

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ЯДЕРНЫЙ УНИВЕРСИТЕТ «МИФИ»

На правах рукописи



Сафонов Илья Владимирович

# МЕТОДЫ ОБРАБОТКИ ИЗОБРАЖЕНИЙ ДЛЯ СИСТЕМ СКАНИРОВАНИЯ И ПЕЧАТИ

2.3.8 – Информатика и информационные процессы

Диссертация на соискание учёной степени  
доктора технических наук

Научный консультант:  
доктор технических наук  
Матвеев Иван Алексеевич

Москва – 2025

## ОГЛАВЛЕНИЕ

Введение .....	5
Глава 1. Системы печати и сканирования.....	18
1.1. Основные определения .....	19
1.2. Краткий обзор устройств печати.....	24
1.3. Преобразование данных в системах сканирования и печати .....	30
1.4. Характеристики вычислительных платформ МФП .....	35
1.5. Оценки производительности и качества печати .....	37
1.6. Анализ требуемых новых режимов систем сканирования и печати .....	41
1.7. Формализация научных задач диссертационного исследования .....	43
Выводы по первой главе.....	45
Глава 2. Методология научно-исследовательской работы при создании решений для систем сканирования и печати.....	47
2.1. Существующие методологии исследований и разработки .....	47
2.2. Методология НИР при создании решений для МФП .....	51
Выводы по второй главе .....	59
Глава 3. Методика режимов сканирования.....	60
3.1. Систематизация искажений изображений при сканировании.....	61
3.2. Совокупность методов и последовательность обработки .....	64
3.3. Коррекция скоса.....	67
3.4. Сегментация на области текста, картинок и фона .....	75
3.5. Подавление печатного растра .....	87
3.6. Улучшение глобального и локального контраста .....	103
3.7. Примеры применения методологии НИР и методики режимов сканирования	110
3.7.1. Сканирование и сохранение документа в PDF с векторизацией контуров символов .....	110
3.7.2. Сканирование и копирование двухсторонних карточек.....	121

Выводы по третьей главе.....	135
Глава 4. Подход к реализации специальных режимов печати.....	137
4.1. Специальные режимы печати.....	137
4.1.1. Экономичная печать.....	138
4.1.2. Печать защитных элементов.....	140
4.2. Локальная модификация кода процессора растровых изображений.....	142
4.3. Примеры применения подхода к реализации режимов печати.....	145
4.3.1. Генерация микросимволов и печать микротекста.....	145
4.3.2. Способ экономичной печати.....	151
4.3.3. Режим черновой печати.....	158
4.3.4. Добавление в документ скрытых цифровых водяных знаков.....	164
Выводы по четвёртой главе.....	171
Глава 5. Подход к оценке и улучшению качества печатаемых фотографий.....	174
5.1. Типичные дефекты фотографий.....	175
5.2. Подход к автоматической коррекции фотографий.....	181
5.3. Примеры применения предложенного подхода.....	183
5.3.1. Способ оценки и повышения резкости.....	183
5.3.2. Детектирование и коррекция эффекта «красных глаз».....	193
5.3.3. Оценка и коррекция затемнённых областей.....	210
5.4. Эффективность методов автоматической оценки и улучшения качества печатаемых фотографий.....	226
Выводы по пятой главе.....	230
Глава 6. Преобразования для изменения размеров и соотношения сторон документов и фотографий.....	231
6.1. Существующие подходы.....	232
6.2. Преобразования для изменения соотношения сторон.....	234
6.3. Результаты.....	246
Выводы по шестой главе.....	251

Глава 7. Принципы автоматического выбора фотографий из набора и размещения изображений на странице .....	253
7.1. Выбор фотографий из коллекции.....	254
7.2. Размещение фотографий на странице .....	269
7.3. Формирование коллажа .....	274
Выводы по седьмой главе.....	281
Заключение .....	282
Список сокращений и обозначений .....	286
Список литературы .....	289
Приложение А. Акты о внедрении результатов работы .....	324

## Введение

**Актуальность темы исследования.** Несмотря на существующую тенденцию перехода на безбумажные технологии и электронный документооборот, Офисные устройства печати – принтеры широко используются как в промышленности и органах государственного управления, так и для персональных нужд. За последние 25 лет мировой рынок печатающих устройств, расходных материалов и сервиса печати продемонстрировал огромный рост [1]. Во второй декаде XXI века произошел массовый переход от принтеров, используемых исключительно для печати и реализованных как периферийные устройства, подключенные по локальному интерфейсу к персональному компьютеру (ПК), к многофункциональным устройствам (МФУ, или многофункциональным принтерам, МФП), позволяющим выполнять печать, сканирование, копирование и другие функции. При этом МФП способно работать как при подключении к ПК по локальному интерфейсу или по сети, так и в качестве самостоятельного устройства. В сочетании с прикладным и системным программным обеспечением (ПО), устанавливаемым на ПК, мобильные устройства, а также в локальной и глобальной сети, современные МФП представляют собой информационные системы сканирования и печати.

В настоящее время развиты базовые технологии офисных устройств печати, однако остаются нерешёнными задачи, связанные с автоматическим повышением качества отпечатков и сохраняемых электронных документов, особенно специальных видов, экономией расходных материалов, добавлением в напечатанные документы защитных элементов. Зачастую пользователь вынужден либо удовлетворяться результатами недостаточно высокого качества, либо вручную обрабатывать изображения с помощью дополнительного ПО, что неэффективно, так как ведёт к затратам времени и средств, требует специальных знаний. Рост мощности вычислительных платформ и бурное развитие теорий цифровой обработки изображений, компьютерного зрения и распознавания

образов привели к появлению большого количества способов обработки фотографий и изображений документов. Однако их применение в системах сканирования и печати часто затруднено в силу специфики процесса обработки информации (цветоделение, растривание и т.п.) и наличия в МФП существенно меньших вычислительных ресурсов по сравнению с ПК. Актуально улучшение потребительских свойств систем сканирования и печати за счёт совершенствования существующих и разработки множества новых автоматических режимов работы, удовлетворяющих требованиям к качеству и производительности, задаваемыми существующими стандартами.

Для модернизации существующих систем требуется добавление дополнительного функционала в существующие МФП за счёт модификации ПО, в частности встроенного ПО (анг. термин *firmware*), например, процессора растровых изображений (*raster image processor, RIP*). Возникает *противоречие в практике* между необходимостью добавления широкого набора новых режимов работы и ограниченными ресурсами МФП, в том числе размером памяти для хранения встроенного ПО.

Научная актуальность исследования проистекает из *противоречия в науке*, которое обусловлено следующими недостатками существующего научно-методического аппарата (НМА) разработки методов обработки изображений для модернизации систем сканирования и печати: каждый отдельный режим сканирования и печати разрабатывается как независимая уникальная научно-исследовательская работа (НИР) без выявления общих черт для классов сходных задач и без учёта уже используемых в МФП функций и архитектуры, а также типичных для подобных НИР рисков; в настоящее время глубоко проработанными являются методы обработки и улучшения изображений реализованные для ПК и предназначенные для просмотра результатов обработки на экране, тогда как улучшение напечатанных с помощью офисной техники изображений исследовано в недостаточной степени. Актуально развитие НМА, в частности формулировка

методологии НИР при создании решений для систем сканирования и печати, формализация методик, принципов и подходов к разработке методов обработки изображений, оптимизированных под возможности программно-аппаратного обеспечения МФП и удовлетворяющих требованиям к качеству и производительности.

Таким образом, *актуальным является изложение новых научно обоснованных технических и организационных решений* по разработке методов обработки изображений для систем сканирования и печати, внедрение которых вносит значительный вклад в развитие индустрии офисной печати страны.

**Степень разработанности темы.** Достижения ряда научных коллективов Российской Федерации в разработке теории и практических подходов к обработке и распознаванию цифровых фотографий и изображений документов имеют непосредственное отношение к рассматриваемым проблемам. В первую очередь, на материалы, представленные в диссертации, влияние оказали научные группы под руководством Ю.М. Баяковского (МГУ им. М.В. Ломоносова), Ю.И. Журавлёва (Вычислительный центр РАН), В.Л. Арлазарова (Институт системного анализа РАН), В.А. Сойфера (Институт систем обработки изображений РАН), В.Б. Бетелина (Научно-исследовательский институт системных исследований РАН), Л.П. Ярославского (Институт проблем передачи информации РАН), В.Н. Вапника (Институт проблем управления РАН).

Основной вклад в разработку технологий принтерной печати внесли американские и японские компании, в первую очередь, Xerox, HP, Lexmark, Epson, Canon и Ricoh. Среди учёных, внёсших заметный вклад в направления обработки, обсуждаемые в диссертации, можно выделить J.P. Allebach, R. Eschbach, R.L. de Queiroz, R. Bala, S.Wang, D.R. Tretter, R.A. Ulichney, H. Luo, J. Yen, N. Moroney, D. Shaked, C.B. Atkins, X. Lin, C.A. Bouman, H.Siddiqui, A.R. Rao, M. Ahmed, R. Ward, K. Seto, T. Kawana. Исследователями корпораций – производителей принтеров и МФП разработано большое число базовых функций

обработки изображений, применяемых в системах сканирования и печати. Тем не менее, существует много возможностей, как для модернизации существующих методов обработки, так и для изобретения и внедрения новых.

**Цель** диссертационного исследования состоит в улучшении потребительских свойств информационных систем сканирования и печати за счёт совершенствования существующих и разработки комплекса новых режимов работы, удовлетворяющих требованиям к качеству и производительности в условиях ограничений, накладываемых аппаратным обеспечением.

**Общая научная задача** – развитие научно-методического аппарата разработки и реализации методов обработки изображений для создания комплекса новых и модернизированных решений в системах сканирования и печати в условиях существующих ограничений и требований.

Общая научная задача декомпозируется на совокупность **частных научных задач**:

1. формулировка методологии научно-исследовательской работы при создании новых решений для систем сканирования и печати;
2. разработка методики (совокупности методов и последовательность их применения) для реализации множества режимов сканирования;
3. формулировка подходов к реализации специальных режимов печати, направленных на экономию расходных материалов и встраивания элементов защиты, и для автоматической слепой оценки и улучшения качества печатаемых фотографий;
4. разработка принципов автоматического выбора и размещения фотографий на странице;

**и частных прикладных задач:**

5. разработка модулей для режимов сканирования, которые корректируют типичные дефекты сканированных изображений и выполняют сегментацию на области текста, картинок и фона;



6. поиск и реализация технических решений для встраивания в процессор растровых изображений новых режимов печати;
7. разработка группы преобразований для изменения размеров и соотношения сторон документов и фотографий без модификации основных смысловых элементов изображения.

**Объектом исследования** являются информационные системы сканирования и печати.

**Предметом исследования** является набор потребительских свойств информационных систем сканирования и печати, а именно режимы сканирования, копирования и печати, связанные с обработкой изображений.

**Научная новизна:**

1. Сформулирована оригинальная спиральная методология научно-исследовательской работы при создании решений для систем сканирования и печати, которая включает в себя структуру и логическую организацию этапов исследования и разработки, и направлена на минимизацию риска отсутствия внедрения результата исследования в продукт, впервые выявлены и ранжированы факторы подобного риска.
2. Изложена методика (совокупность методов и последовательность их применения) режимов сканирования, основанная на предложенной автором комбинации следующих подходов: а) предварительная оценка параметров обработки выполняется по изображению низкого разрешения, затем изображение заданного разрешения обрабатывается без использования дополнительной памяти; б) корректируется скос оригинала, подавляется печатный растр и улучшается визуальное качество изображения; в) выполняется сегментация на области текста, картинок и фона; г) преобразованный результат сканирования сохраняется в метафайловый формат; на основе методики разработаны следующие новые режимы работы МФП: преобразование сканированных изображений в метафайловый формат с векторизацией

символов текста; копирование двухсторонних карточек с автоматическим определением ориентации оригинала; сканирование нескольких оригиналов небольшого размера; сканирование книг.

3. Предложен подход и оригинальные технические решения для реализации в процессоре растровых изображений режима защиты напечатанных документов от копирования за счёт генерации «на лету» микросимволов и микроизображений, что отличает его от известных решений компании Херох, а также режимов экономичной и черновой печати, режима печати скрытых цифровых водяных знаков.
4. Предложен подход для автоматической оценки и улучшения визуального качества печатаемых фотографий, который, в отличие от существующих методов, учитывает пространственное разрешение, размер фотокарточки и технологию печати; на основе данного подхода разработаны следующие новые автоматические способы обработки изображений: коррекции затемнённых участков изображения; детектирования и коррекции областей с эффектом «красных глаз»; оценки степени размытия фотографии и повышения резкости; подготовка для печати анаглифных изображений.
5. Предложена группа оригинальных преобразований для изменения размеров и соотношения сторон документов и фотографий без модификации основных смысловых элементов изображения, которая включает автоматические обрезку и переформатирование изображения за счёт удаления или добавления малозаметных участков, дополнения изображения симметрично отражёнными фрагментами; впервые сформулирован критерий заметности и предложен порядок применения преобразований.
6. На основе сформулированных принципов разработаны новые способы автоматического выбора изображений из коллекции и размещения фотографий на странице, при котором, в отличие от существующих алгоритмов размещения и упаковки, определяются коэффициенты масштабирования изображений;

предложен оригинальный критерий оптимальности размещения, максимизирующий занимаемую на странице площадь и минимизирующий разницу в размере фотографий.

#### **Теоретическая и практическая значимость:**

Теоретическая значимость работы заключается в развитии научно-методического аппарата разработки методов обработки изображений и новых режимов работы для систем сканирования и печати. Сформулированные положения и изложенные практические решения могут использоваться при разработке новых функций МФП. Ряд результатов диссертационного исследования обобщается на методы обработки изображений в иных областях промышленности.

Для предлагаемых методов обработки изображений во многих странах мира получены патенты на изобретение, что подтверждает их новизну, практическую значимость и промышленную применимость. В исследовательском центре компании Samsung и научно-исследовательском центре компании Шлюмберже получены акты о внедрении результатов диссертационного исследования. Результаты работы применялись в прототипах и коммерческих моделях печатающих устройств Samsung, в настоящее время применяются в ряде моделей принтеров компании HP. Предложенные алгоритмы обработки фотографий были адаптированы для ПО смартфонов и SmartTV Samsung.

**Методы исследования:** В работе широко применялись методы цифровой обработки изображений, компьютерного зрения, системного анализа, оптимизации, распознавания образов, включая машинное обучение, а также методы математической статистики для обработки результатов экспериментов. При реализации алгоритмов в виде программного обеспечения использовались технологии параллельных и векторных вычислений.

#### **Положения, выносимые на защиту.**

Итогом решения вышеуказанных научных и прикладных задач являются защищаемые положения, которые соответствуют полученным в диссертации

результатам, обладающим научной новизной, теоретической и практической значимостью, а также промышленной применимостью:

1. Комплекс режимов работы информационных систем сканирования и печати, удовлетворяющий требованиям к качеству и производительности в условиях ограничений, накладываемых аппаратным обеспечением многофункциональных принтеров. (п. 1 паспорта специальности 2.3.8)
2. Методология научно-исследовательской работы при создании технических и технологических решений для информационных систем сканирования и печати, включающая структуру и логическую организацию этапов исследования и разработки, направленная на минимизацию риска отсутствия внедрения результата исследования в продукт. (п. 1)
3. Методика режимов сканирования, включающая: сегментацию на фон, текст и картинки; устранение скоса; подавление печатного растра; улучшение качества изображения; конвертацию в метафайловый формат, а также разработанные на основе этой методики режимы копирования двухсторонних карточек и сохранения сканированного документа в метафайл с векторизацией контуров символов. (п. 3)
4. Подход к реализации специальных режимов печати, таких как черновая, экономичная, защиты от копирования и добавления цифровых водяных знаков, за счёт локальной модификации кода процессора растровых изображений. (п.17)
5. Подход к оценке и улучшению качества печатаемых фотографий, учитывающий размеры бумажного отпечатка, разрешение и технологию печати, а также разработанные на основе данного подхода методы коррекции затемнённых участков изображения, повышения резкости фотографии, коррекции изображений с эффектом «красных глаз». (п. 4)

**Достоверность** полученных результатов и рекомендаций подтверждается проведенными экспериментальными исследованиями, сравнением с существующими техническими решениями и результатами других исследователей,

опытом промышленной эксплуатации разработанных методов, а также апробацией и обсуждениями результатов на всероссийских и международных конференциях, научно-технических семинарах, публикациями в рецензируемых научных изданиях.

