлаеров гипотезы H для эталона T_i . Результатом считается гипотеза с наибольшим количеством инлаеров. При этом, если их количество меньше определённого порогового значения tr_G , тип объекта будет считаться неопределённым. В случае, когда два кандидата имеют одинаковое число инлаеров, рассчитывается остаточная оценка преобразования:

$$e_i(H) = \frac{1}{G_i(H)} \sum_{j=0}^{G_i(H)} \frac{|H(p_j^Q) - p_j^i|}{\delta}, \quad (3.10)$$

где $\delta > 0$, $p_j^Q \in W(Q)$, и $p_j^i \in W(T_i)$.

3.6 Эксперимент

3.6.1 Наборы данных

- **Artour** – содержит изображения объектов 2D искусства. Набор данных состоит из трёх частей:

- база эталонов, 10000 изображений
- положительная выборка, картина на фотографии присутствует в базе, 170 изображений (40 типов)
- отрицательная выборка, картина на фотографии в базе отсутствует, 30 изображений

В качестве эталонов выбраны сканированные изображения и фотографии, размещённые в открытом доступе (каталоги и онлайн-экспозиции [156]). Прототипами являются картины, в выборке представлено 25 различных жанров (импрессионизм, поп-арт, неоклассицизм и т.д.). Для экспериментальной оценки влияния числа типов на скорость и точность метода были выделены случайным образом поднаборы в 100 и 1000 изображений (из них 40 – эталоны классов положительной выборки).

Тестовые выборки составлены из изображений, размещённых различными пользователями в открытых источниках (GooglePhoto и т.п.), а также снимков, сделанных авторов диссертации и соавторами по работам [8; 9]. Изображения – фотографии, снятые на камеры мобильных устройств в музеях и галереях. На

Рисунок 3.5 — Примеры тестовых изображений из набора данных Artour



каждом изображении видно ровно одну картину, сцена может содержать иные объекты искусства, багетные рамы, людей. Пример представлен на 3.5.

- **MIDV-500** [142] описан в разделе 2.4.1.

- **MIDV-2020** [157] – набор изображений документов, основанный на 10 типах из MIDV-500. В наборе для каждого типа документа есть нескольких экземпляров документа с разными персональными данными. Подвыборка фото содержит 100 изображений разрешением 2268×4032 пикселей. Разметка составлена по аналогии с MIDV-500.

3.6.2 Оценка точности и скорости локализации и идентификации объектов

Для детекции использован алгоритм YAPE с радиусом 8 и порогом интенсивности 10. Предварительное сглаживание выполнено с помощью фильтра Гаусса с $\sigma_G = 1.0$. Ряд экспериментов проведён с использованием дескрипторов SURF [73], BinBoost [81], RFD [83], а также RFDoc [158] – бинарный дескриптор, разработанный с участием автора, в размерности 128 бит. Неавторские дескрипторы выбраны из соображений популярности, простоты внедрения и дальнейшей воспроизводимости результатов. Во всех замерах время поиска точек и дескрипторов на шаблонах и добавление их в поисковые структуры не учитывается.

Ввиду специфики используемых наборов данных, будем измерять дополнительный критерий точности локализации, предложенный в авторской публи-

кации [11]:

$$D(q, m) = \max_i \frac{\|m_i - q_i\|_2}{p_m}, \quad p_m = \sum_{j=1}^4 |m_j - m_{j+1}| \quad (3.11)$$

В выражении (3.11) q означает найденный четырёхугольник объекта, а m – корректный четырёхугольник из разметки. Критерий основан на том предположении, что при распознавании документов (наборы MIDV-500 и MIDV-2020) финальной целью, как правило, является распознавание полей данных.

Будем считать, что объект локализован правильно, если правильно определён его тип и выполняется:

$$\begin{aligned} D^* : D &< 0.02 \\ IoU^* : IoU &> 0.9 \end{aligned} \quad (3.12)$$

Скорость исполнения и качество алгоритма, показанные в таблицах 5 и 6, были достигнуты при следующих параметрах:

- 3 вариации масштаба ($1, 2/3, 1/2$), не более $5 \cdot 10^4$ точек на каждом
- дескриптор RFDoc 128-бит для окрестности размером 32×32 , порог $r_{max} = 32$
- ограничения выборки: минимальное расстояние между точками $d_{min} = 50$ (10% от минимальной стороны шаблонов), максимальная невязка масштаба $s_{max} = 0.5$
- ограничения модели: минимальная площадь $t_{area} = 0.05 w h$ для где $w \times h$ - размеры изображения; максимальное несоответствие геометрии по (2.7) $t_{ratio} = 0.5; t_{proj} = 0.5$ для оценки проективности $\frac{|H_{31}| + |H_{32}|}{|H_{33}|}$, где H - матрица преобразования 3×3 .
- без построения поискового индекса (линейное сравнение)

Статистики $\mathcal{K}_1 - \mathcal{K}_4$ 3.5.4, отражают количество промежуточных выборок и моделей, подлежащих анализу на каждом из этапов. В таблице 5 приведена зависимость их значения (а значит, и вычислительной сложности), от наличия ограничений.

Данные 5 показывают, что при введении ограничений и на выборки, и на модели, число моделей-кандидатов перед затратным этапом оценки по инлаерам уменьшается в более чем в 10^4 раз.

В таблице 6 приведены экспериментальные замеры для PESAC на процессорах разного типа с использованием SIMD-векторизации для и без. Точность

Таблица 5 — Таблица зависимости среднего числа промежуточных данных от наличия ограничений разных типов

Модификация	Статистики по блокам ограничений			
	$\bar{\mathcal{K}}$	$\bar{\mathcal{K}}_1$	$\bar{\mathcal{K}}_2$	$\bar{\mathcal{K}}_3$
PESAC, без ограничений	903398	903398	759982	759982
PESAC, ограничения выборки	918471	1342	1335	1335
PESAC, ограничения модели	918881	918881	773025	2081
PESAC, два ограничения	920113	1348	1341	91

Таблица 6 — Среднее время исполнения и точность PESAC на открытом наборе данных MIDV2020.

Модификация	Среднее время одного запуска, мс		Значения метрик		
	CPU: AMD FX-8350	CPU: ARM Cortex-A73	Точность	D^*	IoU [*]
	без SIMD → с SIMD	без SIMD → с SIMD			
PESAC, без ограничений	f32 13999.19 → 4631.29 (-66.91%) f64 12270.14 → 6577.60 (-46.39%)	f32 21592.45 → 11039.61 (-48.87%) f64 21592.45 → 19463.88 (-9.85%)	0.907	0.882	0.883
PESAC, ограничения выборки	f32 151.11 → 118.35 (-21.68%) f64 167.29 → 160.31 (-4.17%)	f32 251.14 → 231.27 (-7.91%) f64 300.00 → 291.17 (-2.94%)	0.933	0.907	0.909
PESAC, ограничения модели	f32 311.67 → 232.17 (-25.50%) f64 316.28 → 270.31 (-14.53%)	f32 535.78 → 445.45 (-16.85%) f64 688.45 → 638.33 (-7.28%)	0.943	0.916	0.919
PESAC, два ограничения	f32 136.08 → 114.31 (-15.99%) f64 151.03 → 148.38 (-1.75%)	f32 217.88 → 212.54 (-2.45%) f64 265.45 → 259.91 (-2.08%)	0.962	0.924	0.927

в таблице соответствует точности классификации. Для сравнения в таблице 7 приведены авторские замеры существующих открытых реализаций SAC-методов. Показано, что PESAC с предложенными критериями достигает наилучшей

Таблица 7 — Среднее время исполнения и точность SAC-методов на открытом наборе данных MIDV2020.