**Апробация работы:** Основные результаты диссертации докладывались на 26 конференциях: IEEE 10th International Symposium on Consumer Electronics – Санкт-Петербург, 2006; 16th International Conference on Computer Graphics and Vision, Graphicon – Новосибирск, 2006; 17th International Conference on Computer Graphics and Vision, Graphicon – Москва, 2007; IS&T/SPIE Electronic Imaging – Сан-Хосе, США, 2008; 31-я конференция молодых ученых и специалистов ИППИ РАН Информационные технологии и системы, ИТиС – Геленджик, 2008; 18th International Conference on Computer Graphics and Vision, Graphicon – Москва, 2008; 9th International conference on Pattern recognition and Image Analysis: New information technologies, PRIA – Нижний Новгород, 2008, IS&T/SPIE Electronic Imaging – Сан-Хосе, США, 2009; 19th International Conference on Computer Graphics and Vision, Graphicon – Москва, 2009; Electronic Imaging – Сан-Хосе, США, 2010; IEEE International Conference on Pattern Recognition, ICPR – Стамбул, Турция, 2010; 20th International Conference on Computer Graphics and Vision, Graphicon – Санкт-Петербург, 2010; 12-я Международная конференция и выставка Цифровая обработка сигналов и ее применение, DSPA – Москва, 2010; 10th International conference on Pattern recognition and Image Analysis: New information technologies, PRIA – Санкт-Петербург, 2010; IS&T/SPIE Electronic Imaging – Сан-Франциско, США, 2011; 21th International Conference on Computer Graphics and Vision, Graphicon – Москва, 2011; 13-я Международная конференция и выставка Цифровая обработка сигналов и ее применение, DSPA – Москва, 2011; Международная конференция Проектирование инженерных и научных приложений в среде MATLAB – Харьков, 2011; IS&T/SPIE Electronic Imaging – Сан-Франциско, США, 2012; 22th International Conference on Computer Graphics and Vision, Graphicon –

Москва, 2012; IS&T/SPIE Electronic Imaging – Сан-Франциско, США, 2013; IS&T/SPIE Electronic Imaging – Сан-Франциско, США, 2014; IS&T/SPIE Electronic Imaging – Сан-Франциско, США, 2015; Проблемы разработки перспективных микро- и нанoeлектронных систем (МЭС) – Зеленоград, 2016; 6th Mediterranean Conference on Embedded Computing, MECO – Бар, Черногория, 2017; 5-я международная конференция Информационные технологии и нанотехнологии, ITNT, Самара, 2019.

**Публикации:** По теме диссертации опубликовано 84 работы, из них:

а) 50 квалификационных работ, опубликованных в соответствии с требованиями п.11 «Положения о присуждении ученых степеней»:

- 7 в рецензируемых журналах, индексируемых Scopus (Q2 и Q3), которые в соответствии с Рекомендацией ВАК от 21.12.2023 приравниваются к публикациям в рецензируемых изданиях категории K1;
- 1 в журнале из списка ВАК (K1);
- 42 патента на изобретение, выданные в Российской Федерации и США, которые приравниваются к публикациям в рецензируемых изданиях, рекомендованных ВАК;

б) 34 работы других видов:

- 2 рецензируемые монографии, изданные издательством Springer, главы индексируются Scopus;
- 2 главы в книгах, главы которых индексируются Scopus;
- 21 материалы международных конференций, индексируемые Scopus и/или Web of Science;
- 9 доклады конференций.

**Личный вклад автора:**

В диссертацию включены положения и результаты, полученные либо лично автором, либо под его непосредственным руководством и при определяющем участии. В работах, опубликованных в соавторстве, личный вклад соискателя

состоит, по меньшей мере, в формулировке методики, методов, подходов, принципов и технических решений для обработки изображений в системах сканирования и печати.

**Соответствие паспорту специальности:** Тема исследования и полученные результаты соответствуют направлениям, перечисленным в пунктах 1, 3, 4, 13 и 17 паспорта специальности 2.3.8 – Информатика и информационные процессы.

**Структура и объём диссертации:** Диссертация состоит из введения, семи глав, заключения и приложения. Полный объём диссертации составляет 325 страниц, включая 107 рисунков, 38 таблиц. Библиография включает 359 источников.

В первой главе вводится формальное математическое определение рассматриваемой предметной области, приведено краткое описание типов систем сканирования и печати, процессов обработки данных в подобных системах, характеристик вычислительных платформ многофункциональных принтеров, подходов к оценке качества их работы и производительности; анализируются актуальные задачи обработки изображений в системах сканирования и печати; вводится математическая постановка научной задачи диссертации.

Во второй главе выполнен обзор существующих методологий выполнения научно-исследовательских работ (НИР) и проектов по разработке программного обеспечения, предлагается основанная на анализе и управлении рисками методология НИР по разработке решений для систем сканирования и печати.

В третьей главе выполнена систематизация искажений изображений, возникающих при сканировании и копировании, предложена методика (совокупность методов и подходов) режимов сканирования, которая включает этапы устранения основных дефектов сканированных изображений и проведение сегментации, необходимой для сохранения изображения в метафайл в соответствии с концепцией смешанного растрового содержимого; предложены способы оценки и коррекции скоса без использования дополнительной памяти,

подавления печатного раstra, улучшения локального контраста, сегментации изображения на области фона, текста и картинок; показаны примеры применения методологии НИР и методики работы режимов сканирования при создании таких новых функций МФП как сканирование и сохранение документа в PDF с векторизацией контуров символов и копирование двухсторонних карточек.

В четвёртой главе рассмотрены специальные режимы печати, предназначенные для экономии расходных материалов и добавления в напечатанный документ защитных элементов; сформулированы требования к режимам печати, реализуемым во встроенном программном обеспечении; предложен подход к реализации режимов специальной печати за счет локальной модификации кода процессора растровых изображений; идентифицированы модули в процессоре растровых изображений для добавления новых режимов печати; показаны примеры применения предлагаемого подхода при разработке режима печати микротекста, режимов экономичной и черновой печати, добавления в напечатанный документ скрытых цифровых водяных знаков.

В пятой главе проанализированы характерные дефекты цифровых фотографий, требующие автоматической коррекции; сформулированы требования к алгоритмам оценки качества изображений и исправления обнаруженных дефектов; предлагается подход к автоматическому улучшению качества печатаемых фотографий, учитывающий размеры бумажного отпечатка, разрешение и технологию печати; показаны примеры применения данного подхода для способа оценки и повышения резкости изображения, алгоритма детектирования и исправления эффекта «красных глаз», метода коррекции затемнённых областей на фотографиях повреждённых контрольным светом; приведены результаты экспериментов, показывающие эффективность предлагаемых методов.

В шестой главе рассмотрены способы преобразования изображений при копировании документов на бумагу отличающегося размера и при печати



фотографий без полей; предложена группа преобразований для изменения размеров и соотношения сторон сканированных документов и фотографий без модификации заметных фрагментов изображения; сформулированы принципы определения заметных участков; описаны способы автоматической обрезки, переформатирование изображений за счёт удаления или добавления малозаметных участков, метод дополнения изображений симметрично отраженными фрагментами.

В седьмой главе предложены принципы автоматического выбора фотографий из набора и их размещения на странице фотокниги, а также формирования коллажа из отобранных изображений; описан алгоритм размещения, позволяющий определять коэффициенты масштабирования изображений; сформулированы критерии оптимальности размещения; рассмотрен алгоритм смешивания перекрывающихся частей фотографий при создании коллажа; предложен способ перемножения прямоугольных матриц 32-разрядных чисел с фиксированной точкой с помощью 16-разрядных векторных инструкций; приведены результаты экспериментов, которые демонстрируют превосходство предлагаемых способов над существующими и хорошую корреляцию с мнением пользователей.

## Глава 1. Системы печати и сканирования

За последние полвека технологии печати и функциональность принтеров (печатающих устройств) существенно эволюционировали. Толковые словари конца XX века определяли значение слова «принтер» как «устройство вывода данных, которое преобразует закодированную информацию, выходящую из процессора, в удобную для чтения форму на бумаге» или «печатающее устройство, подключаемое к компьютеру» [2, 3]. В начале XXI века появились многофункциональные устройства, сочетающие функции принтера, сканера, копира и иногда факса [4]. Сначала подобные устройства называли многофункциональными периферийными устройствами, так как печать и сканирование могли осуществляться только при подключении к компьютеру с помощью интерфейса передачи данных. Позднее многие модели как многофункциональных, так и устройств, предназначенных только для печати (standalone), получили возможность подключения к локальной сети, работы с внешними носителями информации (например, USB флеш-накопителями), экран и элементы управления пользовательского интерфейса, что позволило применять эти устройства как при подключении к компьютеру, так и автономно. Поэтому определение «периферийное» перестало в полной мере отражать назначение устройств печати, и современные многофункциональные устройства называют многофункциональными принтерами (МФП, multifunctional printer, MFP), или многофункциональными устройствами (МФУ, multifunctional device, MFD) или устройствами «всё в одном» (all-in-one, AIO) [5].

Системой сканирования и печати называется совокупность программного и аппаратного обеспечения устройств офисной печати, включая предназначенное для работы с этими устройствами программное обеспечение компьютеров, сетевых и мобильных устройств. Для понимания методов обработки изображений систем сканирования и печати в первой главе вводится необходимая исходная информация: приводятся характерные для рассматриваемой области определения,

даётся краткий обзор устройств печати, описываются типичные потоки данных и особенности вычислительных платформ МФП, анализируются актуальные задачи обработки изображений в системах сканирования и печати, перечисляются подходы к оценке производительности и качества работы.

### 1.1. Основные определения

Введём основные определения и обозначения, используемые в работе. Математические обозначения выполнены в соответствии со стандартом ИСО 80000-2 [6].

При сканировании и печати обрабатывают растровые изображения и данные метафайловых форматов. Растровое изображение  $I = \{p_{rc}\}$ ,  $r \in [0, H - 1]$ ,  $c \in [0, W - 1]$  представляет собой прямоугольную матрицу, каждый элемент которой  $p_{rc}$  называется пикселем, где  $r$  – номер строки,  $c$  – номер столбца; матрица имеет размеры:  $H$  – количество строк (высота),  $W$  – количество столбцов (ширина) [7].

Пиксель  $p_{rc} = \{q_{rcl}\}$ ,  $l \in [1, K]$  может состоять из одной или нескольких компонент цвета в каком-либо цветовом пространстве, где  $l$  – номер канала,  $K$  – количество каналов [8]. Если  $K=1$ , то будем называть такое изображение одноканальным. Иногда при обращении к пикселям матрицы одноканального изображения индекс  $l$  опускается и используется обозначение  $I(r, c)$ .

Если  $K>1$ , то будем называть такое изображение многоканальным. Поскольку в данной работе каналы изображений используются исключительно для обозначения цветов, то будем называть многоканальные изображения цветными. Цветное изображение  $I$  также рассматривается как множество, состоящее из одноканальных изображений (или цветовых каналов) одинакового размера:  $I = \{I_l\}$ ,  $l \in [3, K]$ , в этом случае компоненты цвета  $q_{rcl}$  пикселя  $p_{rc}$  расположены в разных цветовых каналах в строке  $r$  и столбце  $c$ .

Значение компоненты цвета  $q_{rcl} \in [0, 2^{D_l} - 1]$ , где  $D_l$  – глубина цвета на канал, измеряемая в количестве бит на пиксел (bpp, bits per pixel). Глубина цвета  $D$  равна сумме глубин цвета на канал для всех каналов:  $D = \sum_{l=1}^K D_l$ .

Если  $D=1$  (очевидно,  $K=1$ ), то будем называть такое изображение бинарным. Если  $D > 1$  и  $K = 1$  и  $p_{rc}$  используется для обозначения уровня яркости, то будем называть такое изображение полутоновым или изображением в градациях серого.

Пусть цветное изображение состоит из трёх каналов: красного, зелёного и синего:  $I = \{I_R, I_G, I_B\}$ . В данной работе соответствующее ему полутоновое изображение (канал яркости) вычисляется либо как среднее арифметическое из значений цветовых каналов:

$$I_I(r, c) = \frac{I_R(r, c) + I_G(r, c) + I_B(r, c)}{3}, \quad (1.1)$$

либо как максимальное из значений цветовых каналов:

$$I_I(r, c) = \max(I_R(r, c), I_G(r, c), I_B(r, c)). \quad (1.2)$$

Гистограммой значений канала изображения  $I_l$  называется функция, определяющая для каждого  $i \in [0, 2^{D_l} - 1]$  количество пикселей, имеющих значение равно  $i$ :

$$h(i) = |\{(r, c) \mid I_l(r, c) = i\}|, \forall (r, c). \quad (1.3)$$

Назовём нормализованной гистограммой:

$$h'(i) = \frac{h(i)}{HW}. \quad (1.4)$$

Матрицей совместной встречаемости (co-occurrence matrix) канала изображения  $I_l$  называется функция, определяющая для каждого  $i, j \in [0, 2^{D_l} - 1]$  количество пикселей таких, что  $I_l(r, c) = i \wedge I_l(r + s_r, c + s_c) = j$ :

$$h_2(i, j) = |\{(r, c) \mid I_l(r, c) = i \wedge I_l(r + s_r, c + s_c) = j\}|, \forall (r + s_r) \in [0, H], (c + s_c) \in [0, W], \quad (1.5)$$

где смещения  $s_r, s_c \in \mathbb{Z}$ .

Назовём нормализованной матрицей совместной встречаемости:

$$h_2'(i, j) = \frac{h_2(i, j)}{(H - s_r)(W - s_c)}. \quad (1.6)$$

Операция свёртки канала изображения  $I_l$  с ядром свёртки  $\omega$  обозначается  $I_l * \omega$ . В ряде операций над изображениями могут получаться значения пикселей меньше 0 и больше  $2^{D_l} - 1$ , тогда пиксель считается равным 0 или  $2^{D_l} - 1$  соответственно. Дабы не перегружать формулы, функция контроля выхода значений пикселей за пределы диапазона  $[0, 2^{D_l} - 1]$  опущена.

Согласно [9], метафайл — это «файл, который содержит элементы изображения, выполненный с использованием как растровой, так и векторной графики». Современные метафайлы, например portable document format (PDF) [10] являются упорядоченной последовательностью текста, растровых изображений и векторной графики, содержат также дополнительную информацию, необходимую для вывода изображения на устройство отображения, например параметры страницы, данные используемых шрифтов, характеристики цветовых пространств. Абстрагируясь от формата хранения данных в памяти и дополнительной информации, обозначим данные метафайлового формата  $M$  в виде кортежа  $M = (R_1, R_2 \dots R_n)$ . Последовательная отрисовка (используются термины «проигрывание» и «рендеринг») элементов кортежа (записей метафайла)  $R_i$  приводит к формированию изображения на устройстве отображения. Запись метафайла  $R_i$  может быть либо строкой символов  $S$  с указанием шрифта и его атрибутов, либо последовательностью примитивов и команд векторной графики  $G$ , либо растровым изображением  $I$ :  $R_i \in \{S, G, I\}$ . Поскольку метафайл содержит данные метафайлового формата, то для его обозначения также будем использовать обозначение  $M$ .

Перечислим типы преобразований (функций), которые используются для обработки данных при сканировании, копировании и печати:

- преобразование растрового изображения в растровое без изменения размеров и глубины цвета:  $f_1: I \rightarrow I'$ ;
- преобразование растрового изображения в растровое с изменением размеров или глубины цвета:  $f_2: I \rightarrow I_n$ ;
- преобразование растрового изображения в данные метафайлового формата:  $f_3: I \rightarrow M$ ;
- преобразование данных метафайлового формата в растровое изображение:  $f_4: M \rightarrow I$ ;
- преобразование данных метафайлового формата в измененные данные метафайлового формата или другой формат:  $f_5: M \rightarrow M_n$ .

В данной работе рассматриваются преобразования, относящиеся к  $f_1$ ,  $f_2$  и  $f_3$ .

Пиксели изображений и элементы метафайловых форматов заданы в прямоугольной декартовой системе координат, причём каждый в своей. В преобразованиях  $f_2$  и  $f_3$  происходит также преобразование между системами координат. Если не указано иное, то начало координат располагается в левом верхнем углу изображения или страницы, на которую выводятся данные метафайлового формата. Ось абсцисс  $x$  направлена слева направо, ось ординат  $y$  направлена сверху вниз. Целочисленные координаты  $(r, c)$  используются для обозначения положения пикселей в изображении. Помимо целочисленных используются действительные координаты  $(x, y)$ , где целые  $y$  соответствуют  $r$ , а целые  $x$  соответствуют  $c$ .

Количество точек (dot), которое может быть напечатано на единицу длины, называют разрешением печати. Для принтеров разрешение печати обычно указывают в количестве точек на дюйм (dpi, dots per inch). Разрешение может быть изотропным, т.е. одинаковым по горизонтали и вертикали, или анизотропным. В данной работе все результаты приведены для изотропного разрешения печати. Точка является синонимом пикселя в смысле математического описания, данного выше. В зависимости от технологии печати, во многих случаях для передачи цветов

и оттенков требуется пиксел печатаемого изображения преобразовать в группу точек. Поэтому в ряде источников, например [11], различают разрешение, измеряемое в точках и в пикселах на дюйм. Кроме того, иногда в качестве аналога разрешения используется линеатура раstra (количество линий на дюйм) из традиционной типографской печати. Поскольку понятия линеатуры и разрешения в пикселах на дюйм применимы только для подмножества технологических решений печати, в данной работе под разрешением всегда понимается количество точек на дюйм.

Количество точек изображения, регистрируемых сканером на единицу длины, называют разрешением сканирования. Для сканеров разрешение указывается в dpi. В данном случае точка и пиксел являются синонимами. Большинство производителей в списке поддерживаемых разрешений сканирования помимо разрешений, на которых физически выполняется сканирование, указывают более высокие разрешения, получаемые в результате пространственной интерполяции данных. Будем называть оптическими или физическими те разрешения, на которых действительно выполнялось сканирование и не применялась интерполяция.

В данной работе сканирование и копирование считаются сходными операциями и часто рассматриваются как синонимы. Во-первых, сканирование является первым этапом копирования. Во-вторых, сохранённый в результате сканирования файл может быть впоследствии напечатан, что эквивалентно копированию.

Физический объект, помещаемый на стекло сканера, например страница документа, называется оригиналом. Получающаяся в результате копирования страница называется копией. Электронной копией или, просто, копией называется файл, хранящий изображение оригинала.

## 1.2. Краткий обзор устройств печати

Рассмотрим следующие подходы к классификации устройств печати:

- по цвету отпечатка,
- по технологии печати,
- по размеру бумаги,
- по варианту использования устройства.

По цвету отпечатка устройства печати подразделяются на цветные и чёрно-белые. Поскольку основным содержимым документов является чёрно-белый текст, то использование только чёрной краски для печати является экономически целесообразным, и чёрно-белые принтеры занимают значительную долю рынка [1]. Печать фотографий, красочных иллюстраций и многоцветной бизнес-графики производится с помощью цветных устройств. В данной работе в основном обсуждаются методы обработки изображений для систем цветной печати. Тем не менее, предлагаемые подходы и принципы, а также большинство рассматриваемых методов, применимы и для чёрно-белых принтеров.

За пятидесятилетнюю историю развития принтеров было изобретено несколько десятков технологий печати [12]. В настоящее время наиболее массовое применение имеют лазерные, светодиодные (LED), струйные (ink-jet) и термосублимационные (dye-sublimation) принтеры [5].

В лазерных принтерах статический заряд равномерно распределяется по поверхности фотобарабана, затем светодиодным лазером на фотобарабане формируется точечный рисунок печатаемой страницы, далее частички тонера (твёрдой краски) переносятся на фотобарабан и прилипают к участкам, ранее засвеченным лазером, затем тонер переносится на обычную офисную бумагу (с плотностью от 80 до 200 г/м<sup>2</sup>) и фиксируется с помощью нагрева. Светодиодные принтеры по принципу работы похожи на лазерные, но вместо лазера используют линейку светодиодов.



Чёрно-белый лазерный принтер печатает точки только чёрного цвета. Иллюзия присутствия в напечатанном изображении оттенков серого создается за счёт изменения плотности точек при печати с достаточно высоким разрешением. Например, каждый пиксел исходного изображения в оттенках серого можно заменить на матрицу, состоящую из нулей и единиц, где единицы соответствуют печатаемым точкам. Такой процесс называется генерацией полутонового печатного растра (halftone screen). Существует множество разнообразных алгоритмов создания печатного растра как для принтеров, так и для типографской печати [13].

Для цветной печати изображение из аддитивного цветового пространства, например RGB (red – красный, green – зелёный, blue – синий), должно быть преобразовано в субтрактивное [8]. В цветных лазерных принтерах используется цветовое пространство CMYK (cyan – голубой, magenta – пурпурный, yellow – желтый, key – ключевой черный цвет), где CMY являются дополнительными цветами к RGB соответственно, чёрный добавляется из экономических соображений, чтобы печатать чёрный текст и графику, а также создавать более насыщенные тёмные оттенки. При цветной лазерной печати последовательно печатаются растры точек четырёх цветовых каналов. На рисунке 1.1 показан пример печатного растра на изображении, напечатанном на цветном лазерном принтере с разрешением 600 dpi. Для получения данной иллюстрации напечатанная фотография была отсканирована с разрешением 2400 dpi.

В струйном принтере изображение также создается из точек нескольких красок, поэтому требуется преобразование изображения в печатный растр. Жидкие краски распыляются на бумагу в один проход с помощью матрицы дюз, расположенной в перемещаемой над бумагой каретке. Разрешение печати определяется не только точностью позиционирования каретки, но и размером пятна растекания (растискивания) краски на бумаге, которое зависит от объёма выпускаемой из дюзы краски (чернил), характеристик краски и бумаги. Поэтому

высокое разрешение достигается только для определённых сочетаний краски и бумаги. Использование ярких жидких красок и специальной фотобумаги для струйных принтеров позволяет печатать фотографии с насыщенными цветами и достаточно большим количеством оттенков. Для улучшенной передачи цветовых оттенков ряд моделей струйных принтеров помимо красок СМΥК используют от 2 до 8 дополнительных цветов, обеспечивая шести-, восьми-, или двенадцати-цветную печать [15, 16].



Рисунок 1.1 – Печатный растр на изображении, напечатанном на цветном лазерном принтере

Термосублимационные фотопринтеры позволяют печатать фотографии с регулировкой интенсивности оттенка каждой точки, и для этой технологии печати не требуется конвертация изображения в печатный растр. Между линейкой нагревательных элементов и специальной многослойной фотобумагой помещается лента, на которую нанесена краска. При нагревании краска испаряется с ленты и диффундирует в бумагу, которая валиком плотно прижата к ленте. Количество испаряемой краски и, соответственно, оттенок на бумаге зависят от приложенного количества теплоты. Обычно используются три краски СМΥ, которые наносятся

последовательно в 3 прохода, затем фотография покрывается прозрачной защитной плёнкой.

В Таблице 1.1 для рассмотренных типов принтеров приведены типичные значения разрешений и тип бумаги, указаны используемые цветовые пространства и глубина цвета на канал.

Таблица 1.1. Характеристики принтеров в зависимости от технологии печати

Тип принтера	Тип бумаги	Разрешение, dpi	Цветовые пространства	$D_L$ , bpp
Лазерный	Обычная офисная	600 ... 1200	СМΥΚ	1
Светодиодный	Обычная офисная	600 ... 1200	СМΥΚ	1
Струйный	Офисная, фотобумага для струйных принтеров	600 ... 2400	СМΥΚ+0...8 доп. канала	1
Термосублимационный фотопринтер	Специальная	300 ... 400	СМΥ	8

Различные устройства захвата изображений (сканеры и фотокамеры) и устройства отображения (дисплеи и принтеры) по-разному кодируют и показывают одинаковые для восприятия человеком цвета. В современных компьютерных системах, включая системы сканирования и печати, для, насколько это возможно, более точного воспроизведения цвета различными устройствами осуществляется управление цветом (color management) [17]. Для каждого устройства явно или по умолчанию устанавливается цветовой профиль, содержащий информацию для преобразования цветов между цветовым пространством данного устройства и аппаратно-независимым (так называемым, абсолютным) цветовым пространством CIE XYZ 1931 (МКО XYZ 1931) или CIE Lab 1976 (МКО Lab 1976) [18]. Однако, преобразования цвета через аппаратно-независимое цветового пространства для

ряда оттенков оказывается недостаточно, так как разные устройства и краски обеспечивают различный цветовой охват, то есть их множества цветов не совпадают полностью. Этот факт обычно иллюстрируют на графике, являющемся проекцией цветового пространства МКО XYZ 1931 на плоскость  $xu$  (нормализованные координаты  $XY$ ), см. рисунок 1.2 [8]. Цветная фигура, по форме похожая на подкову, содержит цвета, которые видит человек с нормальным цветовым восприятием. Внешняя сторона подковы – спектрально чистые цвета. Линиями красного, синего и жёлтого цвета показаны цветовые охваты пространств AdobeRGB, sRGB и CMYK (для некоторого определенного набора красок) [17].

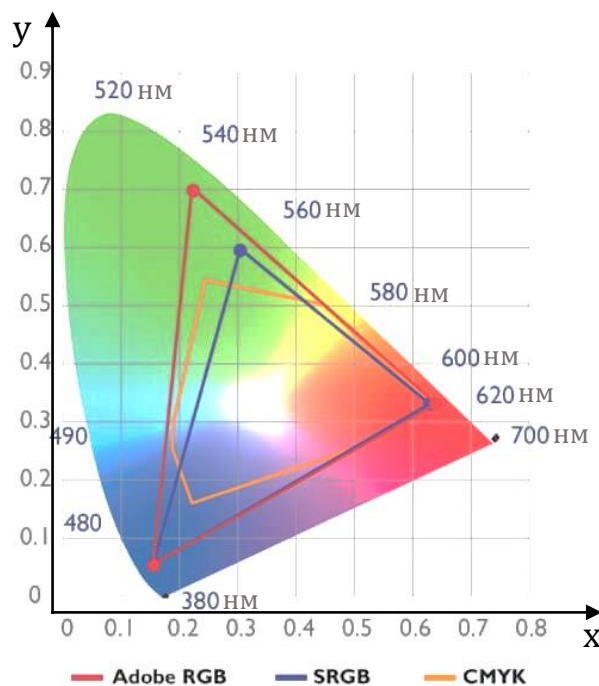


Рисунок 1.2 – Цветовой охват различных цветовых пространств в плоскости  $xu$  пространства МКО XYZ 1931

Для обработки цветов вне области пересечения множеств цветов устройств в профиле, помимо информации для преобразования из аппаратно-независимого цветового пространства, задаётся способ преобразования цветового охвата (gamut mapping). Несмотря на применение систем управления цветом, существуют цвета

на дисплее, которые не могут быть напечатаны без заметного изменения их оттенка, аналогично, существуют цвета на бумажных оригиналах, которые заметно искажаются при сканировании. Эти особенности необходимо учитывать при создании методов обработки изображений для сканирования и печати.

Устройства печати классифицируют по размеру и типу бумаги, на которой может быть выполнена печать, например, различают страничные и рулонные устройства. В данной работе рассматриваются методы, которые разрабатывались для страничных устройств, хотя ряд из них применим и для рулонных. В спецификации любого устройства печати можно найти перечисление поддерживаемых типов и форматов бумаги (например, см. [15, 16]). Существует большое количество стандартных размеров бумаги. Часть из них определяется международными стандартами, например ISO 216 [19], часть национальными стандартами. В Таблице 1.2 приведено несколько примеров форматов бумаги.

Таблица 1.2. Примеры форматов бумаги

Тип бумаги	Название формата	Размер, мм
Листы офисной бумаги	A3	297 x 420
	A4	210 x 297
	A5	148 x 210
	A6	105 x 148
	Ledger	279,4 x 431,8
	Letter	215,9 x 279,4
	Legal	215,9 x 355,6
Карточки	Фотография	100 x 150
	Фото L	90 x 130
	Фото 2L	130 x 180
Конверты	C5	162 x 229
	DL	110 x 220
	Monarch	98,4 x 190,5

По максимальному размеру оригинала для сканирования и копирования МФП подразделяются на устройства А3 и А4.

Устройства печати различаются набором функций, размером и ценой в зависимости от варианта использования. В начале века рынок устройств условно делился на два сегмента: устройства, предназначенные для небольших офисов и персонального использования (SOHO, small office/home office); устройства для предприятий (enterprise). В настоящее время выделяют 5 вариантов использования: дома для персональных нужд; для коммерческого использования в малом и среднем бизнесе; в образовательных организациях, в правительственных учреждениях, для крупных предприятий [1]. Базовая функциональность всех систем сканирования и печати схожа, но требования по производительности, надёжности и дополнительным функциям различаются в зависимости от варианта использования. На предприятиях массово используются высокопроизводительные сетевые А3 МФП, с возможностью двухстраничной печати и оборудованные автоподатчиком бумаги (automatic document feeder, ADF). Для них важными являются функции, связанные с экономией расходных материалов и информационной безопасностью. Для персонального домашнего использования, как правило, выбирают компактные А4 МФП с односторонней печатью. Качественная фотопечать часто является одним из дополнительных требований помимо печати типичных офисных документов, а иногда основным требованием [5].

### **1.3. Преобразование данных в системах сканирования и печати**

Обобщённая схема систем сканирования и печати показана на рисунке 1.3. Данная схема отражает высокоуровневую архитектуру большинства МФП. Каждый блок в схеме сам по себе имеет сложную организацию, например драйвер печати операционной системы (ОС) Windows [20]. Тем не менее, обобщённая

схема позволяет описать потоки преобразования данных в рассматриваемых системах.

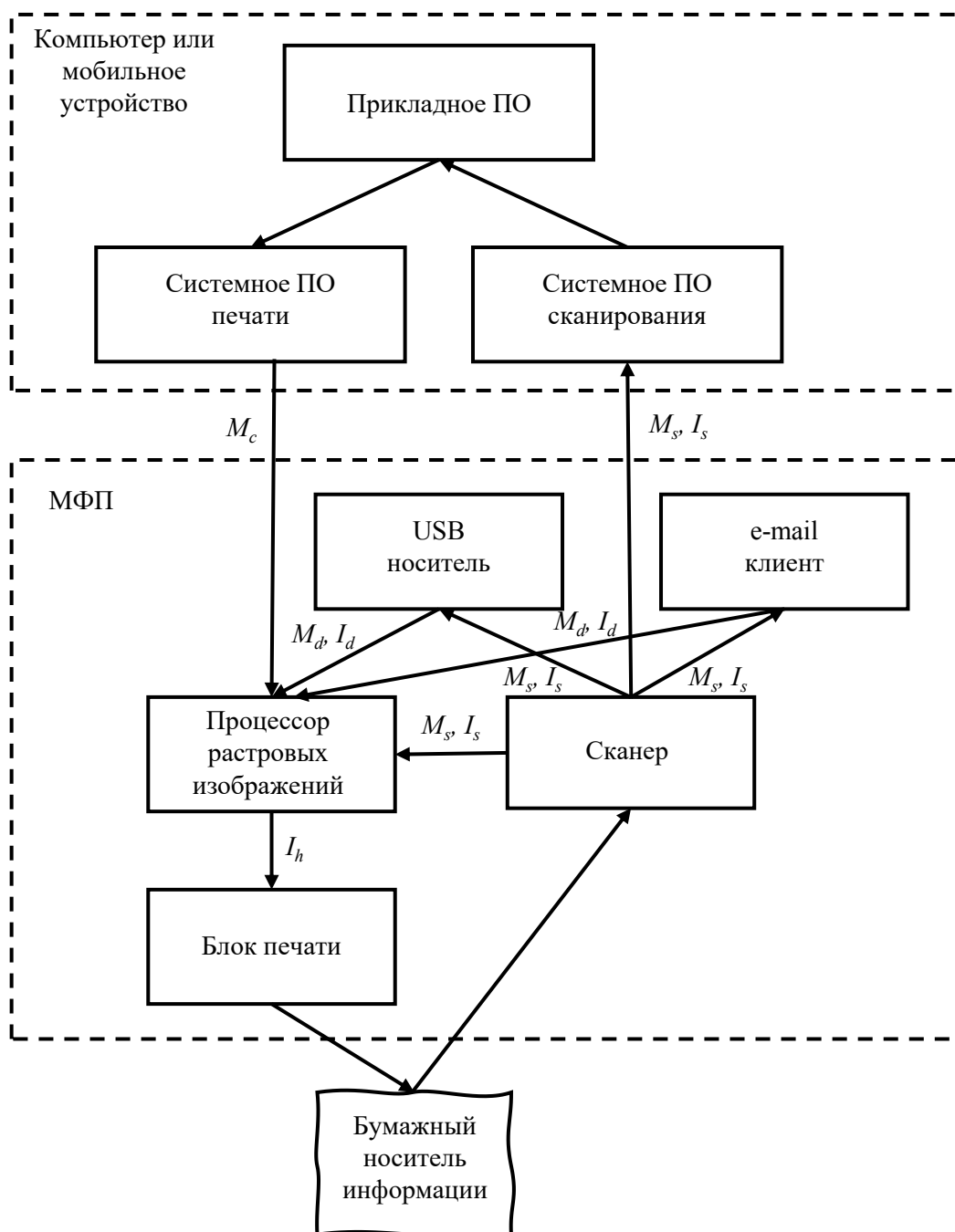


Рисунок 1.3 – Обобщенная схема систем сканирования и печати

Для выполнения печати прикладное программное обеспечение (ПО) персонального компьютера или мобильного устройства обращается через

программный интерфейс к системному ПО печати, которое либо целиком поставляется с ОС (Windows, Linux, MacOS, Android, iOS), либо частично поставляется производителем устройства печати, а частично ОС. К системному относятся ПО очереди печати и драйверы. Одной из задач драйверов является преобразование печатаемой информации в метафайл  $M_c$ , который передается в устройство печати и может быть интерпретирован им. Метафайл  $M_c$  может быть открытого формата, например PDF [10] или PS (Postscript) [21], или проприетарного формата производителя принтеров, например HPGL (Hewlett-Packard graphics language) [22]. Передача метафайла может выполняться по локальному или сетевому интерфейсу, каждый из которых может быть как проводным, так и беспроводным.