Модификация	Среднее время одного запуска RANSAC без вырождений, мс		Значения метрик	
	CPU: AMD FX-8350	CPU: ARM Cortex-A73	Точность	IoU
RANSAC (OpenCV)	24584.61	48374.03	0.903	0.878
USAC_DEFAULT (OpenCV)	321.96	485.31	0.905	0.887
USAC_PROSAC (OpenCV)	254.65	373.09	0.895	0.848
USAC_FAST (OpenCV)	318.10	452.33	0.908	0.887
USAC_MAGSAC (OpenCV)	369.09	593.07	0.909	0.879
Latent RANSAC	120.03	241.78	0.894	0.794
Space-Partitioning RANSAC	81.60	138.61	0.865	0.750

точности при показателях скорости, близких к наиболее быстрым последним

Таблица 8 — Точность классификации и локализации на MIDV-500

Метод	MIDV-500 фото		MIDV-500 сканы	
	Точность	D*	Точность	D*
[159]	0.725	-	0.622	-
(1)	0.461	0.417	0.318	0.268
(1)+АЧ2	-	-	0.676	0.590
(1)+АЧ1	0.760	0.700	-	-

Таблица 9 — Среднее время классификации и локализации на MIDV-500

Метод	Среднее время одного запуска, мс	
	iPhone 6 (Apple A8)	Intel Core i7-4770S
[159]	4220	1060
(1)+АЧ2	2820	780
(1)+АЧ1	350	100

модификациям. При этом ускорение за счёт SIMD выражено тем меньше, чем меньше заведомо ошибочных гипотез. Далее рассмотрим влияние поиска и анализа протяжённых признаков (прямых и четырёхугольников) на точность и скорость. Дополнительно был протестирован подход, описанный в [159]. Метод реализован согласно предоставленному в работе описанию, исходные изображения-эталоны и зоны с персональными данными идентичны авторским. Авторы используют точки и дескрипторы SURF, RANSAC и FLANN в реализации OpenCV, а так же проверки углов и сторон для полученного четырёхугольника. Для тестирования был выбран оригинальный дескриптор RFD 297-бит для окрестности размером 32×32 , порог $r_{max} = 60$, FLANN модифицирован согласно 3.3.2. Число проверяемых классов-кандидатов $K = 8$, прочие параметры идентичны. В таблицах 8 и 9 приведены результаты работы алгоритмов. Обозначим как (1) базовый замер с YAPE, RFD, FLANN и PESAC. Заметим следующее: а) хотя RFD показывает себя существенно слабей SURF из-за меньшей адаптивности к масштабу и повороту, это с превышением компенсируется за счёт использования протяжённых примитивов; б) несмотря на этапы выделения линий, четырёхугольников и 4 шаблона в индексе, предлагаемый метод существенно быстрей.

В эксперименте, показанном в таблице 10, PESAC и RANSAC отрабатывают одинаковое число итераций (не более 8000), т.е. ранний выход в PESAC не

Таблица 10 — Вклад примитивов и ограничений на точность и скорость метода, подвыборка MIDV-500

Дескриптор	Параметры модификации			Точность	D*	Относительная скорость
	-SAC	Ограничения	Прямые и четырёхугольники			
SURF	RANSAC (OpenCV)		недоступно	82.08%	76.03%	100%
	PESAC	-	-	81.93%	76.54%	~87%
	PESAC	-	+	82.38%	77.91%	~115%
	PESAC	+	-	82.60%	76.66%	~21%
	PESAC	+	+	83.18%	78.15%	~29%
BinBoost-256	RANSAC (OpenCV)		недоступно	81.84%	74.61%	~100%
	PESAC	-	-	81.75%	76.13%	~87%
	PESAC	-	+	82.37%	77.75%	~115%
	PESAC	+	-	82.50%	76.37%	~21%
	PESAC	+	+	83.14%	78.03%	~29%
BinBoost-128	RANSAC (OpenCV)		недоступно	77.21%	68.23%	~100%
	PESAC	-	-	77.00%	70.04%	~87%
	PESAC	-	+	78.60%	73.67%	~115%
	PESAC	+	-	79.01%	70.64%	~21%
	PESAC	+	+	80.19%	74.47%	~29%

Таблица 11 — Размер коллекции шаблонов с дескрипторами SURF и BinBoost

Дескриптор	Размер, бит	Объём чистый, Мб	Объём сжатый, Мб
SURF	2048	82.5	23.5
BinBoost-256	256	11.5	2.9
BinBoost-128	128	5.8	1.4

выполняется. Совокупность таблиц 10 и 11 показывает, что выделение протяжённых признаков можно повысить точность независимо от типа используемых дескрипторов. В то же время, разумно, что наибольшее повышение отмечается при слабых дескрипторах. Это позволяет достигать высокой точности при существенном ускорении расчётов на всех этапах, а также при более компактной базе шаблонов (в 6 раз меньше памяти относительно SURF требует BinBoost-256).

В заключении рассмотрим классификацию снимков картин. Отличие данного замера в том, что можно проследить, как меняется точность и скорость построенного метода в зависимости от допустимого числа классов. Использован метод **(1)+АЧ1**.

Таблицы 12 и 13 показывают следующее:

Таблица 12 — Точность классификации изображений Artour

Число типов	Проективная модель		Модель сдвиг-масштаб	
	без фильтра	фильтр точек	без фильтра	фильтр точек
100	0.958	0.988	0.964	0.994
1000	0.947	0.970	0.947	0.976
10000	0.917	0.941	0.917	0.964

Таблица 13 — Зависимость времени классификации от числа типов и типа модели

Число типов	Среднее время одного запуска на Snapdragon 625 (8× ARM Cortex-A53), мс	
	Проективная модель	Модель сдвиг-масштаб
100	280	255
1000	313	285
10000	361	333

- простая фильтрация точек за $O(n)$ их числа (по найденным четырёхугольникам, см. 3.4.2) позволяет уменьшить число ошибок классификации
- сочетание фильтрации и модель сдвига-масштаба позволяет компенсировать ошибки классификации, возникающие при росте числа допустимых типов.

Во всех случаях время работы метода на мобильном позволяет обрабатывать более 1 кадра в секунду, что даёт возможность дальнейшего улучшения за счёт интеграции покадровых результатов.

3.7 Выводы по главе

В данной главе предложен подход к построению SAC-методов, который включает обобщённые краевые условия, определяемые предметной областью и задачей. Для программной реализации подхода предложен формат представления данных, который обеспечивает локальность в памяти и возможность быстрой обработки с помощью векторных инструкций процессора.

Разработанный метод позволяет выбрать из сопоставленных признаков геометрически согласованный набор и оценить параметры преобразования. Реализация подхода для оценки проективного преобразования в рамках задачи демонстрирует более высокую скорость и точность, чем другие фреймворки и отдельные методы.

Предложен быстрый метод локализации и идентификации плоских объектов на основе сопоставления с шаблоном типа в компактном представлении. Для задания типа достаточно 1 изображения-эталона, шаблон включает локальные признаки (точки и их дескрипторы) и размеры эталона. Для уменьшения памяти, необходимой для хранения представлений предложено выделять такие протяжённые признаки, как прямые и четырёхугольники, перед локальными. Часть параметров преобразования (угловые искажения) оценивается через найденные четырёхугольники и оценки точек схода прямых. При реализации прямые и четырёхугольники получены как промежуточные и финальные результаты методов, предложенных в 2.2. Оценки параметров моделей выполнены с помощью реализации разработанного подхода (PESAC). На данных с объектами живописи и структурированными документами показано:

- введение протяжённых признаков повышает точность классификации и локализации, независимо от типа локальных признаков
- менее устойчивые, эффективные по сложности и памяти локальные признаки, в сочетании с протяжёнными, позволяют добиться точности выше, чем более информативные локальные признаки, допуская снижение затрат памяти в 6 раз без потерь.

Заключение

Основные результаты работы заключаются в следующем.

1. Для формирования изображения в модели камеры-обскуры и прообраза-прямоугольника с известными геометрическими характеристиками показано:

- сторона образа-четырёхугольника, отсутствующая по наблюдению, может быть восстановлена аналитически по трём наблюдаемым (найденным)
- можно оценить соответствие образа-четырёхугольника прообразу: объект-параллелограмм порождает только образы-четырёхугольники, углы которых лежат а) в плоскости на расстоянии равном фокусному от центра координат; б) на катетах четырёхгранного угла, образованного параллелограммом и центром координат.

2. Предложен метод локализации прямоугольного объекта при заданных зонах для его сторон в кадре с оригинальным фильтром границ. Предложен быстрый метод локализации прямоугольного объекта, произвольно размещённого на кадре или скане. В методе предложено сочетать границы разных детекторов для повышения точности за счёт типов границ, что подтверждено экспериментально. Показано, что использование при локализации оценки прообраза и восстановление стороны, повышают её устойчивость и точность.

3. Разработаны новые методы поиска текстовых признаков с известным ограниченным набором паттернов на примере МЧЗ а) при ограниченном повороте объекта (кадры мобильных устройств); б) при произвольном повороте (смешанный поток кадров, фото и сканов). При ограниченном повороте поиск основан на выделении строк на изображении, где символы представлены особыми точками. Предложено выделять с помощью преобразования Хафа кластеры точек, образующие прямые-строки, и фильтровать точки сцены. При произвольном повороте символы рассматриваются как точки - центры компонент связности. Элементы сцены фильтруются как компоненты по размеру и взаимной удалённости. Оба метода допускают отказ при отсутствии кластеров, похожих на паттерны. Показано, что известные паттерны текста в зоне позволяют выделить зону среди прочих без анализа и распознавания всех символов.