Большинство современных МФП позволяет печатать данные с USB носителей, например, USB-флешек, мобильных жёстких дисков, цифровых камер и смартфонов, подключенных в режиме носителя. Как правило, поддерживаются печать из метафайлов  $M_d$  открытых форматов, в первую очередь PDF, и печать растровых изображений  $I_d$  из файлов графических форматов, в первую очередь JPEG [23] (строго говоря, JPEG – это способ сжатия, а формат хранения сжатых с помощью него изображений называется JFIF, но следуя сложившимся обычаям, формат также будем называть JPEG). Многие МФП имеют встроенный клиент электронной почты, который может получать сообщения, где вложением является метафайл  $M_d$  или изображение  $I_d$ , предназначенные для печати. В режиме копирования сканер передает для печати сканированное изображение  $I_s$ , которое может быть вставлено в метафайл  $M_s$ .

Все предназначенные для печати метафайлы и отдельные растровые изображения поступают в процессор растровых изображений, который интерпретирует последовательность поступающих данных, выполняет их отрисовку в координатах печатаемого изображения  $I_h$  с преобразованием в печатный растр, если того требует используемая технология печати. Процессор



растровых изображений реализуется в виде встроенного ПО (firmware), которое часть обработки выполняет на центральном процессоре, а часть на цифровых процессорах обработки сигналов (digital signal processor, DSP) и специализированных микросхемах (application specific integrated circuit, ASIC). Изображение  $I_h$  поступает в блок печати, который непосредственно переносит изображение на бумагу.

При сканировании из пользовательского интерфейса МФП сканированное изображение в формате графического файла  $I_s$  или преобразованным в метафайл  $M_s$  может быть сохранено на USB-носителе, или отправлено по электронной почте, или записано в папку на ПК. При сканировании из прикладной программы ПК происходит обращение к системному ПО, например TWAIN драйверу [24], поставляемому производителем устройства. TWAIN драйвер обычно имеет интерфейс пользователя для задания параметров сканирования и выполнения операций по коррекции и конвертированию изображений.

Объём памяти для хранения несжатого сканированного изображения может превышать 100 Мб, поэтому при сохранении изображение сжимают. При сохранении в форматы графических файлов, как правило, используется способ сжатия JPEG. При сохранении в метафайлы PDF или XPS применяются несколько подходов, в зависимости от выбранного пользователем назначения документа (для пересылки по e-mail, для просмотра на экране, для последующей качественной печати и т.п.), которые регулируют соотношение между качеством и степенью сжатия. В простейшем случае метафайл содержит только сканированное изображение, сжатое JPEG. Однако предпочтительным является использование концепции модели смешанного растрового содержимого (mixed raster content, MRC), которая подразумевает сегментацию изображения на области фона, текста и картинок, затем задание цвета фона и сохранение областей текста и картинок различными способами [25]. Чтобы различать целое сканированное изображение и

фотографии/рисунки в нём будем называть последние картинками. Схема обработки при использовании MRC показана на рисунке 1.4.

Области картинок, как правило, сжимаются JPEG, а области текста сохраняются в виде сжатого бинарного изображения (применяются различные способы сжатия как с потерей, так и без потерь информации). Дополнительно может выполняться оптическое распознавание символов (OCR) для создания метафайла с возможностью текстового поиска.



Рисунок 1.4 – Схема обработки при использовании концепции модели смешанного растрового содержимого.

Таким образом, печатаемые данные проходят через длинный конвейер преобразований (printing pipeline), и функции по их обработке могут быть встроены в различные блоки, начиная от прикладного ПО, заканчивая процессором растровых изображений. Похожая ситуация имеет место и с конвейером обработки сканированных изображений (scanning pipeline). Печатаемые данные из любых источников финальные преобразования претерпевают в процессоре растровых изображений. Результаты сканирования начинают обработку в ПО МФП. Поэтому

функции, реализованные во встроенном ПО или специализированных микросхемах, имеют наиболее общее применение по сравнению, например, с обработкой в системном ПО какой-то определённой ОС. С другой стороны, требования по производительности и ограничения на потребляемые ресурсы для встроенного ПО МФП существенно более строгие, чем для прикладного ПО ПК.

#### **1.4. Характеристики вычислительных платформ МФП**

Производители устройств печати в спецификациях либо не указывают характеристики встроенных вычислительных платформ (embedded system), например [15], либо ограничиваются только указанием размера оперативной памяти, например [16]. Только в последние годы некоторые производители стали указывать семейство используемого центрального процессора и тактовую частоту, например Cortex-A9 800 МГц в [26]. Объясняется это тем, что обработка данных в МФП выполняется системой на кристалле (system-on-chip, SoC), включающей помимо центрального процессора несколько процессоров цифровой обработки сигналов и интегральные схемы специального назначения [27]. Характеристики центрального процессора слабо влияют на производительность большинства базовых операций обработки изображений MFP, таких как преобразование между цветовыми пространствами, компрессия и декомпрессия JPEG, преобразование растровых изображений в печатный растр, так как данные операции в основном выполняются на отдельных схемах системы на кристалле. Однако при реализации ряда дополнительных функций, например, специальных режимов преобразования сканированных данных в метафайл, вычислительные возможности центрального процессора являются серьёзным ограничением сложности используемых алгоритмов.

Размер доступной оперативной памяти требует значительных усилий от разработчиков встроенного ПО. Для хранения сканируемых и печатаемых

изображений требуется несколько десятков или даже сотен Мб, а размер оперативной памяти устройств печати существенно меньше, чем в ПК. Часто размер оперативной памяти МФП меньше, чем результат сканирования с высоким оптическим разрешением. Например, МФП Samsung CLX-3160 имеет 128 Мб оперативной памяти и позволяет сканировать с оптическим разрешением 1200х600 dpi, глубина цвета сканированного изображения 24 bpp [28]. Легко подсчитать, что сканирование страницы А4 дает изображение размера около 200 Мб. Поэтому при разработке встроенного программного обеспечения стараются минимизировать количество изображений, хранимых в памяти, разрабатывают алгоритмы для обработки изображений по полосам, иногда хранят изображение в сжатом виде.

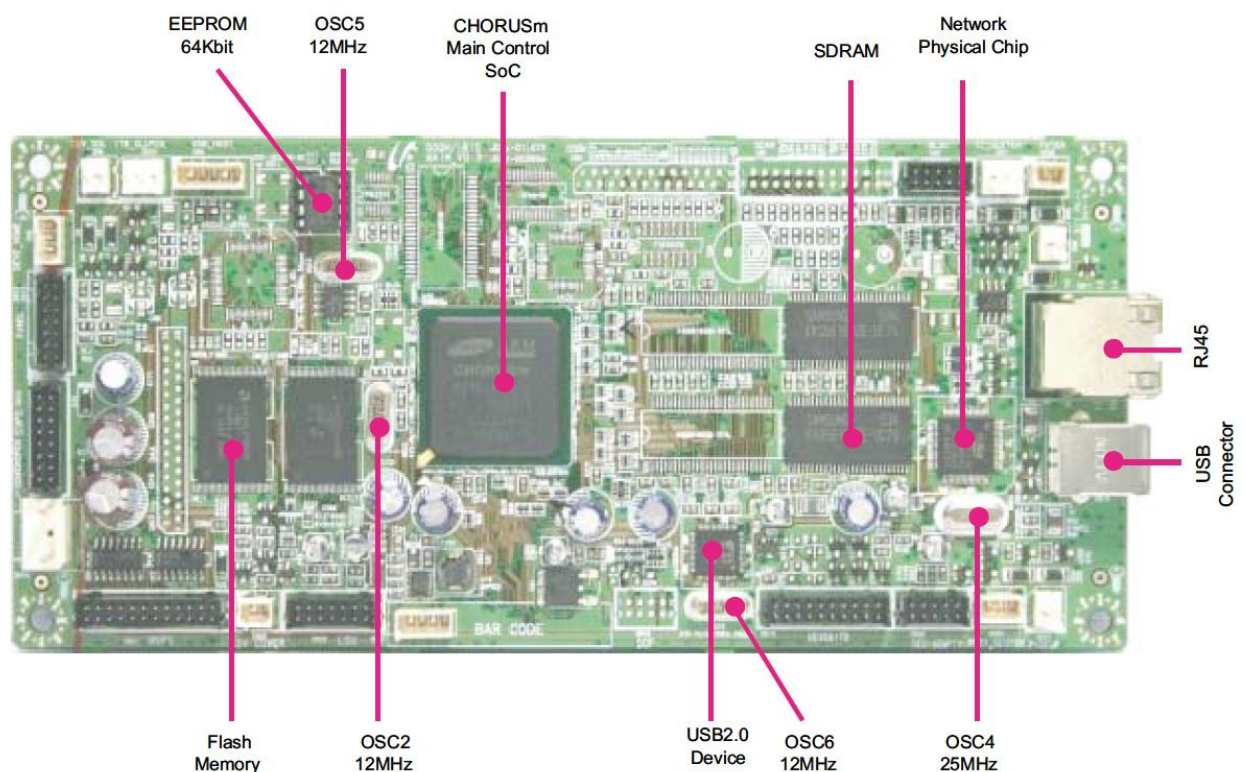


Рисунок 1.5 – Материнская плата МФП Samsung CLX-3160.

Еще одним ограничением при добавлении новой функциональности МФП является размер флеш-памяти, используемой для хранения встроенного ПО. Если свободного пространства для кода, реализующего новые функции, недостаточно,

то эти функции невозможно добавить в существующие устройства путём перепрошивки (обновления) встроенного ПО.

Для демонстрации набора компонент обеспечивающих обработку данных в МФП на рис 1.5 показана материнская плата Samsung CLX-3160, выпуск которого начат в 2006 году [28]. Это старое устройство выбрано для иллюстрации намеренно, так как, с одной стороны, оно имеет все базовые компоненты любого МФП, с другой стороны, его материнская плата не так велика, по сравнению с платами большинства современных МФП, которые содержат множество отдельных схем для поддержки интерфейсов. Заметим, что в устройствах печати помимо материнской платы имеется ещё несколько плат с контроллерами для управления аппаратурой сканирования и печати.

### **1.5. Оценки производительности и качества печати**

Производительность (или скорость) печати и копирования, измеряемая в количестве страниц в минуту, является одной из основных характеристик МФП. Производители устройств печати постоянно соревнуются друг с другом в увеличении производительности и побуждают пользователей покупать более быстрые принтеры. Для оценки скорости печати, копирования и сканирования существуют методики измерения и тестовые документы, разработанные производителями устройств печати, независимыми тестовыми лабораториями и международной организацией по стандартизации (International Standard Organization, ISO, ИСО). Стандарты ИСО, относящиеся к устройствам печати, являются рекомендательными для большинства стран, в том числе Российской Федерации, однако для ряда рынков следование данным стандартам при указании характеристик аппаратуры является обязательным требованием.

Существует несколько стандартов ИСО, регламентирующих оценку производительности МФП в том или ином режиме работы, например ИСО 24734

[29] определяет тестовые файлы, процедуру измерения и измеряемые величины, такие как время выхода первой копии документа (first set out time, FSOT), фактическую пропускную способность (effective throughput, EFTP), теоретическую предельную пропускную способность (estimated saturated throughput, ESAT) для оценки скорости печати. ИСО 29183 [30] содержит аналогичные сведения для оценки скорости копирования для документа, помещённого на стекло планшетного сканера. Характеристики производительности настолько важны для производителей устройств печати, что любые модификации устройств, ведущие к ухудшению данных характеристик, обычно отвергаются. Это требует от разработчиков либо реализовывать очень быстрые способы обработки, либо добавлять в устройства и поставляемое с ними ПО такие новые функции, производительность которых не оценивается с помощью принятых в индустрии тестов.

Время обработки изображений непосредственно влияет на производительность: чем меньше время обработки, тем выше производительность. Везде в тексте диссертации, где упоминается время или скорость печати, имеется в виду величина FSOT, а для сканирования и копирования аналогичный по смыслу параметр. Ряд результатов обработки, не относящихся к производительности, также приведён для тестовых документов из стандарта ИСО 24734, потому что эти тестовые файлы хорошо знакомы специалистам, работающим в индустрии офисной печати. Версия ИСО 24734:2021 [29] определяет три тестовых файла (pdf, doc и xls), каждый из которых содержит по 4 страницы. Первые страницы этих документов показаны на рисунке 1.6.

Обеспечение высокого (для данной технологии) качества печати, сканирования и копирования также является одной из задач производителей МФП. Аналогично оценкам производительности, оценки качества делаются на основе методик и тестовых изображений (таблиц) самих производителей, независимых тестовых лабораторий и стандартов ИСО. Как правило, существующие методики в

зависимости от оцениваемой операции предлагают либо напечатать, либо отсканировать, либо скопировать набор тестовых таблиц, и оценить полученный результат визуально и иногда количественно в соответствии со сформулированными критериями. Оцениваются точность передачи оттенков серого и цветов, равномерность толщины линий, разрешение, иногда строится частотно-контрастная характеристика. Для ряда количественных оценок параметров качества, например, определения координат цвета в аппаратно-независимом цветовом пространстве, требуется специальная аппаратура. Монография [31] описывает принципы оценки качества по тестовым таблицам при работе с цветными изображениями. На рисунке 1.7 показана одна из тестовых таблиц, используемых в стандарте ИСО 15775 [32], который определяет методику оценки качества копирования цветными МФП.

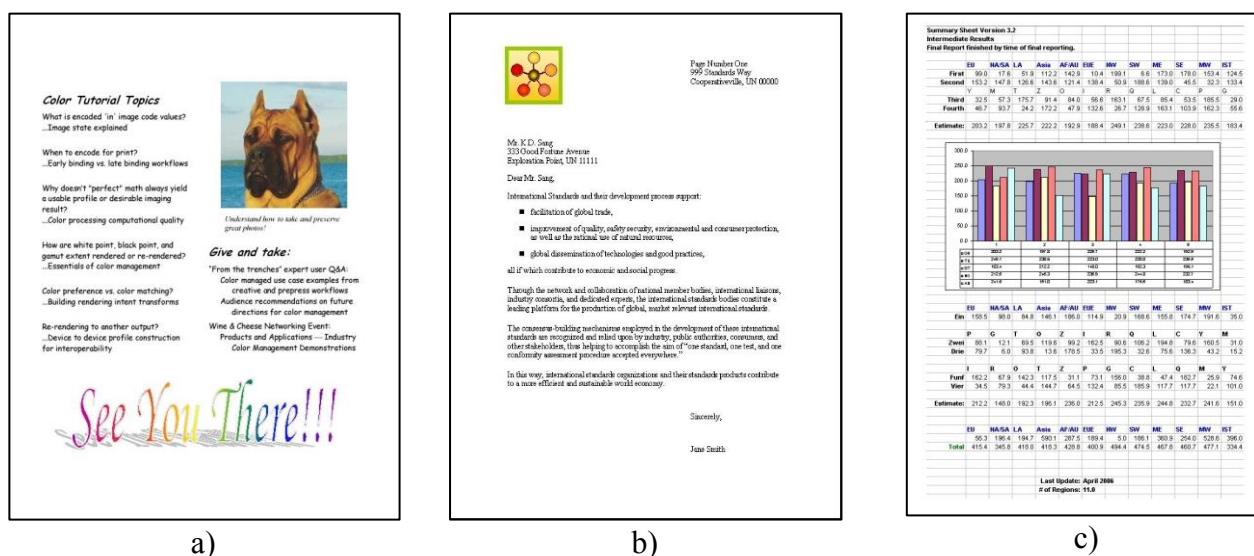


Рисунок 1.6 – Первые страницы тестовых документов стандарта 24734 [29]: a) pdf; b) doc; c) xls.