4. Предложено обобщение для семейства методов консенсусных выборок (PESAC). PESAC включает обобщённые краевые условия, определяемые пред-

метной областью и задачей. Также PESAC сохраняет локальность памяти и оперирует данными в специальном представлении, что позволяет использовать SIMD-векторизацию для более эффективного вычисления. Для программной реализации подхода предложен формат представления данных, который обеспечивает локальность в памяти и возможность быстрой обработки с помощью векторных инструкций процессора. Разработанный метод позволяет выбрать из сопоставленных признаков геометрически согласованный набор и оценить параметры преобразования. Реализация подхода для оценки проективного преобразования в рамках задачи демонстрирует более высокую скорость и точность, чем другие фреймворки и отдельные методы.

5. Разработан быстрый метод локализации и идентификации плоских объектов на основе сопоставления с пар изображений, представленных наборами характерных признаков (компактное представление). Для задания типа достаточно 1 изображения-эталона, модель включает локальные признаки - точки и их дескрипторы - и размеры эталона. Предложено выделять прямые и четырёхугольники перед локальными признаками для предварительной оценки угловых искажений и масштаба. Оценки параметров моделей выполнены с помощью реализации PESAC для проективного преобразования. Выполнены оригинальные исследования на данных с объектами живописи и структурированными документами, которые показали:

- введение протяжённых признаков повышает точность классификации и локализации, независимо от типа локальных признаков
- вычислительно эффективные локальные признаки (менее затратные по памяти, но и менее устойчивые), в сочетании с протяжёнными, позволяют добиться точности выше, чем более информативные локальные признаки, допуская снижение затрат памяти в 6 раз без потерь.

6. Показано, что реализация предложенных методов позволяет их использовать в режиме реального времени на мобильных устройствах. Результаты работы реализованы на C++ в качестве программных компонент и внедрены в программное обеспечение «Smart ARTour», «Smart IDReader» и «Smart Code Engine» компании ООО «Смарт Энджинс Сервис». Данное ПО интегрировано в мобильные и веб-приложения ряда компаний, в т.ч. АльфаБанк, ВТБ и др., а также является частью программно-аппаратного комплекса, который используется государственными структурами РФ.

Список литературы

1. Скорюкина, Н. С. Метод локализации машиночитаемых зон, инвариантный к условиям съёмки / Н. С. Скорюкина // Труды ИСА РАН. — 2017. — Т. 67, № 4. — С. 81—86.
2. Savelyev, B. I. A method for machine-readable zones location based on a combination the Hough transform and feature points / B. I. Savelyev, N. S. Skoryukina, V. V. Arlazarov // Bulletin of the South Ural State University, Series: Mathematical Modelling, Programming and Computer Software. — 2022. — Vol. 15, no. 2. — P. 100—110. — DOI: 10.14529/mmp220208.
3. Kolmakov, S. I. Machine-Readable Zones Detection in Images Captured by Mobile Devices' Cameras / S. I. Kolmakov, N. S. Skoryukina, V. V. Arlazarov // Pattern Recognit. Image Anal. — 2020. — Vol. 30, no. 3. — P. 489—495. — DOI: 10.1134/S105466182003013X.
4. Real Time Rectangular Document Detection on Mobile Devices / N. Skoryukina [et al.] // ICMV 2014. Vol. 9445. — 2015. — 94452A1—94452A6. — DOI: 10.1117/12.2181377.
5. Improved algorithm of ID card detection by a priori knowledge of the document aspect ratio / D. V. Tropin [et al.] // ICMV 2020. Vol. 11605. — 2021. — 116051F1—116051F9. — DOI: 10.1117/12.2587029.
6. Segments Graph-Based Approach for Document Capture in a Smartphone Video Stream / A. Zhukovsky [et al.] // ICDAR 2017. Vol. 1. — 2017. — P. 337—342. — DOI: 10.1109/ICDAR.2017.63.
7. Ключевые аспекты распознавания документов с использованием малоразмерных цифровых камер / Д. В. Полевой [и др.] // Вестник РФФИ. — 2016. — № 4. — С. 97—108. — DOI: 10.22204/2410-4639-2016-092-04-97-108.
8. Метод распознавания объектов живописи в неконтролируемых условиях с обучением по одному примеру / Н. С. Скорюкина [и др.] // Труды ИСА РАН. — 2018. — Т. 68, Спецвыпуск № S1. — С. 5—14. — DOI: 10.14357/20790279180501.

9. *Skoryukina, N. S.* 2D Art recognition in uncontrolled conditions using one-shot learning / N. S. Skoryukina, D. P. Nikolaev, V. V. Arlazarov // ICMV 2018. Vol. 11041. — 2019. — 110412E1—110412E8. — DOI: 10.1117/12.2523017.
10. *Skoryukina, N.* Fast method of ID documents location and type identification for mobile and server application / N. Skoryukina, V. V. Arlazarov, D. P. Nikolaev // ICDAR 2019. — 2020. — P. 850—857. — DOI: 10.1109/IC-DAR.2019.00141.
11. Document localization algorithms based on feature points and straight lines / N. Skoryukina [et al.] // ICMV 2017. Vol. 10696. — 2018. — 106961H1—106961H8. — DOI: 10.1117/12.2311478.
12. *Skoryukina, N.* Memory consumption reduction for identity document classification with local and global features combination / N. Skoryukina, V. V. Arlazarov, A. Milovzorov // Thirteenth International Conference on Machine Vision. Vol. 11605. — SPIE. 2021. — P. 416—423.
13. Impact of geometrical restrictions in RANSAC sampling on the ID document classification / N. Skoryukina [et al.] // ICMV 2019. Vol. 11433. — 2020. — P. 1143306-1—1143306-7. — DOI: 10.1117/12.2559306.
14. PESAC, the Generalized Framework for RANSAC-Based Methods on SIMD Computing Platforms / E. O. Rybakova [et al.] // IEEE Access. — 2023. — Vol. 11. — P. 82151—82166. — DOI: 10.1109/ACCESS.2023.3301777.
15. Fast Keypoint Filtering for Feature-Based Identity Documents Classification on Complex Background / N. Z. Valishina [et al.] // ICMV 2023. Vol. 13072. — 2024. — P. 1307205-1—1307205-9. — DOI: 10.1117/12.3023194.
16. *Andreopoulos, A.* 50 years of object recognition: Directions forward / A. Andreopoulos, J. K. Tsotsos // Computer vision and image understanding. — 2013. — Vol. 117, no. 8. — P. 827—891.
17. *Арлазаров, В.* Алгоритмы распознавания и технологии ввода текстов в ЭВМ / В. Арлазаров, О. Славин // Информационные технологии и вычислительные системы. — 1996. — № 1. — C. 48—54.

18. Automated bank cheque verification using image processing and deep learning methods / P. Agrawal [и др.] // Multimedia Tools and Applications. — 2021. — Т. 80. — С. 5319—5350.
19. *Rzemyk, T.* Biometrics in the criminal justice system and society today / T. Rzemyk // Effective Physical Security. — 2017. — С. 249—254.
20. Biometric recognition: Challenges and opportunities / N. R. Council, W. B. Committee [и др.]. — 2010.
21. *Zhou, L.* Computer Vision Techniques in Manufacturing / L. Zhou, L. Zhang, N. Konz // IEEE Transactions on Systems, Man, and Cybernetics: Systems. — 2023. — Т. 53, № 1. — С. 105—117.
22. *Ejiri, M.* Machine Vision in Early Days: Japan's Pioneering Contributions / M. Ejiri // Computer Vision – ACCV 2007. — Springer Berlin Heidelberg, 2007. — Р. 35—53.
23. SAS: Painting Detection and Recognition via Smart Art System With Mobile Devices / Z. Wang [и др.] // IEEE Access. — 2019. — Т. 7. — С. 135563—135572.
24. *Arora, R. S.* TitleTowards automated classification of fine-art painting style : PhD thesis / Arora Ravneet Singh. — New Brunswick : Rutgers University, 10/2012. — An optional note.
25. *Borko, H.* Automatic Document Classification / H. Borko, M. Bernick // J. ACM. — New York, NY, USA, 1963. — Апр. — Т. 10, № 2. — С. 151—162.
26. *Augereau, O.* Semi-structured document image matching and recognition / O. Augereau, N. Journet, J.-P. Domenger // Document Recognition and Retrieval XX. Т. 8658 / под ред. R. Zanibbi, B. Coüasnon. — International Society for Optics, Photonics. SPIE, 2013. — С. 865804.
27. Журавлев, Ю. Распознавание, классификация, прогноз: Математические методы и их применение / Ю. Журавлев, в. т. и. а. Российской академии наук Отделение информатики. — Nauka, 1989.
28. Анализ особенностей использования стационарных и мобильных малоразмерных цифровых видео камер для распознавания документов / В. В. Аплазаров [и др.] // ИТиВС / под ред. П. Ю. Соломонович. — Адрес: 119333, г. Москва, ул. Вавилова, д.44, корп.2, 2014. — № 3. — С. 71—81.