Методики из стандартов ИСО предлагают только качественные критерии оценки качества иллюстраций и фотографий. Для количественного сравнения между собой качества работы различных методов и/или устройств обычно проводят опрос (интервью) группы из нескольких десятков пользователей в



контролируемых условиях. Работа [33] подробно анализирует подходы, используемые для оценки качества печати. Сравнимые изображения печатают, а затем просматривают и оценивают в одинаковых условиях освещённости. При отборе пользователей с помощью тестов Исихары [34] убеждаются, что они имеют нормальное цветовое зрение. Для получения количественной оценки используются ранжирование изображений с последующим вычислением среднего балла мнений (mean opinion score, MOS) [35] или метод парных сравнений [36].

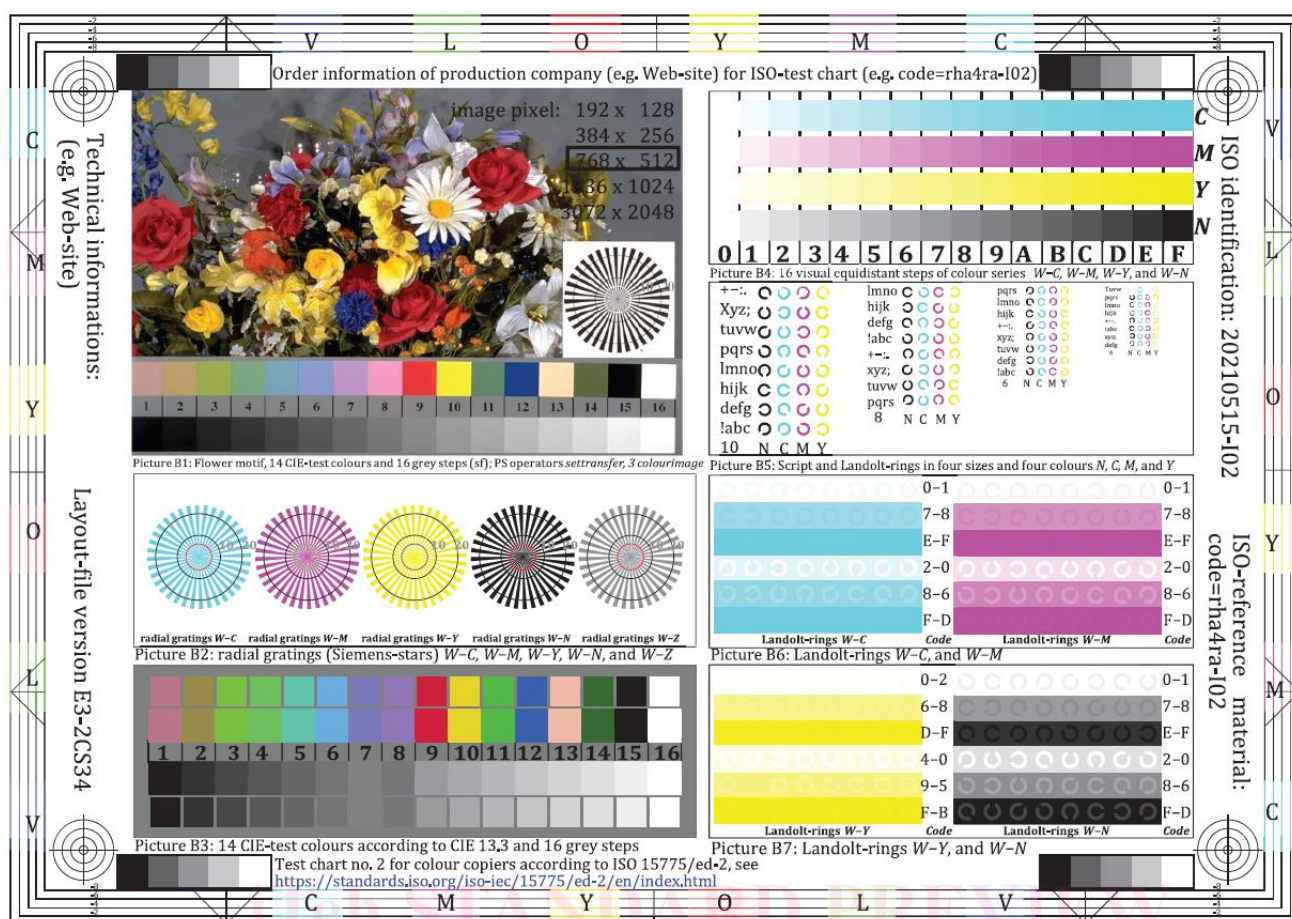


Рисунок 1.7 – Тестовая таблица из стандарта ИСО 15775 [32].

В данной работе оценка качества изображений также осуществлялась по тестовым таблицам и путём опроса (интервью) группы пользователей для напечатанных изображений. Группа пользователей формировалась из студентов



третьего и четвёртого курса Национального исследовательского ядерного университета «МИФИ».

### **1.6. Анализ требуемых новых режимов систем сканирования и печати**

В большинстве моделей МФП реализован только базовый функционал печати, сканирования и копирования, работающий единообразно вне зависимости от вида оригинала и назначения результата. В промышленности, государственных и образовательных учреждениях часто возникает необходимость модификации МФП путём разработки специальных режимов печати, сканирования и копирования, работа которых имеет определённую специфику в зависимости от типа оригинала, требуемых результатов по качеству и размеру получающихся файлов. Зачастую для подготовки документа для печати или обработки результата сканирования пользователь вынужден использовать дополнительное ПО, вручную редактировать изображение, что требует затрат времени и средств. В качестве примера можно привести сканирование многостраничного документа и сохранение результата в файл, поддерживающий хранение изображений нескольких страниц. Если какие-то страницы перевёрнуты вверх ногами, то пользователь вынужден редактировать файл и вручную переворачивать изображения страниц. Предпочтительнее определять ориентацию страницы в процессе сканирования и переворачивать изображение автоматически.

Можно выделить ряд задач, для которых требуется разработка специальных режимов сканирования и копирования, а для них новых методов обработки изображений:

- при сканировании оригиналов определённого типа возникают специфические искажения, требующие коррекции, например, затемнение и дисторсия символов при сканировании книг;

- выполняется сканирование определённого вида многостраничных документов, где есть специальные правила размещения изображений на страницах в итоговом файле, например, копирование двухстраничных идентификационных карточек на одну страницу;
- одновременно выполняется сканирование нескольких оригиналов, каждое из которых требуется трактовать как отдельное изображение, например, несколько чеков помещаются на стекло планшетного сканера и сканируются, изображение каждого одинаковым образом ориентированного чека при сохранении в файл помещается на отдельную страницу или, наоборот, компактно размещаются на страницах;
- последовательно сканируются несколько частей одного оригинала, требуется объединить сканированные части в единое изображение, например, постер размером А3 сканируется по частям на сканере размера А4, части объединяются в единое изображение;
- заданы специальные требования к качеству изображения и ограничения на размер файла для хранения результата сканирования;
- выполняется копирование в режиме экономии расходных материалов, например, не копируются пустые страницы и снижается расход тонера/краски.
- выполняется копирование на бумагу, размер которой отличается от размера оригинала, например, документ размера Letter копируется на бумагу размера А4.

К специальным режимам печати относятся режимы, предназначенные для экономии расходных материалов (бумаги и/или краски) и печати защитных элементов, например, для добавления цифровых водяных знаков в напечатанный документ и защиты документа от копирования.

В настоящее время пользователи имеют возможность снимать с помощью фотокамер и смартфонов сотни фотографий в день. Иногда возникает желание или необходимость напечатать ряд изображений, не все из которых имеют высокое

качество. Печать некачественных фотографий ведёт к излишнему расходу краски, бумаги и ресурса печатающего устройства. Визуальная оценка качества изображений пользователем и отбор качественных фотографий занимает дополнительное время и не всегда может быть сделана адекватно при просмотре на экране смартфона или компьютера. Качество большого числа изображений можно улучшить. Безусловно существует огромное количество программ для коррекции фотографий. Однако их применение зачастую требует от пользователя специальных знаний, занимает дополнительное время. Разработка полностью автоматических методов оценки и улучшения качества печатаемых фотографий позволяет исключить печать некачественных фотографий, значительно ускоряет подготовку изображений для печати, повышает качество напечатанных фотокарточек, в целом, упрощает для пользователя процедуру подготовки фотографий к печати.

Срок эксплуатации МФП превышает 10–15 лет. Требуется добавление новых режимов/функций в существующие системы сканирования и печати без изменения аппаратной части, а за счёт модификации системного и/или встроенного программного обеспечения. В сочетании с требованиями к качеству и производительности операций печати, сканирования и копирования, и с учётом ограничений аппаратных платформ МФП, разработка вычислительно эффективных методов, имеющих компактную реализацию, является не простой и актуальной задачей.

### **1.7. Формализация научных задач диссертационного исследования**

В результате анализа объекта и предмета исследования можно ввести следующую математическую формализацию общей научной задачи диссертационного исследования. Общая научная задача – развитие научно-методического аппарата  $\{M\}$  разработки и реализации методов обработки

изображений для создания в системах сканирования и печати  $S$  комплекса новых и модернизированных режимов работы  $F$ , удовлетворяющих требованиям к производительности  $P$  и качеству  $Q$ , в условиях ограничений ресурсов  $R$ :

$$\{M\}: (S, R, F, P, Q) \rightarrow (F_a - F_b) > n, \quad (1.7)$$

где  $F_a$  – множество режимов  $S$ , включая новые разработанные,  $F_b$  – исходное множество режимов  $S$ ,  $F_b \subset F_a$ ;  $n > 5$ ;

при ограничениях:

- а.  $\sum_{i=1}^{|F_a|} r_j(f_i) < r_j$ ,  $r_j \in R$ , где  $r_j(f_i)$  – потребление  $j$ -го ресурса режимом  $f_i \in F_a$ ,  $i = 1..|F_a|$ ;
- б.  $q_i \in Q$ ,  $Q \subset Q_d$ , где  $q_i$  – критерий качества работы  $f_i$ ,  $Q_d$  – множество допустимых значений критериев качества;
- с.  $p_i \geq p_{di}$ ,  $p_i \in P$ ,  $p_{di} \in P_d$ , где  $p_i$  – производительность  $f_i$ ,  $p_{di}$  – минимально допустимая производительность для  $f_i$ .

Заметим, что данная формулировка требует не ухудшить качество и производительность режимов  $F_b$ , которые существовали до внедрения комплекса новых режимов, а также подразумевает совместное использование ресурсов системы режимами.

Элементами НМА  $\{M\}$  являются методология научно-исследовательской работы при создании новых решений для систем сканирования и печати, методика реализации множества режимов сканирования, подходы к реализации специальных режимов печати и автоматического улучшения качества печатаемых фотографий, принципы автоматического выбора и размещения изображений на странице. Элементы  $\{M\}$  формулируются в ходе решения частных задач диссертационного исследования и подробно рассматриваются в последующих главах.

### **Выводы по первой главе**

Введены основные определения и обозначения, используемые в данной работе. Выполнен краткий обзор систем печати и сканирования, потоков данных, параметров встроенных вычислительных платформ, характеристик производительности и качества. Проведён анализ требуемых новых режимов систем сканирования и печати. Формализованы научные задачи диссертационного исследования. В главе сделаны следующие выводы:

- требования по производительности достаточно высоки, а встроенные вычислительные платформы устройств печати имеют скромные характеристики для типичных объемов обрабатываемых данных, что требует разработки быстрых методов обработки, эффективно использующих оперативную память;
- преобразование данных при печати и сканировании состоит из множества этапов, во многих случаях существует несколько альтернативных этапов в конвейере обработки, куда может быть встроена новая функции обработки изображений;
- в индустрии устройств печати качество оценивают по тестовым таблицам как путём измерения количественных показателей, так и на основе визуальной оценки; сравнительную оценку качества осуществляют по результатам опроса группы пользователей, используя ранжирование или парное сравнение напечатанных изображений;
- актуальны разработка для МФП новых специальных режимов сканирования и печати, методов автоматической оценки и улучшения качества печатаемых фотографий;
- часто требуется добавление новой функциональности в существующие системы сканирования и печати без модификации аппаратной части, а за счёт обновления системного и встроенного ПО;

- общей научной задачей диссертационного исследования является развитие научно-методического аппарата разработки и реализации методов обработки изображений для создания в системах сканирования и печати комплекса новых и модернизированных режимов работы, удовлетворяющих требованиям к производительности и качеству, в условиях ограничений ресурсов системы;
- элементы НМА, такие как методология научно-исследовательской работы при создании новых решений для систем сканирования и печати, методика реализации множества режимов сканирования, подходы к реализации специальных режимов печати и автоматического улучшения качества печатаемых фотографий, принципы автоматического выбора и размещения изображений на странице, формулируются в ходе решения частных задач диссертационного исследования

## **Глава 2. Методология научно-исследовательской работы при создании решений для систем сканирования и печати**

В компаниях по производству продуктов электроники процесс разработки новых функций для устройств и ПО построен, как правило, следующим образом: сначала исследовательским подразделением выполняется научно-исследовательская работа (НИР), затем результаты НИР всесторонне оцениваются и при условии положительных оценок передаются в производственные подразделения для реализации в продукте. Задачи НИР должны быть решены за ограниченное время. Для эффективного и успешного решения задач требуется формализация структуры и организации данного рода деятельности. Согласно [37] «методология — это учение о структуре, логической организации, методах и средствах деятельности». Таким образом, методология НИР при создании решений для систем сканирования и печати призвана упростить и ускорить разработку функционала МФП, обеспечить соответствие результата требованиям к производительности и качеству.

### **2.1. Существующие методологии исследований и разработки**

Дадим краткий обзор методологий, которые позволяют организовать и формализовать деятельность по решению задач исследований и разработки. Начнём с перечисления этапов научного исследования, затем рассмотрим этапы научно-исследовательской (НИР) и опытно-конструкторской работы (ОКР) в промышленности, завершим методологиями разработки ПО.

Выделяют следующие этапы научного исследования [38]:

1. Выбор темы и обоснование ее актуальности;
2. Ознакомление с проблемой посредством обзора существующей информации по данной теме;
3. Уточнение темы и формулировка плана научно-исследовательской работы;

4. Собственно исследование, содержание которого зависит от используемых методов (экспериментальные, теоретические и т.д.);
5. Обработка и анализ результатов исследования, формулировка выводов;
6. Оформление документации с результатами исследования.

ГОСТ 15.101-2021 [39] определяет общие этапы НИР при разработке продукции:

1. Выбор направления исследований на основе анализа современного состояния проблемы, в том числе результатов поиска объектов интеллектуальной собственности по данной теме;
2. Теоретические и экспериментальные исследования для решения поставленных задач;
3. Обобщение и оценка результатов исследований, выпуск научно-технической документации;
4. Приёмка НИР в соответствии с критериями, определёнными в техническом задании (ТЗ).

ГОСТ 15.301-2016 [40] определяет общие этапы ОКР:

1. Разработка ТЗ на ОКР;
2. Собственно проведение ОКР, включая разработку технологической документации и создание опытного образца;
3. Испытание опытного образца на соответствие заявленным в ТЗ характеристикам.

При составлении договоров на выполнение НИР и ОКР содержание работ на каждом из перечисленных этапов конкретизируется и уточняется в зависимости от отрасли промышленности и решаемой проблемы.

Для разработки ПО предложено множество методологий [41, 42]. Классическая каскадная методология (называемая также «водопад», waterfall) состоит из следующих этапов: анализ проекта, анализ требований, проектирование продукта, разработка продукта, системное тестирование, поддержка. Каждый



последующий этап стартует только по окончании предыдущего. Предполагается, что все требования к итоговым продуктам известны заранее, неопределённость отсутствует, никаких новых запросов на изменения в ходе проекта не произойдёт. Данные предположения являются идеализированными, поэтому в настоящее время классическая каскадная методология применяется редко, зато возникло множество вариантов её развития. Одним из таких вариантов является итеративная каскадная методология, в которой возможен возврат на предыдущие этапы. Ещё один вариант – это инкрементная каскадная методология, где требования по функциональности ПО увеличиваются от версии к версии, каждая версия разрабатывается, например, по итеративной каскадной методологии. В рамках инкрементной методологии рассматривают также разбиение ПО на модули и отдельную разработку каждого модуля с последующей их интеграцией в единый комплекс. В V-методологии поэтапная разработка с чётко определенной последовательностью этапов совмещается с тестированием. Каскадные методологии разработки призваны контролировать сроки и бюджет выполнения проектов по разработке ПО, но их успешное применение возможно при условии высокой определённости с требованиями к ПО на старте проекта, что нехарактерно для исследовательских задач.

Спиральная методология основана на анализе рисков [43]. Риски – это любые неопределённости и события, которые могут помешать достичь целей проекта. В данной методологии циклически повторяются следующие основные этапы: определение задач, альтернатив и ограничений; оценка альтернатив, выделение рисков и способов борьбы с ними; разработка и верификация очередной версии ПО; планирование следующей итерации. На рисунке 2.1 показана схема спиральной методологии (заимствовано из [41]). Основные этапы разбиваются на детальные шаги. Как правило, чем больше витков, тем больше сроки и бюджет проекта. Спиральная методология позволяет подстраиваться под изменения

требований к ПО. На каждом витке спирали создается некий прототип продукта. Успех проекта во многом определяется качеством анализа и управления рисками.

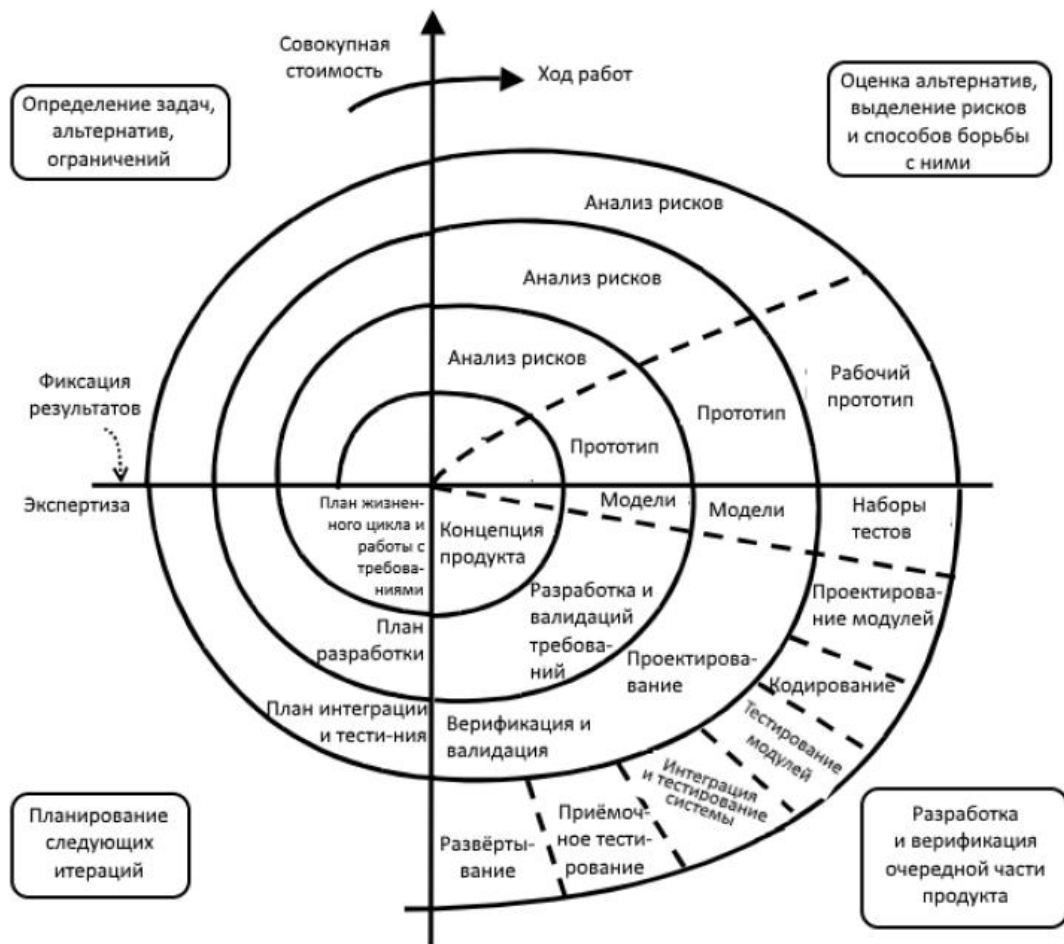


Рисунок 2.1 – Схема спиральной методологии разработки ПО.

В настоящее время массово используются разновидности гибкой (agile) методологии разработки ПО, которые предусматривают постоянное привлечение заказчика или пользователей для оценки текущих результатов, формулировки задач и расстановки приоритетов. В рамках гибкой методологии члены проектной команды регулярно общаются друг с другом, ежедневно обсуждают статус текущих задач. Планирование осуществляется на короткие промежутки времени, так называемые спринты, типичная длительность которых две недели или месяц. По завершению спринта планируется деятельность для следующего спринта. При

планировании стремятся в конце каждого спринта иметь работоспособный прототип или версию ПО.

На практике, как правило, методология разработки ПО используются не в её «классическом» варианте, а в модифицированном виде с учётом особенностей решаемых задач и структуры организации, часто применяются комбинации элементов нескольких методологий.

## **2.2. Методология НИР при создании решений для МФП**

Предлагаемая методология является результатом индуктивного обобщения опыта автора, полученного при работе в исследовательских подразделениях компаний по производству продуктов электроники. Тема для НИР может быть сформулирована как в виде достаточно широкого направления, в рамках которого необходимо найти более узкую задачу, например, создание нового режима защиты документов от копирования, так и более конкретно, например, разработка метода коррекции эффекта «красных глаз» на фотографиях.

Критерии успешности выполнения проектов НИР, как правило, определены достаточно размыто. Будем считать НИР успешным, если его результаты были реализованы в каком-либо МФП или поставляемом с ним ПО. Такой критерий является довольно амбициозным, так как зачастую исполнители НИР не осведомлены обо всем многообразии факторов, которые влияют на принятие решения о реализации разработанных методов в продуктах. Основываясь на данном критерии, нельзя сделать вывод об успешности проекта сразу по его окончании, необходимо время, чтобы понять, нашли ли результаты НИР практическое применение или нет. Надо отметить, что только часть результатов НИР реализуется в продуктах.

### **Риски НИР при разработке для систем сканирования и печати**

Ретроспективный анализ около 40 проектов НИР выполненных за 3 года автором и его коллегами позволил выявить факторы, которые воспрепятствовали реализации в продуктах их результатов. На стадии выполнения проекта эти факторы можно считать рисками, которые ещё не реализовались. Соответственно, минимизация рисков повышает вероятность успешности НИР [44]. Таким образом, целесообразно сформулировать методологию НИР при разработке методов обработки изображений для систем сканирования и печати на основе анализа и управления рисками.

Риски, выявленные в результате ретроспективного анализа проектов НИР, можно разделить на четыре категории:

- связанные с вопросами интеллектуальной собственности;
- связанные с качеством работы;
- связанные с производительностью;
- связанные с реализацией на целевой вычислительной платформе.

Рассмотрим эти риски подробнее. Выпуск продукта, который в какой-то своей части нарушает чей-то патент на изобретение, может повлечь серьёзные финансовые потери компании. Обнаружение факта подобного нарушения может сделать реализацию результата НИР в продукте невозможным или отложит её до решения юридических вопросов с правообладателем или перехода патента в общественное достояние. Также необходимо рассмотреть вопрос защиты интеллектуальной собственности результатов НИР, чтобы, как минимум, усложнить реализацию похожего функционала конкурентам. Если это не сделано на стадии НИР, то способ защиты (подача патента на изобретение, раскрытие информации в виде публикации, объявление информации секретом производства) должен быть выбран до реализации в продукте. Использование открытого или проприетарного ПО с нарушением их лицензионного соглашения может привести

к финансовым и репутационным потерям или потребовать раскрытия исходного кода продукта.

По оценкам фокус-групп пользователей полученный в результате НИР способ обработки может давать неудовлетворительное качество или не заинтересовать пользователей. Очевидно, что в этих случаях принимается решение не реализовывать полученные результаты в продукте. В ходе тестов качества печати или копирования, принятых в индустрии печатных устройств, может быть выяснено, что разработанные прототипы обеспечивают качество хуже, чем у конкурентов. В такой ситуации решение о реализации в продукте принимается в зависимости от текущих требований рынка. По результатам стандартных оценок производительности, таких как описанные в первой главе, характеристики МФП могут заметно ухудшиться. Это также, как правило, служит причиной как минимум отложить реализацию до момента, когда более мощное аппаратное обеспечение позволит выполнять обработку быстрее.

Существует риск невозможности реализовать результат НИР на целевой платформе по причине высоких требований к ресурсам встроенной вычислительной платформы или из-за того, что выпуск данного типа продуктов был прекращён по экономическим соображениям. Сложности с переносом программного прототипа на целевую платформу, например из-за отсутствия тестов и документации, могут привести к высоким затратам или длительным срокам реализации в продукте.

Таблица 2.1 содержит риски НИР при создании решений для МФП, ранжированные с помощью метода анализа иерархий (МАИ, analytic hierarchy process, АНР) [45], для каждого риска определён вес, характеризующий его относительную важность. По весу можно выделить 3 группы рисков:  $\rho 1$ - $\rho 2$  ведут к отказу от реализации результатов НИР;  $\rho 3$ - $\rho 5$  ведут либо к отказу от реализации, либо принятие решения будет отложено на несколько лет;  $\rho 6$ - $\rho 10$  вероятно повлекут задержку с реализацией на несколько месяцев.

Таблица 2.1. Риски НИР при создании решений для МФП, ранжированные с помощью МАИ

Обозначение	Риск	Вес
р1	Функция непривлекательна для пользователей	0,190
р2	Нарушается чужая интеллектуальная собственность	0,182
р3	Невозможно реализовать на целевой платформе	0,141
р4	Неудовлетворительное качество по оценкам пользователей	0,139
р5	Ухудшаются показатели производительности	0,138
р6	Используется ПО с неподходящей лицензией	0,059
р7	Высокие затраты на реализацию в продукте	0,045
р8	Качество хуже, чем у конкурентов	0,045
р9	Длительные сроки реализации в продукте	0,039
р10	Вопрос защиты интеллектуальной собственности не решен	0,020

### Последовательность этапов НИР

Предлагается основанная на анализе и управлении рисками методология НИР при создании решений для МФП [46]. Базовая идея похожа на спиральную методологию разработки ПО, но содержание этапов отличается. Схема методологии приведена на рисунке 2.2. Обычно срок выполнения НИР для разработки некоторого метода обработки изображений для МФП составляет от полугода до года.

Методология НИР подразумевает, что последовательность этапов делает три витка, переходя от определения задач, ограничений и альтернатив к анализу рисков, а затем к разработке. Однако, если при оценке риска выясняется, что он достаточно высок, то возможно полное либо частичное повторение предыдущего витка (при условии, что это не противоречит срокам выполнения НИР).

Проект начинается с поиска идеи для новой функции для систем сканирования и печати. На данном этапе изучается продукция конкурентов;

анализируются задачи, возникающие у различных групп пользователей; рассматриваются возможные альтернативы (часто с помощью, так называемого, мозгового штурма). Далее создаётся чрезвычайно упрощённый макет продукта, чтобы подтвердить или опровергнуть гипотезу, что пользователи будут использовать продукт в данном виде. Для подобного рода макетов Алберто Савойя ввел термин «претотип» (pretotype) [47]. В области проектирования визуальных интерфейсов пользователей аналогом «претотипа» является макет с низкой детализацией (low-fidelity mockup), который позволяет оценить, насколько интерфейс будет понятен и удобен [48]. Однако концепция «претотипа» шире, так как позволяет оценить не только интерфейс пользователя, но и иные характеристики продукта, например, качество работы ПО оптические распознавания символов для текста, напечатанного в экономичном режиме. Поскольку термин «претотип» не получил широкого распространения, будем использовать термин макет. При разработке методов обработки изображений подобный макет это, как правило, вручную нарисованное или обработанное изображение.

На следующем этапе для макета выполняется предварительная оценка пользовательских свойств. Данный этап предназначен для оценки и минимизации рисков  $\rho_1$  и  $\rho_4$ . Если риски велики, то предпочтительнее вернуться на первый этап. Если же макет пользователям понравился, то выполняется переход к поиску и анализу патентной информации по данной тематике. Для всестороннего глубокого поиска патентной информации нужно обладать рядом навыков и иметь доступ к специальным поисковым средствам. Однако, даже поверхностный анализ с помощью публично доступных поисковых систем позволяет минимизировать риск  $\rho_2$ .

Заключительным этапом первого витка спирали является выбор критериев оценки качества разрабатываемого метода, подготовка тестовых данных и формулировка задачи исследования. Задачей исследования является разработка

метода обработки, удовлетворяющего сформулированным критериям. Критерии должны быть количественными, для качественных критериев необходимо предложить способ преобразования их в количественные. Предпочтительно использовать критерии, известные в индустрии сканирования и печати или похожие на известные, что упрощает последующее внедрение результата НИР в продукт.



Рисунок 2.2 – Схема методологии НИР при создании решений для МФП.

Второй виток спирали начинается с обзора литературы и решений конкурентов по выбранной теме. Данный этап характерен для любого научного исследования. Его целью является поиск подходов для решения задачи НИР. Далее



подход, который рассматривается как наиболее перспективный применяется для разработки алгоритма и первого прототипа ПО. Как правило, первый прототип программы разрабатывается с помощью высокоуровневых средств, например MATLAB [49] или пакетов Python для обработки изображений, сигналов, машинного обучения и тому подобное. Использование высокоуровневых средств позволяет относительно быстро получить прототип для оценки качества работы разработанного алгоритма. Однако использовать высокоуровневые средства следует с учётом последующей реализации на целевой платформе, так как с их помощью легко получить алгоритм с неоправданно высокими требованиями к вычислительным ресурсам. С другой стороны, внесение на данном этапе излишне жёстких ограничений на сложность алгоритма может привести к невозможности достичь требуемых критериев качества.

На следующем этапе путём тестирования первого прототипа выполняется оценка качества работы разработанного метода. Заметим, что время обработки на данном этапе не оценивается. Если показатели качества оказываются неудовлетворительными, то осуществляется возврат на предыдущие этапы, то есть выполняется разработка другого алгоритма и первого прототипа ПО. Таким образом, этот этап предназначен для минимизации рисков  $\rho_4$  и  $\rho_8$ .

Если же качество работы разработанного метода удовлетворительное, то выполняется идентификация этапов в конвейере печати или сканирования, подходящих для реализации данного метода. Как было отмечено в первой главе, как правило, существует несколько альтернативных этапов в конвейере, куда можно встроить разрабатываемый метод обработки изображений. Например, метод экономичной печати можно реализовать во встроенном ПО МФП, а можно в драйвере принтера. По сути, данный поиск альтернатив предназначен для поиска запасных вариантов реализации метода в том случае, если его реализация на целевой платформе окажется невозможна, или его производительность окажется низкой, или стоимость или время реализации окажутся высокими. Данная

деятельность направлена на управление рисками  $\rho_3$ ,  $\rho_5$ ,  $\rho_7$  и  $\rho_9$ . Она начинается третий виток методологии.

Одним из результатов этапа идентификации возможных мест для реализации метода являются требования ко второму прототипу ПО, который реализуется на C/C++ с учётом возможностей и ограничений идентифицированных вычислительных платформ. Далее для минимизации риска  $\rho_6$  выполняется анализ лицензий используемых библиотек ПО (если таковые имеются). Далее выполняется оценка качества и времени работы для второго прототипа, что направлено на управление рисками  $\rho_3$ ,  $\rho_4$ ,  $\rho_5$  и  $\rho_8$ . Если сформулированные на первом витке критерии удовлетворены, то для управления риском  $\rho_{10}$  осуществляется переход к анализу подходов к защите разработанного метода как объекта интеллектуальной собственности. Завершают третий виток анализ требований производственных подразделений к документированию и оформлению кода (которые, как правило, существенно более строгие чем в исследовательских отделах) и подготовка документации и тестов в соответствии с этими требованиями. Данные этапы предназначены для минимизации рисков  $\rho_7$  и  $\rho_9$ .