29. *Coughlan, J.* The Manhattan World Assumption: Regularities in Scene Statistics which Enable Bayesian Inference / J. Coughlan, A. L. Yuille // Advances in Neural Information Processing Systems. T. 13 / под ред. Т. Leen, Т. Dietterich, V. Tresp. — MIT Press, 2000.
30. *Davis, L. S.* A survey of edge detection techniques / L. S. Davis // Computer Graphics and Image Processing. — 1975. — Т. 4, № 3. — С. 248—270.
31. *Canny, J.* A Computational Approach to Edge Detection / J. Canny // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1986. — Т. PAMI—8, № 6. — С. 679—698.
32. *Melin, P.* An improved method for edge detection based on interval type-2 fuzzy logic / P. Melin, O. Mendoza, O. Castillo // Expert Systems with Applications. — 2010. — Т. 37, № 12. — С. 8527—8535.
33. *Evans, A.* A morphological gradient approach to color edge detection / A. Evans, X. Liu // IEEE Transactions on Image Processing. — 2006. — Т. 15, № 6. — С. 1454—1463.
34. *Versaci, M.* Image edge detection: A new approach based on fuzzy entropy and fuzzy divergence / M. Versaci, F. C. Morabito // International Journal of Fuzzy Systems. — 2021. — Т. 23, № 4. — С. 918—936.
35. *Di Zenzo, S.* A note on the gradient of a multi-image / S. Di Zenzo // Computer Vision, Graphics, and Image Processing. — 1986. — Т. 33, № 1. — С. 116—125.
36. *Martin, D.* Learning to detect natural image boundaries using local brightness, color, and texture cues / D. Martin, C. Fowlkes, J. Malik // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2004. — Т. 26, № 5. — С. 530—549.
37. *Muntarina, K.* Notes on edge detection approaches / K. Muntarina, S. B. Shorif, M. S. Uddin // Evolving Systems. — 2022. — Т. 13, № 1. — С. 169—182.
38. Convolutional Oriented Boundaries: From Image Segmentation to High-Level Tasks / K.-K. Maninis [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2018. — Т. 40, № 4. — С. 819—833.
39. *Xie, S.* Holistically-Nested Edge Detection / S. Xie, Z. Tu // Proceedings of the IEEE International Conference on Computer Vision (ICCV). — 2015.

40. Have We Solved Edge Detection? A Review of State-of-the-Art Datasets and DNN Based Techniques / M. Mubashar [и др.] // IEEE Access. — 2022. — Т. 10. — С. 70541—70552.
41. *Grompone von Gioi, R.* A Brief Analysis of the Holistically-Nested Edge Detector / R. Grompone von Gioi, G. Randall // Image Processing On Line. — 2022. — Т. 12. — С. 369—377.
42. *Al-Amaren, A.* RHN: A Residual Holistic Neural Network for Edge Detection / A. Al-Amaren, M. O. Ahmad, M. N. S. Swamy // IEEE Access. — 2021. — Т. 9. — С. 74646—74658.
43. *Duda, R. O.* Use of the Hough transformation to detect lines and curves in pictures / R. O. Duda, P. E. Hart // Commun. ACM. — New York, NY, USA, 1972. — Янв. — Т. 15, № 1. — С. 11—15.
44. *Lo, R.-c.* Gray-scale hough transform for thick line detection in gray-scale images / R.-c. Lo, W.-H. Tsai // Pattern Recognition. — 1995. — Т. 28, № 5. — С. 647—661.
45. *Brady, M. L.* Fast parallel discrete approximation algorithms for the radon transform / M. L. Brady, W. Yong // Proceedings of the Fourth Annual ACM Symposium on Parallel Algorithms and Architectures. — San Diego, California, USA : Association for Computing Machinery, 1992. — С. 91—99. — (SPAA '92).
46. On a fast Hough/Radon transform as a compact summation scheme over digital straight line segments / D. P. Nikolaev [и др.] // Mathematics. — 2023. — Т. 11, № 15. — С. 3336-1—3336-22.
47. Эффективная реализация быстрого преобразования Хафа с использованием сопроцессора CPCА / Ф. А. Аникеев [и др.] // Программирование / под ред. А. А. Ишханович. — 2021. — № 5. — С. 3—11.
48. *Nevatia, R.* Linear feature extraction and description / R. Nevatia, K. Ramesh Babu // Computer Graphics and Image Processing. — 1980. — Т. 13, № 3. — С. 257—269.
49. *Kahn, P.* A fast line finder for vision-guided robot navigation / P. Kahn, L. Kitchen, E. Riseman // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1990. — Т. 12, № 11. — С. 1098—1102.

50. LSD: A Fast Line Segment Detector with a False Detection Control / R. Grompone von Gioi [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2010. — Т. 32, № 4. — С. 722—732.
51. *Akinlar, C.* EDLines: A real-time line segment detector with a false detection control / C. Akinlar, C. Topal // Pattern Recognition Letters. — 2011. — Т. 32, № 13. — С. 1633—1642.
52. *Suárez, I.* ELSED: Enhanced line Segment drawing / I. Suárez, J. M. Buenaposada, L. Baumela // Pattern Recogn. — USA, 2022. — Июль. — Т. 127, № C.
53. *Zhang, Y.* AG3line: Active grouping and geometry-gradient combined validation for fast line segment extraction / Y. Zhang, D. Wei, Y. Li // Pattern Recognition. — 2021. — Т. 113. — С. 107834.
54. End-to-End Object Detection with Transformers / N. Carion [и др.] // Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I. — Berlin, Heidelberg : Springer-Verlag, 2020. — С. 213—229.
55. Line Segment Detection Using Transformers without Edges / Y. Xu [и др.] // 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2021. — С. 4255—4264.
56. *Teplyakov, L.* LSDNet: Trainable Modification of LSD Algorithm for Real-Time Line Segment Detection / L. Teplyakov, L. Erlygin, E. Shvets // IEEE Access. — 2022. — Т. 10. — С. 45256—45265.
57. DeepLSD: Line Segment Detection and Refinement with Deep Image Gradients / R. Pautrat [и др.] // 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2023. — С. 17327—17336.
58. *Zhang, Z.* Whiteboard scanning and image enhancement : tech. rep. / Z. Zhang, L.-W. He ; Microsoft Research, Microsoft Corporation. — 2004. — P. 1—24. — MSR-TR-2003—39.
59. *Hartl, A.* Rectangular target extraction for mobile augmented reality applications / A. Hartl, G. Reitmayr // Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). — 2012. — С. 81—84.

60. Contour model based homography estimation of texture-less planar objects in uncalibrated images / Y. Zhang [и др.] // Pattern Recognition. — 2016. — Т. 52. — С. 375—383.
61. Comic storyboard extraction via edge segment analysis / Y. Wang [и др.] // Multimedia Tools and Applications. — 2016. — Т. 75. — С. 2637—2654.
62. Deep vanishing point detection: Geometric priors make dataset variations vanish / Y. Lin [и др.] // 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2022. — С. 6093—6103.
63. Machine Learning Techniques for Identity Document Verification in Uncontrolled Environments: A Case Study / A. Castelblanco [и др.] // Pattern Recognition / под ред. K. M. Figueroa Mora [и др.]. — Cham : Springer International Publishing, 2020. — С. 271—281.
64. Coarse-to-fine document localization in natural scene image with regional attention and recursive corner refinement / A. Zhu [и др.] // . Т. 22. — 2019. — С. 351—360.
65. HU-PageScan: a fully convolutional neural network for document page crop / R. B. das Neves Junior [и др.] // IET Image Processing. — 2020. — Т. 14, № 15. — С. 3890—3898.
66. *Sheshkus, A. V.* Houghencoder: neural network architecture for document image semantic segmentation / A. V. Sheshkus, D. P. Nikolaev, V. L. Arlazarov // IEEE ICIP 2020. — Washington, DC, United States : IEEE Computer Society, 2020. — С. 1946—1950.
67. *Morevec, H. P.* Towards automatic visual obstacle avoidance / H. P. Morevec // Proceedings of the 5th International Joint Conference on Artificial Intelligence - Volume 2. — Cambridge, USA : Morgan Kaufmann Publishers Inc., 1977. — С. 584. — (IJCAI'77).
68. *Harris, C.* A Combined Corner and Edge Detector / C. Harris, M. Stephens // Proceedings of the Alvey Vision Conference. — Alvey Vision Club, 1988. — С. 23.1—23.6.
69. *Mikolajczyk, K.* Scale & Affine Invariant Interest Point Detectors / K. Mikolajczyk, C. Schmid. — 2004. — Окт.

70. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry / Z. Zhang [и др.] // Artificial Intelligence. — 1995. — Т. 78, № 1. — С. 87—119. — Special Volume on Computer Vision.
71. Schmid, C. Local grayvalue invariants for image retrieval / C. Schmid, R. Mohr // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 1997. — Т. 19, № 5. — С. 530—535.
72. Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints / D. G. Lowe // Int. J. Comput. Vision. — USA, 2004. — Нояб. — Т. 60, № 2. — С. 91—110.
73. Bay, H. SURF: Speeded Up Robust Features / H. Bay, T. Tuytelaars, L. Van Gool // Computer Vision – ECCV 2006 / под ред. A. Leonardis, H. Bischof, A. Pinz. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2006. — С. 404—417.
74. Viola, P. Rapid object detection using a boosted cascade of simple features / P. Viola, M. Jones // Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. Т. 1. — 2001. — С. I—I.
75. LDAHash: Improved Matching with Smaller Descriptors / C. Strecha [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2012. — Т. 34, № 1. — С. 66—78.
76. Brief: Binary robust independent elementary features / M. Calonder [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2010. — С. 778—792.
77. ORB: An efficient alternative to SIFT or SURF / E. Rublee [и др.] // 2011 International Conference on Computer Vision. — 2011. — С. 2564—2571.
78. Leutenegger, S. BRISK: Binary Robust invariant scalable keypoints / S. Leutenegger, M. Chli, R. Y. Siegwart // 2011 International Conference on Computer Vision. — 2011. — С. 2548—2555.
79. Alahi, A. FREAK: Fast Retina Keypoint / A. Alahi, R. Ortiz, P. Vandergheynst // 2012 IEEE Conference on Computer Vision and Pattern Recognition. — 2012. — С. 510—517.

80. *Brown, M.* Discriminative Learning of Local Image Descriptors / M. Brown, G. Hua, S. Winder // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2011. — Т. 33, № 1. — С. 43—57.
81. Boosting Binary Keypoint Descriptors / T. Trzcinski [и др.] // 2013 IEEE Conference on Computer Vision and Pattern Recognition. — 2013. — С. 2874—2881.
82. *Freund, Y.* A desicion-theoretic generalization of on-line learning and an application to boosting / Y. Freund, R. E. Schapire // Computational Learning Theory / под ред. P. Vitányi. — Berlin, Heidelberg : Springer Berlin Heidelberg, 1995. — С. 23—37.
83. Receptive Fields Selection for Binary Feature Description / B. Fan [и др.] // IEEE Transactions on Image Processing. — 2014. — Т. 23, № 6. — С. 2583—2595.
84. BEBLID: Boosted efficient binary local image descriptor / I. Suárez [и др.] // Pattern Recognition Letters. — 2020. — Т. 133. — С. 366—372.
85. HPatches: A Benchmark and Evaluation of Handcrafted and Learned Local Descriptors / V. Balntas [и др.] // 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2017. — С. 3852—3861.
86. MatchNet: Unifying feature and metric learning for patch-based matching / X. Han [и др.] // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — 2015. — С. 3279—3286.
87. *Vassileios Balntas Edgar Riba, D. P.* Learning local feature descriptors with triplets and shallow convolutional neural networks / D. P. Vassileios Balntas Edgar Riba, D. P. Mikolajczyk // Proceedings of the British Machine Vision Conference (BMVC) / под ред. E. R. H. Richard C. Wilson, W. A. P. Smith. — BMVA Press, 09.2016. — С. 119.1—119.11.
88. Improved ORB Algorithm Using Three-Patch Method and Local Gray Difference / C. Ma [и др.] // Sensors. — 2020. — Т. 20, № 4.
89. *Richardson, A.* Learning convolutional filters for interest point detection / A. Richardson, E. Olson // 2013 IEEE International Conference on Robotics and Automation. — 2013. — С. 631—637.

90. *Barroso-Laguna, A.* Key.Net: Keypoint Detection by Handcrafted and Learned CNN Filters Revisited / A. Barroso-Laguna, K. Mikolajczyk // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2023. — Т. 45, № 1. — С. 698—711.
91. D2-Net: A Trainable CNN for Joint Description and Detection of Local Features / M. Dusmanu [и др.] // 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2019. — С. 8084—8093.
92. ASLFeat: Learning Local Features of Accurate Shape and Localization / Z. Luo [и др.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2020. — С. 6588—6597.
93. *Bentley, J. L.* Multidimensional binary search trees used for associative searching / J. L. Bentley // Commun. ACM. — New York, NY, USA, 1975. — Сент. — Т. 18, № 9. — С. 509—517.
94. *Muja, M.* Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration / M. Muja, D. G. Lowe // VISAPP 2009 - Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, Lisboa, Portugal, February 5-8, 2009 - Volume 1 / под ред. A. Ranchordas, H. Araújo. — INSTICC Press, 2009. — С. 331—340.
95. *Guttman, A.* R-trees: a dynamic index structure for spatial searching / A. Guttman // SIGMOD Rec. — New York, NY, USA, 1984. — Июнь. — Т. 14, № 2. — С. 47—57.
96. *Indyk, P.* Approximate nearest neighbors: towards removing the curse of dimensionality / P. Indyk, R. Motwani // Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing. — Dallas, Texas, USA : Association for Computing Machinery, 1998. — С. 604—613. — (STOC '98). — URL: <https://doi.org/10.1145/276698.276876>.
97. *Norouzi, M.* Fast search in Hamming space with multi-index hashing / M. Norouzi, A. Punjani, D. J. Fleet // 2012 IEEE Conference on Computer Vision and Pattern Recognition. — 2012. — С. 3108—3115.
98. *Jégou, H.* Product Quantization for Nearest Neighbor Search / H. Jégou, M. Douze, C. Schmid // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2011. — Т. 33, № 1. — С. 117—128.

99. *Johnson, J.* Billion-Scale Similarity Search with GPUs / J. Johnson, M. Douze, H. Jégou // IEEE Transactions on Big Data. — 2021. — Т. 7, № 3. — С. 535—547.
100. Neural ranking models for document retrieval / M. Trabelsi [и др.] // Inf. Retr. — USA, 2021. — Дек. — Т. 24, № 6. — С. 400—444.
101. Jointly Optimizing Query Encoder and Product Quantization to Improve Retrieval Performance / J. Zhan [и др.] // Proceedings of the 30th ACM International Conference on Information & Knowledge Management. — Virtual Event, Queensland, Australia : Association for Computing Machinery, 2021. — С. 2487—2496. — (CIKM '21).
102. *Stewart, C. V.* Robust Parameter Estimation in Computer Vision / C. V. Stewart // SIAM Review. — 1999. — Т. 41, № 3. — С. 513—537.
103. *Huber, P. J.* Robust Regression: Asymptotics, Conjectures and Monte Carlo / P. J. Huber // The Annals of Statistics. — 1973. — Т. 1, № 5. — С. 799—821. — (Дата обр. 19.06.2024).
104. A review on robust M-estimators for regression analysis / D. de Menezes [и др.] // Computers & Chemical Engineering. — 2021. — Т. 147. — С. 107254.
105. *Fair, R. C.* On the Robust Estimation of Econometric Models / R. C. Fair // Annals of Economic and Social Measurement, Volume 3, number 4. — NBER, 10.1974. — С. 667—677.
106. *Fischler, M. A.* Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography / M. A. Fischler, R. C. Bolles // Commun. ACM. — New York, NY, USA, 1981. — Июнь. — Т. 24, № 6. — С. 381—395.
107. *Torr, P.* MLESAC / P. Torr, A. Zisserman // Comput. Vis. Image Underst. — USA, 2000. — Апр. — Т. 78, № 1. — С. 138—156.
108. *Chum, O.* Locally Optimized RANSAC / O. Chum, J. Matas, J. Kittler // Pattern Recognition / под ред. B. Michaelis, G. Krell. — Berlin, Heidelberg : Springer Berlin Heidelberg, 2003. — С. 236—243.
109. *Barath, D.* Graph-Cut RANSAC / D. Barath, J. Matas // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. — 2018. — С. 6733—6741.

110. *Barath, D.* MAGSAC: Marginalizing Sample Consensus / D. Barath, J. Matas, J. Noskova. — 2019. — Июнь.
111. *Chum, O.* Matching with PROSAC - progressive sample consensus / O. Chum, J. Matas // 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Т. 1. — 2005. — 220—226 vol. 1.
112. *Brachmann, E.* Neural-Guided RANSAC: Learning Where to Sample Model Hypotheses / E. Brachmann, C. Rother // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). — 2019. — С. 4321—4330.
113. USAC: A Universal Framework for Random Sample Consensus / R. Raguram [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2013. — Т. 35, № 8. — С. 2022—2038.
114. MAGSAC++, a Fast, Reliable and Accurate Robust Estimator / D. Baráth [и др.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2020. — С. 1301—1309.
115. *Ivashechkin, M.* VSAC: Efficient and Accurate Estimator for H and F / M. Ivashechkin, D. Barath, J. Matas // 2021 IEEE/CVF International Conference on Computer Vision (ICCV). — 2021. — С. 15223—15232.
116. *Barath, D.* Graph-Cut RANSAC: Local Optimization on Spatially Coherent Structures / D. Barath, J. Matas // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2022. — Т. 44, № 9. — С. 4961—4974.
117. *Wei, T.* Adaptive Reordering Sampler with Neurally Guided MAGSAC / T. Wei, J. Matas, D. Barath. — 2023. — Окт.
118. RANSAC for Robotic Applications: A Survey / J. M. Martínez-Otzeta [и др.] // Sensors. — 2023. — Т. 23, № 1.
119. SuperGlue: Learning Feature Matching With Graph Neural Networks / P.-E. Sarlin [и др.] // 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2020. — С. 4937—4946.
120. LoFTR: Detector-Free Local Feature Matching with Transformers / J. Sun [и др.] // 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). — 2021. — С. 8918—8927.

121. GlueStick: Robust Image Matching by Sticking Points and Lines Together / R. Pautrat [и др.] // 2023 IEEE/CVF International Conference on Computer Vision (ICCV). — 2023. — С. 9672—9682.
122. Mobile visual search on printed documents using text and low bit-rate features / S. S. Tsai [и др.] // 2011 18th IEEE International Conference on Image Processing. — 2011. — С. 2601—2604.
123. Words Matter: Scene Text for Image Classification and Retrieval / S. Karaoglu [и др.] // IEEE Transactions on Multimedia. — 2017. — Т. 19, № 5. — С. 1063—1076.
124. Integrating Scene Text and Visual Appearance for Fine-Grained Image Classification / X. Bai [и др.] // IEEE Access. — 2018. — Т. 6. — С. 66322—66335.
125. Fine-grained Image Classification and Retrieval by Combining Visual and Locally Pooled Textual Features / A. Mafla [и др.] // 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). — 2020. — С. 2939—2948.
126. Кунина, И. Слепая компенсация радиальной дисторсии на одиночном изображении с использованием быстрого преобразования Хафа / И. Кунина, С. Гладилин, Д. Николаев // Компьютерная оптика. — 2016. — Т. 40, № 3. — С. 395—403.
127. Белогаев, А. А. Техника выделения ортотропных границ изображения и её применение к ряду задач автоматической классификации транспортных средств в видеопотоке / А. А. Белогаев, Е. Г. Кузнецова, Д. П. Николаев // ИТиС 2014. — ИППИ РАН, 2014. — С. 184—189.
128. Шахно, К. У. Сборник задач по элементарной математике повышенной трудности / К. У. Шахно. — Рипол Классик, 1965.
129. Hartley, R. Multiple view geometry in computer vision / R. Hartley, A. Zisserman. — USA : Cambridge University Press, 2000.
130. Segments Graph-Based Approach for Smartphone Document Capture / A. E. Zhukovsky [и др.] // ICMV 2015. Т. 9875 / под ред. A. V. P. R. D. Nikolaev. — Society of Photo-Optical Instrumentation Engineers (SPIE), 2015. — 98750P1—98750P7.

131. Использование графа пересечений в задаче обнаружения документа на изображении, полученном со смартфона / В. В. Арлазаров [и др.] // ИИиПР / под ред. С. И. Анатольевич, академик РАН. — 2016. — № 2. — С. 60—69.
132. *Organization, I. C. A.* Machine readable travel documents. Т. 9303 / I. C. A. Organization. — International Civil Aviation Organization, 2013.
133. *Visilter, Y. V.* Development of OCR system for portable passport and visa reader / Y. V. Visilter, S. Y. Zheltov, A. A. Lukin // Document Recognition and Retrieval VI. Т. 3651 / под ред. D. P. Lopresti, J. Zhou. — International Society for Optics, Photonics. SPIE, 1999. — С. 194—199.
134. *Bessmeltsev, V.* High-speed OCR algorithm for portable passport readers / V. Bessmeltsev, E. Bulushev, N. Goloshevsky // Графикон'2011. — 2011. — С. 29—32.
135. *Kwon, Y.* Verification of the Document Components from Dual Extraction of MRTD Information / Y. Kwon, J. Kim // Graphics Recognition. Recent Advances and New Opportunities, 7th International Workshop, GREC 2007, Curitiba, Brazil, September 20-21, 2007. Selected Papers. Т. 5046 / под ред. W. Liu, J. Lladós, J. Ogier. — Springer, 2007. — С. 235—244. — (Lecture Notes in Computer Science).
136. Проблемы распознавания машиночитаемых зон с использованием малоформатных цифровых камер мобильных устройств / К. Б. Булатов [и др.] // Труды ИСА РАН / под ред. П. Ю. Соломонович [и др.]. — 119312, г. Москва, проспект 60-летия Октября, д.9, к.501, 2015. — Т. 65, № 3. — С. 85—93.
137. *Lee, H.* Character recognition for the machine reader zone of electronic identity cards / H. Lee, N. Kwak // 2015 IEEE International Conference on Image Processing (ICIP). — 2015. — С. 387—391.
138. *Тлебалдинова, А. С.* ДЕТЕКТИРОВАНИЕ МАШИННО-ЧИТАЕМОЙ ЗОНЫ ID-ДОКУМЕНТОВ В РЕЖИМЕ ОНЛАЙН / А. С. Тлебалдинова, Е. Б. Куандык, М. С. Асылбекова // ЛУЧШАЯ НАУЧНАЯ СТАТЬЯ 2018. — 2018. — С. 12—16.

139. *Hartl., A.* Real-time Detection and Recognition of Machine-Readable Zones with Mobile Devices / A. Hartl., C. Arth., D. Schmalstieg. // Proceedings of the 10th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2015) - Volume 1: VISAPP. — INSTICC. SciTePress, 2015. — C. 79—87.
140. *Lukoyanov, A.* Modification of YAPE keypoint detection algorithm for wide local contrast range images / A. Lukoyanov, D. Nikolaev, I. Konovalenko // Tenth International Conference on Machine Vision (ICMV 2017). T. 10696. — International Society for Optics, Photonics. SPIE, 2018. — C. 305—312.
141. ICDAR2015 competition on smartphone document capture and OCR (SmartDoc) / J. Burie [и др.] // 2015 13th International Conference on Document Analysis and Recognition (ICDAR). — 2015. — C. 1161—1165.
142. MIDV-500: A Dataset for Identity Document Analysis and Recognition on Mobile Devices in Video Stream / V. V. Arlazarov [и др.] // Computer Optics / под ред. S. V. Aleksandrovich. — 151, Molodogvardeyskaya street, Samara, 443001, 2019. — T. 43, № 5. — C. 818—824.
143. SmartDoc 2017 Video Capture: Mobile Document Acquisition in Video Mode / J. Chazalon [и др.] // 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). T. 04. — 2017. — C. 11—16.
144. *An, J.* Rectification of planar targets using line segments / J. An, H. I. Koo, N. I. Cho // Mach. Vision Appl. — 2017. — T. 28, № 1/2. — C. 91—100.
145. *Puybareau, É.* Real-Time Document Detection in Smartphone Videos / É. Puybareau, T. Géraud // 2018 25th IEEE International Conference on Image Processing (ICIP). — 2018. — C. 1498—1502.
146. *Zhou, B.* Fine-Grained Visual Recognition in Mobile Augmented Reality for Technical Support / B. Zhou, S. Güven // IEEE Transactions on Visualization and Computer Graphics. — 2020. — T. 26, № 12. — C. 3514—3523.
147. *Krumnikl, M.* Facial Emotion Recognition for Mobile Devices: A Practical Review / M. Krumnikl, V. Maiwald // IEEE Access. — 2024. — T. 12. — C. 15735—15747.