Примеры использования методологии НИР приведены в разделах 3.7.1 и 3.7.2. Следует отметить, что успешное применение данной методологии требует от сотрудников исследовательских подразделений расширять свои компетенции в смежные области, связанные: с разработкой ПО и аппаратного обеспечения; анализом патентной информации и защитой объектов интеллектуальной собственности; анализом особенностей лицензионных соглашений ПО.

Выполнение НИР в соответствии с данной методологией позволило примерно в три раза увеличить количество результатов исследований, реализованных в системном или встроенном ПО МФП.

## **Выводы по второй главе**

Выполнен обзор методологий научного исследования, НИР и ОКР в промышленности, разработки ПО. Для НИР при создании решений для систем сканирования печати в качестве критерия успеха выбран факт реализации результата НИР в продукте. На основе ретроспективного анализа выявлены факторы, которые препятствовали реализации результатов. На стадии НИР данные факторы можно рассматривать как риски. С помощью метода анализа иерархий риски ранжированы по важности.

Предложена спиральная методология НИР при создании решений для МФП, основанная на анализе и управлении рисками. Данная методология является результатом индуктивного обобщения опыта автора и использует ряд элементов существующих методологий исследований. Применение методологии позволило в три раза повысить количество результатов НИР внедряемых в системное или встроенное программное обеспечение систем сканирования и печати.

### Глава 3. Методика режимов сканирования

Многие современные МФП имеют несколько режимов сканирования и копирования, которые различаются по типу сканируемых оригиналов или по способу сохранения информации. Обычно эти режимы реализуют во встроенном ПО МФП. Целесообразно не вести разработку каждого режима с нуля, а обобщить и формализовать приёмы решения подобных задач, т.е. согласно [3] выработать методику – «совокупность методов, приёмов практического выполнения чего-либо». Такая методика рекомендует комплекс методов обработки сканированных изображений и последовательность их применения.

К методике предъявляются следующие требования:

- комплекс методов должен включать коррекцию типичных искажений сканированных изображений, а также артефактов, возникающих при сохранении сканированной информации;
- последовательность обработки должна быть гибко настраиваемой в зависимости от особенностей каждого режима;
- обработка должна быть достаточно быстрой, чтобы удовлетворять требованиям к производительности ряда операций [30];
- методы обработки должны функционировать в условиях ограничений аппаратных платформ МФП.

Одним из основных ограничений является объём используемой оперативной памяти. Существуют МФП, объём оперативной памяти которых позволяет хранить только одно изображение, сканированное с максимальным оптическим разрешением. Это требует разработки алгоритмов, способных вести обработку без выделения дополнительных массивов памяти, сравнимых по размеру с объёмом для хранения сканированного изображения (в англоязычной литературе такие алгоритмы называют in-place или in-situ).

### 3.1. Систематизация искажений изображений при сканировании

Рассмотрим типичные искажения изображений, сканированных с помощью планшетного сканера [50], а также артефакты, возникающие при сохранении изображений в файлы графических и метафайловых форматов. Таблица 3.1. содержит описание наиболее распространённых искажений, оценку частоты их встречаемости, а также приоритет (чем меньше значение, тем выше приоритет) коррекции данного искажения во встроенном ПО МФП. Приоритезация выполнена с помощью МАИ [45]. На рисунке 3.1 показаны примеры искажений. Проведённый анализ показал необходимость в первую очередь повышать чёткость изображения, подавлять печатный растр и корректировать скос.

Таблица 3.1. Искажения изображений при сканировании и сохранении

Искажение	Описание	Частота встречаемости	Приоритет коррекции во встроенном ПО
Скос	Пользователь может поместить оригинал под небольшим углом к краям сканера, в результате сканирования изображение повернуто относительно горизонтальной оси на некоторый угол (см. рисунок 3.7). Такое искажение называют скосом (skew) или наклоном. Скос замечен и негативно оценивается пользователями, когда угол скоса по модулю больше $0,2^\circ$ [51].	Часто	3
Печатный растр	При сканировании линейкой датчиков сканера регулярной решётки растра (см. рисунок 1.1) в результате биения их периодов возможно возникновение эффекта муара (или муарового узора) –	Часто	2

	нежелательной текстуры на изображении [52]. Предпочтительно иметь изображение с большим количеством оттенков вместо изображения, состоящего из точек только нескольких цветов.		
Размытие	При сканировании происходит потеря чёткости изображения из-за размытия резких границ (см. рисунок 3.25). Особенно заметна деградация визуального качества текста при многократном копировании, что влияет на качество OCR.	Всегда	1
Шум	Отношение уровня сигнала к шуму у датчиков сканеров высокое [14], кроме того, часть шума подавляется на аппаратном уровне в датчике, поэтому для офисных МФП подавление шума датчиков не является первоочередной задачей. Иногда шумом сканированных изображений называют дефекты, связанные с нежелательными особенностями оригинала: частичками краски в области фона, грязью на стекле сканера, просвечиванием изображения с обратной стороны бумаги.	Редко	6
Падение контраста	Зачастую ряд областей оригинала имеет слабый контраст, при сканировании контраст также может уменьшиться. Желательно не только не допустить падения контраста, но и улучшить визуальное качество сканированного изображения.	Иногда	5
Искажения в области	При сканировании сшитых оригиналов, таких как книги и журналы, часть страницы в области корешка не прилегает плотно к	Книги сканируют	7

корешка книги	стеклу сканера, что приводит к геометрическим и яркостным искажениям [53].	относительно редко	
Артефакты сжатия	Большая степень сжатия ведёт к появлению в сохранённых изображениях артефактов компрессии или бинаризации текста. Важным является поиск подходов для обеспечения наилучшего качества сохранённого сканированного изображения при фиксированном размере файла [54].	Всегда при сильном сжатии	4

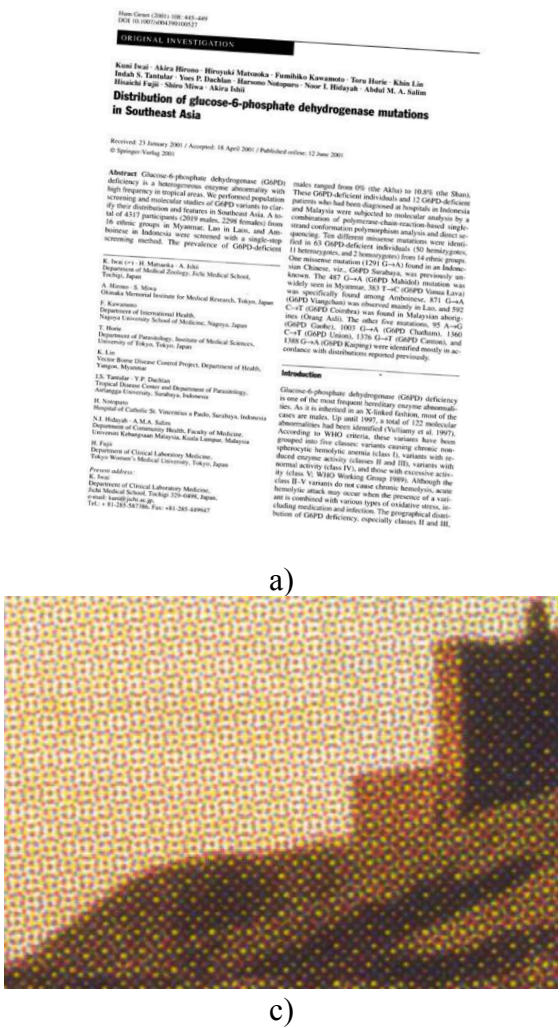


Рисунок 3.1 – Примеры искажений сканированных изображений: а) скос; б) искажения в области корешка книги (правая сторона изображения); с) печатный растр; д) артефакты сжатия.

pression dates back more than 35 years. The initial in this field was on the development of analog meth- ransmission bandwidth, a process called *bandwidth* t of the digital computer and subsequent develop- ated circuits, however, caused interest to shift from sion approaches. With the relatively recent adoption. nal image compression standards, the field has un- th through the practical application of the theoretic 940s, when C. E. Shannon and others first formulat- of information and its representation, transmission,

ression is recognized as an “enabling technology.” In mentioned, image compression is the natural technol- ased spatial resolutions of today’s imaging sensors and ion standards. Furthermore, image compression plays, ortant and diverse applications, including televideo- ing (the use of satellite imagery for weather and other ns), document and medical imaging, facsimile trans-

### 3.2. Совокупность методов и последовательность обработки

Предлагаемая методика включает:

- предварительную оценку параметров обработки по изображению, сканированному с низким разрешением (как правило, 75 dpi), а затем сканирование с заданным пользователем разрешением (например, 300 dpi) и обработка, которая осуществляется без использования дополнительной памяти (in-place);
- устранение основных искажений сканированных изображений: скоса, размытия, печатного раstra, потери контраста;
- проведение сегментации, необходимой для сохранения изображения в метафайл в соответствии с концепцией модели смешанного растрового содержимого, т.е. текст, картинки и фон сохраняются отдельно, что позволяет уменьшить артефакты сжатия.

Изображение с низким разрешением используется также для предварительного просмотра в пользовательском интерфейсе при необходимости обеспечить какие-либо действия пользователя [55]. Сканирование с низким разрешением выполняется в несколько раз быстрее, чем с заданным, тем не менее данная операция может увеличивать общее время обработки, что необходимо учитывать в зависимости от требований к режиму сканирования. Заметим, что общее время обработки может уменьшиться в том случае, когда с заданным разрешением сканируется оригинал существенно меньший размера стекла сканера за счёт сканирования только той области, в которой расположен оригинал. Если аппаратные возможности МФП позволяют, то методика допускает сканирование сразу в заданном разрешении и получения изображения низкого разрешения путём децимации (уменьшения с интерполяцией «ближайшего соседа») сканированного.

Рекомендуемая последовательность этапов обработки для режимов сканирования показана на рисунке 3.2. Методика является гибкой, в зависимости



от задач и ограничений конкретного режима сканирования допускается исключение отдельных этапов.

Целями обработки изображения низкого разрешения являются: определение положения оригинала на стекле сканера; определение угла скоса; сегментация на области (блоки) фона, текста и картинок (или фона и области переднего плана); оценка параметров в соответствии с конкретным режимом сканирования (если требуется).

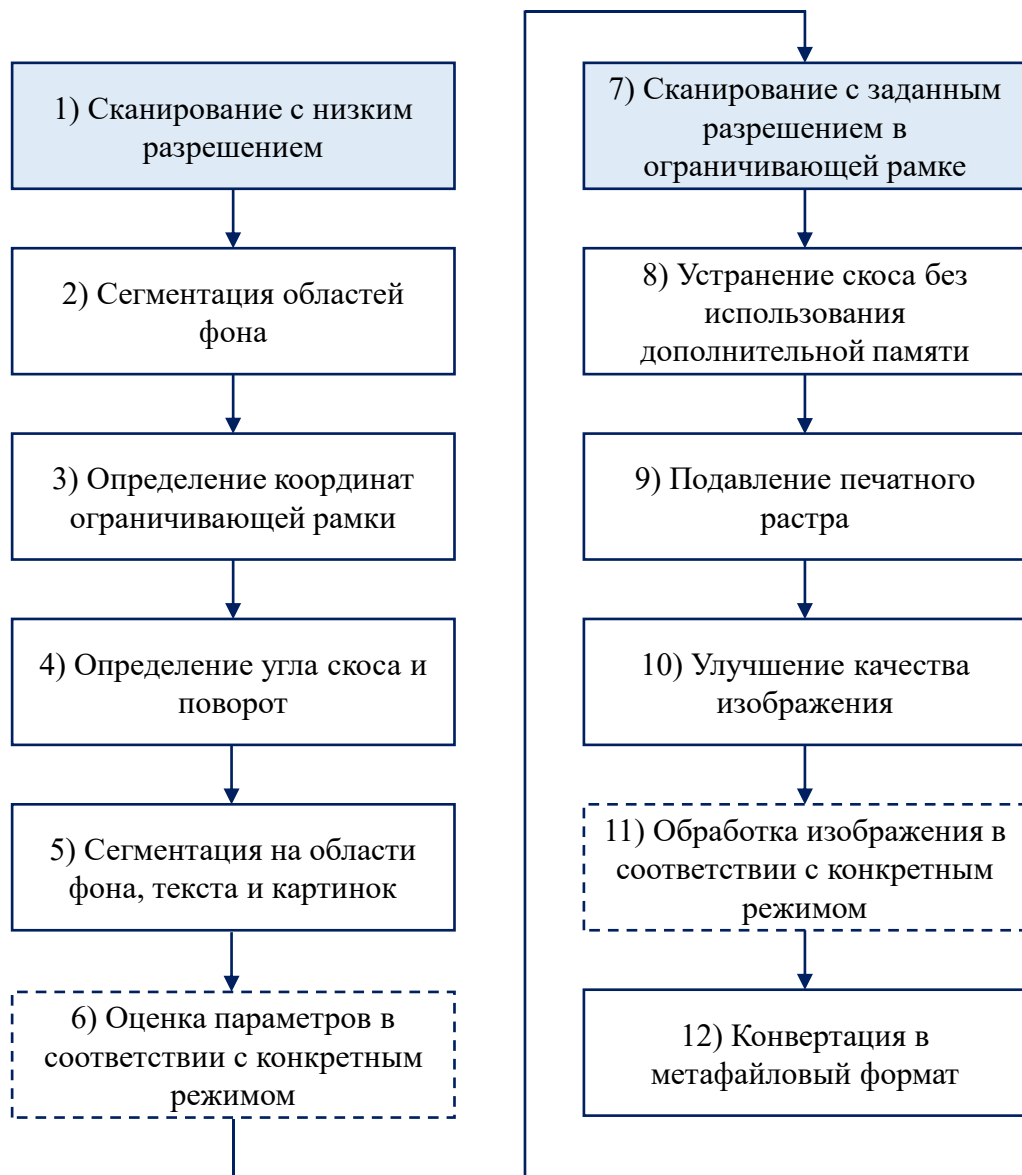


Рисунок 3.2 – Последовательность этапов обработки для режимов сканирования.

После получения изображения низкого разрешения (этап 1) выполняется сегментация блоков на фоновые и остальные (этап 2). Алгоритм сегментации описан в разделе 3.4. Для блоков, не относящихся к фону, вычисляются координаты ограничивающей рамки (этап 3). Определяется угол скоса  $\phi$  (этап 4). Если  $|\phi| > 0,2^\circ$ , то считается, что скос обнаружен, и выполняется поворот изображения низкого разрешения на  $-\phi$ . Далее выполняется сегментация на области фона, текста и картинок (этап 5), причём, если поворота не было, то переиспользуются полученные ранее результаты сегментации на фон и остальные области. В соответствии с конкретным режимом сканирования могут выполняться оценки параметров на изображении низкого разрешения (этап 6).

После сканирования изображения заданного разрешения (этап 7), если скос обнаружен, то выполняется поворот без использования дополнительного буфера памяти (этап 8). Затем подавляется печатный растр (этап 9). При этом повышается резкость изображения. Далее могут выполняться операции по повышению качества изображения (этап 10), в частности, повышается глобальный и локальный контраст. Следующим этапом (11) является обработка в соответствии с конкретным режимом сканирования. На заключительном этапе (12) результат обработки записывается в метафайл, причём способ записи зависит от режима сканирования и, как правило, выполняется в соответствии с концепцией модели смешанного растрового содержимого.

Предложенная методика позволяет использовать различные методы для каждого из этапов обработки, например, разные подходы для сегментации изображений, но важно, чтобы они удовлетворяли ограничениям вычислительной платформы и критериям качества. Для разработки режимов сканирования и копирования в соответствии с методикой были предложены следующие новые способы:

- оценки и коррекции скоса [56] (см. раздел 3.3);
- сегментации на области фона, текста и картинок [57–60] (см. раздел 3.4);

- подавления печатного раstra и устранения размытия [61–63] (см. раздел 3.5);
- улучшения глобального и локального контраста [64, 65] (см. раздел 3.6).

Методика была применена при разработке следующих режимов работы МФП:

- преобразования сканированного изображения в метафайл PDF с векторизацией контуров символов [66–68] (см. раздел 3.7.1);
- копирования двухсторонних карточек [69–72] (см. раздел 3.7.2);
- сканирования оригиналов маленького размера [73–75];
- сканирования книг [53, 76, 77].