148. *Simon, M.* Fine-grained classification of identity document types with only one example / M. Simon, E. Rodner, J. Denzler // 2015 14th IAPR International Conference on Machine Vision Applications (MVA). — 2015. — C. 126—129.
149. An effective rigidity constraint for improving RANSAC in homography estimation / D. Monnin [и др.] // Advanced Concepts for Intelligent Vision Systems: 12th International Conference, ACIVS 2010, Sydney, Australia, December 13-16, 2010, Proceedings, Part II 12. — Springer. 2010. — C. 203—214.
150. Speeding-up homography estimation in mobile devices / P. Márquez-Neila [и др.] // Journal of Real-Time Image Processing. — 2016. — Т. 11. — C. 141—154.
151. Accelerated RANSAC for Accurate Image Registration in Aerial Video Surveillance / J. Zheng [и др.] // IEEE Access. — 2021. — Т. 9. — C. 36775—36790.
152. *Chung, K.-L.* A Novel and Effective Cooperative RANSAC Image Matching Method Using Geometry Histogram-Based Constructed Reduced Correspondence Set / K.-L. Chung, Y.-C. Tseng, H.-Y. Chen // Remote Sensing. — 2022. — Т. 14, № 14.
153. *Nakano, G.* Algebraic Constraint for Preserving Convexity of Planar Homography / G. Nakano // 2021 International Conference on 3D Vision (3DV). — 2021. — C. 126—135.
154. *Moisan, L.* Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers / L. Moisan, P. Moulon, P. Monasse // Image Processing On Line. — 2012. — Т. 2. — C. 56—73.
155. Fast projective image rectification for planar objects with Manhattan structure / J. Shemiakina [и др.] // ICMV 2019. Т. 11433. — Society of Photo-Optical Instrumentation Engineers (SPIE), 2020. — 114331N1—114331N9.
156. WikiArt. Visual Art Encyclopedia [Электронный ресурс]. — URL: <https://www.wikiart.org> (дата обр. 19.09.2024).
157. MIDV-2020: a comprehensive benchmark dataset for identity document analysis / K. Bulatov [и др.] // Computer Optics. — 2022. — Т. 46, № 2. — C. 252—270.

158. RFDoc: memory efficient local descriptors for ID documents localization and classification / D. P. Matalov [и др.] // ICDAR 2021. Т. 12822. — 2021. — С. 209—224. — (Lecture Notes in Computer Science (LNCS)).
159. Complex Document Classification and Localization Application on Identity Document Images / A. M. Awal [и др.] // 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). Т. 1. — IEEE. 2017. — С. 426—431.

Приложение А

Патенты и свидетельства



US011574492B2

(12) United States Patent Skoryukina et al.

(10) Patent No.: US 11,574,492 B2
(45) Date of Patent: Feb. 7, 2023

(54) EFFICIENT LOCATION AND IDENTIFICATION OF DOCUMENTS IN IMAGES

G06V 10/758 (2022.01); G06V 30/418 (2022.01); G06V 30/10 (2022.01)

(71) Applicant: Smart Engines Service, LLC, Moscow (RU)

(55) Field of Classification Search

None

See application file for complete search history.

(72) Inventors: Natalya Sergeevna Skoryukina, Moscow (RU); Vladimir Viktorovich Arlazarov, Moscow (RU); Dmitry Petrovich Nikolaev, Moscow (RU); Igor Aleksandrovich Faradjev, Moscow (RU)

(56) References Cited

U.S. PATENT DOCUMENTS

6,310,984 B2 * 10/2001 Sansom-Wai G06V 10/24
382/173
8,705,836 B2 * 4/2014 Gorski G06V 30/274
382/137

(73) Assignee: SMART ENGINES SERVICE, LLC, Moscow (RU)

(Continued)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 113 days.

OTHER PUBLICATIONS

Attivissimo et al., "An automatic reader of identity documents." In 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 3525-3530. IEEE, 2019. (Year: 2019).*

(Continued)

(21) Appl. No.: 17/237,596

Primary Examiner — Feng Niu

(22) Filed: Apr. 22, 2021

(74) Attorney, Agent, or Firm — Procopio, Cory, Hargreaves & Savitch LLP

(65) Prior Publication Data

US 2022/0067363 A1 Mar. 3, 2022

(Continued)

(30) Foreign Application Priority Data

(57) ABSTRACT

Sep. 2, 2020 (RU) 2020129039

Efficient location and identification of documents in images. In an embodiment, at least one quadrangle is extracted from an image based on line(s) extracted from the image. Parameter(s) are determined from the quadrangle(s), and keypoints are extracted from the image based on the parameter(s). Input descriptors are calculated for the keypoints and used to match the keypoints to reference keypoints, to identify classification candidate(s) that represent a template image of a type of document. The type of document and distortion parameter(s) are determined based on the classification candidate(s).

(51) Int. Cl.

G06K 9/46 (2006.01)

G06K 9/66 (2006.01)

G06V 30/413 (2022.01)

G06K 9/62 (2022.01)

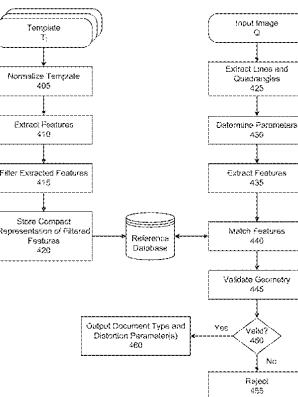
G06V 10/40 (2022.01)

(Continued)

(52) U.S. Cl.

CPC G06V 30/413 (2022.01); G06K 9/6232 (2013.01); G06V 10/40 (2022.01); G06V 10/44 (2022.01); G06V 10/757 (2022.01);

26 Claims, 6 Drawing Sheets





US010354142B2

(12) **United States Patent**
Arlazarov et al.(10) **Patent No.:** US 10,354,142 B2
(45) **Date of Patent:** Jul. 16, 2019(54) **METHOD FOR HOLOGRAPHIC ELEMENTS DETECTION IN VIDEO STREAM**(71) Applicant: **Smart Engines Service LLC**, Moscow (RU)(72) Inventors: **Vladimir Viktorovich Arlazarov**, Moscow (RU); **Timofey Sergeevich Chernov**, Dzerzhinsky (RU); **Dmitry Petrovich Nikolaev**, Moscow (RU); **Natalya Sergeevna Skoryukina**, Domodedovo (RU); **Oleg Anatolyevitch Slavin**, Moscow (RU)(73) Assignee: **SMART ENGINES SERVICE LLC**, Moscow (RU)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 152 days.

(21) Appl. No.: **15/668,455**(22) Filed: **Aug. 3, 2017**(65) **Prior Publication Data**

US 2018/0247125 A1 Aug. 30, 2018

(30) **Foreign Application Priority Data**

Feb. 27, 2017 (RU) 2017106048

(51) **Int. Cl.**
G06K 9/00 (2006.01)
G06K 9/46 (2006.01)

(Continued)

(52) **U.S. Cl.**
CPC **G06K 9/00711** (2013.01); **G06K 9/00442** (2013.01); **G06K 9/2054** (2013.01);
(Continued)(58) **Field of Classification Search**
CPC G06K 9/00; G06F 3/00
(Continued)(56) **References Cited**

U.S. PATENT DOCUMENTS

9,442,459 B2 * 9/2016 Dluhos G03H 1/0005

FOREIGN PATENT DOCUMENTS

CN 101915617 B 8/2012
CN 103196560 A 7/2013

(Continued)

OTHER PUBLICATIONS

Hartl, A., et al., AR-Based Hologram Detection on Security Documents Using a Mobile Phone, Springer International Publishing, 2014, pp. 335-346.

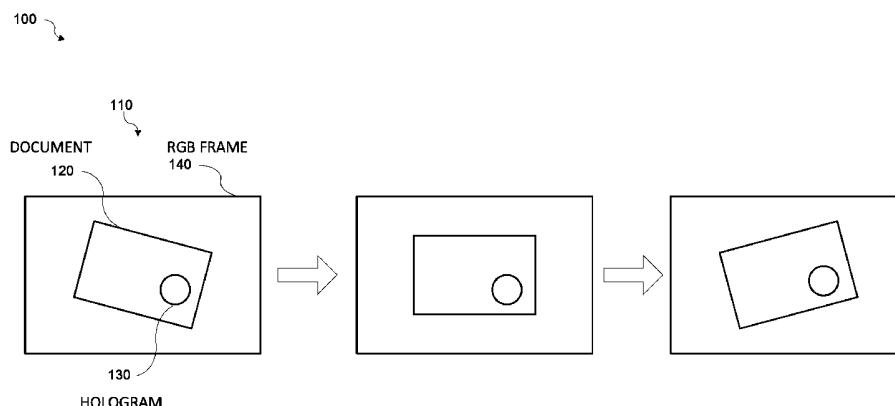
(Continued)

Primary Examiner — Abolfazl Tabatabai

(74) Attorney, Agent, or Firm — Procopio; Mark W. Catanese; Noel C. Gillespie

(57) **ABSTRACT**

A method for detecting holographic elements in a video stream containing images in the form of documents includes: processing of a video stream in which the document image is stabilized; constructing saturation and color tone maps; analyzing color characteristics in image regions; constructing histograms of color characteristics; estimating a change in the color characteristics at least in part based on data obtained by calculating a difference between the histograms of a current and a previous frame; constructing an integrated map of hologram presence estimates by combining calculated estimates for all video stream frames based at least in part on the estimation of the change in color characteristics; and determining final regions of the holographic elements based at least in part on the integrated map of the hologram presence estimates.

15 Claims, 4 Drawing Sheets

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2021610141

Программа анализа геометрических свойств локальных
особенностей на отдельных проекциях

Правообладатель: **Федеральное государственное учреждение
«Федеральный исследовательский центр «Информатика и
управление» Российской академии наук (ФИЦ ИУ РАН) (RU)**

Авторы: **Скорюкина Наталья Сергеевна (RU),
Путинцев Дмитрий Николаевич (RU)**

Заявка № 2020667463

Дата поступления **24 декабря 2020 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **12 января 2021 г.**

Руководитель Федеральной службы
по интеллектуальной собственности



 Г.П. Ильин

РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2022668729

Smart Template Matcher

Правообладатель: **Общество с ограниченной
ответственностью «Смарт Энджинс Сервис» (RU)**

Авторы: **Скорюкина Наталья Сергеевна (RU), Маталов
Даниил Павлович (RU)**

Заявка № **2022667955**

Дата поступления **29 сентября 2022 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **11 октября 2022 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Ю.С. Зубов



РОССИЙСКАЯ ФЕДЕРАЦИЯ



ПАТЕНТ

НА ПОЛЕЗНУЮ МОДЕЛЬ

№ 204787

Система удаленной регистрации абонентов сети связи с использованием мобильного устройства

Патентообладатель: **Общество с ограниченной
ответственностью "СМАРТ ЭНДЖИНС СЕРВИС"
(RU)**

Авторы: **Безматерных Павел Владимирович (RU), Арлазаров
Владимир Викторович (RU), Арлазаров Никита
Викторович (RU), Скорюкина Наталья Сергеевна (RU),
Славин Олег Анатольевич (RU)**

Заявка № 20211100924

Приоритет полезной модели **18 января 2021 г.**

Дата государственной регистрации
в Государственном реестре полезных
моделей Российской Федерации **10 июня 2021 г.**

Срок действия исключительного права
на полезную модель истекает **18 января 2031 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

Г.П. Ильин



РОССИЙСКАЯ ФЕДЕРАЦИЯ



СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018615952

Программа поиска плоских ригидных объектов "Smart ARTour"

Правообладатель: *Общество с ограниченной ответственностью
«Смарт Энджинс Сервис» (RU)*

Авторы: *Арлазаров Владимир Викторович (RU), Булатов
Константин Булатович (RU), Николаев Дмитрий Петрович
(RU), Скорюкина Наталья Сергеевна (RU)*

Заявка № **2018612805**

Дата поступления **23 марта 2018 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **18 мая 2018 г.**

Руководитель Федеральной службы
по интеллектуальной собственности

 Г.П. Ивлиев



Приложение Б

Акты о внедрении



ООО Смарт Энджинс Сервис 117312, город Москва,
ОГРН: 1167746085297 пр-кт 60-летия Октября, д. 9
ИНН: 7728328449

Т: +7 (495) 649-82-60
E: office@smartengines.ru
<https://smartengines.ru>

03.12.2024 № 127

На № от

АКТ

об использовании (внедрении) результатов диссертационной работы Скорюкиной Натальи Сергеевны «Методы локализации и идентификации плоских ригидных объектов на изображениях» в программных продуктах ООО «Смарт Энджинс Сервис»

Настоящий акт выдан Скорюкиной Н.С. для предоставления в диссертационный совет
24.1.224.01.

Результаты диссертационной работы «Методы локализации и идентификации
плоских ригидных объектов на изображениях» внедрены в семейство программных
продуктов распознавания изображений, а именно:

- Программа поиска плоских ригидных объектов «Smart ARTour»
- Smart ID Engine
- Smart Code Engine

Указанные программы внедрены в различных областях экономики и управления.
Широко задействован функционал распознавания документов, включая паспорта,
водительские удостоверения, визы, паспорта транспортных средств и пр.:

- для автоматизации процессов регистрации и обслуживания пользователей у
ведущих банков РФ – Альфа-банк, ВТБ, Т-банк, Газпромбанк, Росбанк,
Райффайзен и др.;
- для автоматизации процессов обслуживания пользователей в ряде страховых
компаний – Альфа-страхование, Ингострах и др.;
- для идентификации абонентов у операторов при продаже SIM-карт – МТС,
Билайн и Мегафон;
- для продажи билетов и регистрации пассажиров в транспортных и
авиакомпаниях – РЖД, SITA, Turkish Airlines, Croatia Airlines, RCCL.

Кроме того, программно-аппаратный комплекс на основе указанных программ
используется:

- в аэропорту Шереметьево для пересечения границы;
- системой изготовления паспортно-визовых документов ГС МИР;
- при регистрации самозанятых и ИП в мобильном приложении ФНС России.

Общее число организаций, использующих решения, основные на перечисленных
программах, составляет более 200 по всему миру.

Исполнительный директор
ООО «Смарт Энджинс Сервис»
Усилин Сергей Александрович

A

Назначение: в диссертационный совет 24.1.224.01

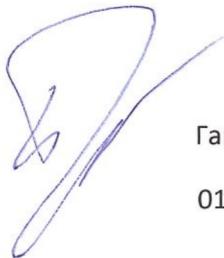
АКТ

об использовании (внедрении) результатов диссертационной работы Скорюкиной Натальи Сергеевны «Методы локализации и идентификации плоских ригидных объектов на изображениях» в АО «Альфа-Банк»

Результаты диссертационной работы «Методы локализации и идентификации плоских ригидных объектов на изображениях» обладают высокой значимостью и актуальностью для решения задачи автоматического ввода данных документов в информационных системах и приложениях, особенно на мобильных устройствах.

Методы локализации и идентификации (классификации) объектов на изображениях, разработанные Скорюкиной Н.С. и реализованные в составе программного обеспечения ООО «Смарт Энджинс Сервис», позволяют ускорить обработку документов в дистанционных каналах обслуживания и внутренних бизнес-процессах компании.

Руководитель департамента развития
цифровых каналов физических лиц



Гаврилов Борис Васильевич

01.12.2024

A

Назначение: в диссертационный совет 24.1.224.01

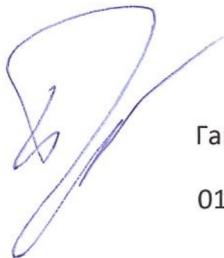
АКТ

об использовании (внедрении) результатов диссертационной работы Скорюкиной Натальи Сергеевны «Методы локализации и идентификации плоских ригидных объектов на изображениях» в АО «Альфа-Банк»

Результаты диссертационной работы «Методы локализации и идентификации плоских ригидных объектов на изображениях» обладают высокой значимостью и актуальностью для решения задачи автоматического ввода данных документов в информационных системах и приложениях, особенно на мобильных устройствах.

Методы локализации и идентификации (классификации) объектов на изображениях, разработанные Скорюкиной Н.С. и реализованные в составе программного обеспечения ООО «Смарт Энджинс Сервис», позволяют ускорить обработку документов в дистанционных каналах обслуживания и внутренних бизнес-процессах компании.

Руководитель департамента развития
цифровых каналов физических лиц



Гаврилов Борис Васильевич

01.12.2024



03.12.2024 Назначение:

в диссертационный совет 24.1.224.01

АКТ

об использовании (внедрении) результатов диссертационной работы Скорюкиной Натальи Сергеевны «Методы локализации и идентификации плоских ригидных объектов на изображениях» в АО «АльфаСтрахование»

Результаты диссертационной работы «Методы локализации и идентификации плоских ригидных объектов на изображениях» обладают высокой значимостью и актуальностью для решения задачи автоматического ввода данных документов в информационных системах и приложениях, особенно на мобильных устройствах.

Методы локализации и идентификации (классификации) объектов на изображениях, разработанные Скорюкиной Н.С. позволяют улучшить качество и скорость обработки данных в страховании. Авторские реализации методов на C++ в составе программного обеспечения ООО «Смарт Энджинс Сервис» внедрены и используются в системах и приложениях АО «АльфаСтрахование».

Директор
операционного департамента

Корецкая Н.Г.