### **3.3. Коррекция скоса**

#### **Существующие подходы к оценке угла скоса**

Алгоритмы определения угла скоса для его последующей коррекции широко обсуждаются в литературе. В обзоре [51] предложено разбиение способов определения угла скоса на четыре группы:

- анализ проекций;
- применение преобразований Хафа или Радона [78];
- поиск характерных точек, например, угловых, и анализ их положения на изображении;
- анализ ориентации связных областей.

Абсолютное большинство существующих алгоритмов ограничены определением ориентацией изображений документов, содержащих преимущественно текст. Например, способ [79], основанный на модели английского языка, не работает на изображениях текста ряда других языков, равно как неприменим для различного рода картинок. Алгоритм [80] вычисляет ориентацию прямых, проходящих через центры масс связных компонент бинарного изображения, что позволяет определять угол скоса только для символов

текста некоторого диапазона размеров. Способ [81] предлагает описывать блоки текста и картинки прямоугольниками и определяет скос на основе статистической оценки множества ориентаций прямоугольников. Сходный в идее подход, но применяемый к каждой связной области, описан в [82]. Перечисленные способы предназначены для обработки изображений сканированных с разрешением 150 dpi и выше, они требуют разметки связных областей и вычисления их характеристик, что ведёт к использованию дополнительной памяти и заметным вычислительным затратам.

В [83] рассматривается способ, основанный на разбиении изображения на квадратные блоки и выполнения преобразования Радона для каждого блока. Способ [84] детектирует линии с помощью модифицированного преобразования Хафа, затем определяет группы параллельных линий. Алгоритмы, основанные на преобразованиях Хафа и Радона для поиска прямых линий, способны определять скос для изображений низкого разрешения. Однако качество их работы сильно зависит от содержимого изображения. На рисунке 3.3 зелёным цветом показаны отрезки прямых, детектированные с помощью преобразования Хафа на изображении низкого разрешения. Если для изображения с рисунка 3.3а угол скоса может быть определен на основе ориентации большинства отрезков, то на изображении с рисунка 3.3b отрезки в области двумерного штрихкода не параллельны краям карточки.



Рисунок 3.3 – Примеры детектирования отрезков прямых с помощью преобразования Хафа; а) лицевая сторона карточки; б) обратная сторона карточки.

### Способ определения угла скоса

Предлагается способ определения угла скоса, основанный на определении ориентации ближайших по столбцу к верхнему и нижнему краю ограничивающей рамки видимых элементов изображения [55]. Способ устойчиво работает как для изображений со строками текста на любом языке и любого размера шрифта, так и для картинок. Последовательность шагов способа показана на рисунке 3.4.



Рисунок 3.4 – Схема способа оценки угла скоса.

На шаге 1 изображение  $I$  в пределах ограничивающей рамки с координатами левого верхнего угла  $(r_{\min}, c_{\min})$  и правого нижнего угла  $(r_{\max}, c_{\max})$  конвертируется в полутоновое  $I_I$  по формуле (1.1). На шаге 2 для  $I_I$  вычисляется порог  $T$  и выполняется бинаризация:

$$I_1(r, c) = \begin{cases} 0, & I_I(r, c) \leq T, \\ 1, & I_I(r, c) > T. \end{cases} \quad (3.1)$$

Для выбора порога может использоваться алгоритм Оцу [85]. На шаге 3, как показано синими стрелочками на рисунке 3.5, для каждого столбца в пределах ограничивающей рамки вычисляется массив расстояний  $L_u$  от верхнего края рамки до ближайшего чёрного пикселя (равного 0):

$$L_u(c) = \min((r_{\max} - r_{\min}), (\min_{r \in [r_{\min}, r_{\max}]} (r | I_1(r, c) = 0) - r_{\min})), \quad (3.2)$$

$$\forall c \in [c_{\min}, c_{\max}].$$

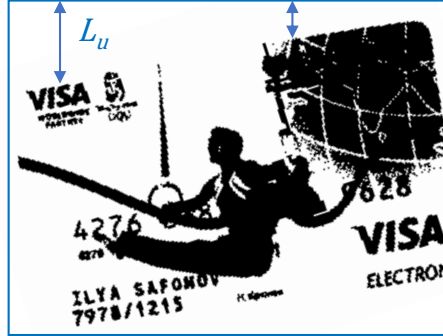


Рисунок 3.5 – Бинаризованное изображение в пределах ограничивающей рамки.

Шаге 4 — это фильтрация массива  $L_u$  минимальным фильтром с апертурой  $2s + 1$ :

$$L'_u(c) = \min_{c \in [c-s, c+s]} (L_u(c)), \quad \forall c \in [c_{\min} + s, c_{\max} - s]. \quad (3.3)$$

На шаге 5 сначала вычисляются конечные разности  $dL_u$  между элементами  $L'_u$  с шагом  $2s + 1$ :

$$dL_u(c) = L'_u(c + s) - L'_u(c - s), \quad \forall c \in [c_{\min} + 2s, c_{\max} - 2s]. \quad (3.4)$$

Далее вычисляется множество пар  $(v_1, v_2)$  индексов массива  $L'_u$ , соответствующих началу и концу отрезков линий, близких к прямым, причём: отрезок должен быть достаточно длинным, т.е. иметь разность  $v_2 - v_1$  больше либо равную порогу  $T_2$ ; конечная разность на всём его протяжении должна быть около 0, т.е. меньше либо равной порогу  $T_1$ ; диапазоны отрезков не пересекаются:

$$\begin{aligned} \{(v_1, v_2)_i\} &= \{|dL_u(c)| \leq T_1, \forall c \in [v_1, v_2], (v_2 - v_1) \geq T_2\}, \\ v_1(i) &> v_2(i-1), v_2(i) < v_1(i+1), i \in [1, N_u], \end{aligned} \quad (3.5)$$

где  $N_u$  – это количество пар в данном множестве.

Вычислим для каждого отрезка длины прилежащих катетов  $\Delta x$  и противолежащих катетов  $\Delta y$ :

$$\Delta x(i) = v_2(i) - v_1(i), \quad (3.6)$$

$$\Delta y(i) = L'_u(v_2(i)) - L'_u(v_1(i)). \quad (3.7)$$

Аналогичным (3.2) образом вычисляется массив расстояний  $L_b$  от нижнего края рамки до ближайшего пикселя нулевой яркости, и для него выполняются операции по формулам (3.3) – (3.7). В результате обработки расстояний от верхнего и нижнего края рамки получаются множества длин прилежащих и противолежащих катетов:  $\{\Delta x(k)\}, \{\Delta y(k)\}, k \in [1, N], N$  – количество элементов в множестве. Вычислим для каждого отрезка угол ориентации:

$$\alpha(k) = \arctg(\Delta y(k)/\Delta x(k)). \quad (3.8)$$

В множестве  $\{\alpha(k)\}$ , как правило, присутствует разброс значений углов. Требуется определить преимущественную ориентацию отрезков. Это осуществляется с помощью итерационного алгоритма (шаг 6). Для начала рассмотрим частные случаи: если  $N = 0$ , то ориентацию определить невозможно; если  $N = 1$ , то угол скоса  $\phi = \alpha(1)$ . Далее выполняются итерационный алгоритм, где вначале каждой итерации оценивается угол скоса  $\phi$  как взвешенное среднее и среднеквадратичное отклонение  $\delta$ :

$$\phi = \frac{\sum_{k=1}^N \alpha(k) \Delta x(k)}{\sum_{k=1}^N \Delta x(k)}, \quad (3.9)$$

$$\delta = \sqrt{\frac{\sum_{k=1}^N (\phi - \alpha(k))^2}{N - 1}}. \quad (3.10)$$

Если  $\alpha(k) \notin [\phi - \delta, \phi + \delta]$ , то  $\alpha(k)$  и  $\Delta x(k)$  исключаются из соответствующих множеств, а  $N$  уменьшается на 1. Итерации завершаются либо когда  $\{\alpha(k)\} = \emptyset$ , что означает угол не определён, либо когда  $\alpha(k) \in [\phi - \delta, \phi + \delta], \forall k$ , тогда  $\phi$  есть оценка угла скоса.

### Поворот изображения

Если  $|\phi| > 0,2^\circ$ , то требуется поворот изображения на угол  $-\phi$ . Вычисление координат  $(x, y)$  в системе координат исходного изображения из значений координат  $(x', y')$  в системе координат повернутого изображения обычно выполняется с помощью следующего произведения вектора строки координат на обратную матрицу поворота:

$$\begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} x' & y' \end{bmatrix} \begin{bmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{bmatrix}. \quad (3.11)$$

Для определения значений пикселя в  $(x', y')$  для каждого цветового канала по ближайшим к  $(x, y)$  пикселям выполняется интерполяция. Например, при интерполяции ближайшего соседа берутся значения единственного ближайшего пикселя, при билинейной интерполяции берутся значения четырёх соседних пикселей, а при бикубической – шестнадцати [8]. Реализация поворота по формуле (3.11) требует дополнительного буфера памяти для хранения по меньшей мере части повернутого изображения, причём в худшем случае, когда  $\phi$  около  $45^\circ$  размер буфера достигает размера половины изображения, что при сканировании с высоким разрешением может превышать объём доступной оперативной памяти МФП. Известно несколько способов представления матрицы поворота из (3.11) в виде произведения двух или трёх матриц [86]. Следующее представление в виде



произведения двух матриц позволяет производить поворот без использования дополнительного буфера памяти непосредственно в памяти, используемой для хранения исходного изображения:

$$\begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} x' & y' \end{bmatrix} \begin{bmatrix} \cos \phi & 0 \\ \sin \phi & 1 \end{bmatrix} \begin{bmatrix} 1 & -\operatorname{tg} \phi \\ 0 & \sec \phi \end{bmatrix}. \quad (3.12)$$

Произведение вектора–строки с координатами на первую матрицу дает сдвиг пикселей изображения по  $x$ , при этом координата  $y$  не изменяется. Затем выполняется произведение на вторую матрицу, что даст сдвиг пикселей изображения по  $y$ , координата  $x$  не изменяется. Для получения значений пикселей в сдвигах используется одномерная линейная или кубическая интерполяция. Рисунок 3.6 демонстрирует пример поворота изображения с помощью сдвига по строкам, затем по столбцам.



Рисунок 3.6 – Пример поворота изображения с помощью сдвига по строкам, затем по столбцам.

Повёрнутое таким образом изображение получается чуть более размытым по сравнению с реализацией поворота по формуле (3.11) и использования двумерных билинейной или бикубической интерполяций соответственно. Однако размытие, возникающее в результате интерполяции, корректируется на последующих этапах обработки заодно с размытием, возникающем при сканировании.

### Результаты

На рисунке 3.7 показан пример коррекции скоса для сканированного изображения статьи с двумя колонками текста.

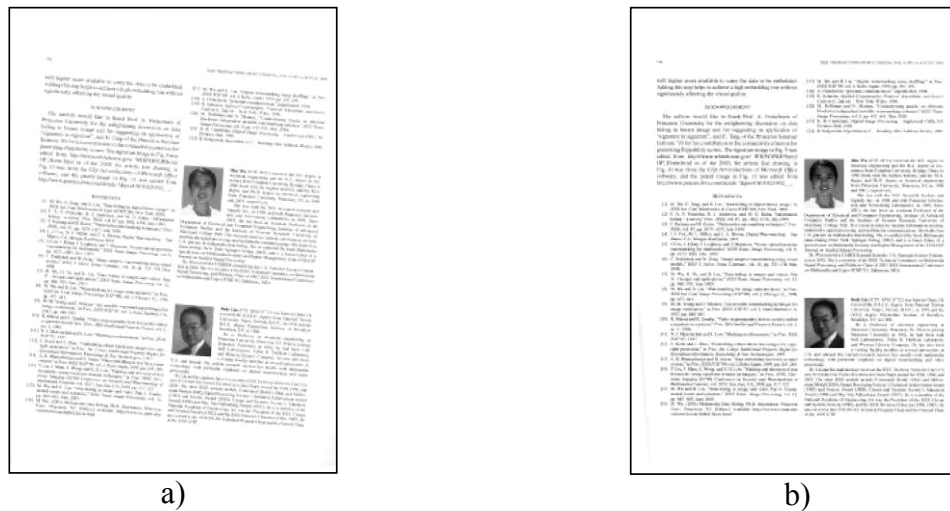


Рисунок 3.7 – Пример коррекции скоса: а) исходное изображение; б) результат коррекции.

Для оценки качества предложенного способа коррекции скоса и сравнения его с программными и аппаратными продуктами был сформирован набор из 10 разнообразных оригиналов, включающий в себя: две фотокарточки, две банковские карты, двое водительских прав, две страницы с текстом и две напечатанные на принтере картинки. Для сравнения работы сканеров были изготовлены специальные шаблоны, чтобы одинаковым образом ориентировать оригиналы на стекле. Абсолютные значения исходных углов скоса были в диапазоне от  $5^\circ$  до  $15^\circ$ . В таблице 3.2 приведены следующие результаты оценки качества коррекции скоса: процент скорректированных изображений от общего числа оригиналов и  $|\overline{\phi}|$  – среднее значение модуля угла скоса, оставшегося после поворота. Предлагаемый способ по качеству работы соответствует лучшим из рассмотренных, при этом является вычислительно простым и быстрым, позволяет выполнять коррекцию скоса без использования большого объема дополнительной памяти. Теоретическим ограничением предложенного способа является то, что угол скоса определяется на основе ориентации элементов, расположенных только сверху и снизу оригинала. На практике их ориентация в подавляющем большинстве случаев совпадает с ориентацией оригинала.

Таблица 3.2. Сравнение результатов коррекции скоса различными продуктами

Продукт	% скорректированных изображений	$ \phi $ после коррекции
HP G4010	100	0,2
Epson V300	40	3,5
Canon 8800	90	1,3
ReadIRIS	100	0,2
FineReader	60	2,2
Предлагаемый	100	0,2

### 3.4. Сегментация на области текста, картинок и фона

#### Подходы к сегментации

Сегментация изображения сканированного документа на области текста, картинок и фона является частным случаем более общей задачи анализа схемы документа (document layout analysis, DLA), которая в различных аспектах рассматривается в большом количестве публикаций [87–89]. В способах анализа схемы документа используется либо восходящий (bottom-up) [90], либо нисходящий (top-down) [91], подход. В первом случае анализ начинается с объектов нижнего уровня, таких как отдельные пиксели и прямоугольные блоки, затем соседние объекты объединяются в более крупные области. Во втором случае анализируется целое изображение с целью разделить его на области заданных классов. Большинство описанных подходов к сегментации документов основано на машинном обучении. В ряде работ отмечается, что области текста, особенно на изображениях, сканированных с низким разрешением, и растрованные картинки можно считать текстурами, поэтому при построении классификаторов с помощью классических алгоритмов машинного обучения целесообразно использовать текстурные признаки [92], а для решения задачи с помощью глубокого обучения применять сети, спроектированные для классификации или сегментации текстур [93].

## **Особенности сегментации изображений документов для сохранения в метафайл**

Существуют следующие особенности задачи сегментации для сохранения сканированного изображения в соответствии с концепцией модели MRC:

- не требуется сегментировать картинки и блоки текста точно по краю областей, допускается обозначить часть примыкающего фона, как относящуюся к ним;
- цена ошибки классификации для разных классов различается, ложно положительные области фона ведут к потере части визуальной информации об изображении, с точки зрения последующего распознавания символов предпочтительнее классифицировать текст на картинках как области текста, чем как картинки;
- ошибочная классификация области текста как картинки и наоборот не ведут к полной потере части визуальной информации об изображении, но могут приводить к искажениям при сохранении в метафайл.

### **Последовательность этапов способа сегментации**

Исходя из обзора существующих алгоритмов, особенностей решаемой задачи и соображений, что способ сегментации должен быть вычислительно простым для реализации во встроенном ПО МФП, а также адаптируемым для различных режимов сканирования, предлагается использовать восходящий подход, где на первом этапе по текстурным признакам выполняется классификация частично перекрывающихся блоков (квадратных участков) изображения, на втором этапе неперекрывающиеся части блоков объединяются в более крупные области [57–59]. Под адаптацией имеется в виду, в частности, то, что в соответствии с предлагаемой методикой в ряде режимов сканирования требуется классификация только на области фона и переднего плана; в зависимости от режима сохранения в метафайл области текста и картинок должны быть прямоугольными или могут быть произвольной формы и т.п. Помимо применения результатов сегментации при

сохранении документа в метафайл, они могут применяться при анимированном показе уменьшенного изображения в интерфейсе пользователя МФП [94, 95].

### Набор данных

Для выбора и обучения классификаторов, применяемых на первом этапе, а также настройки параметров алгоритмов второго этапа собран и вручную размечен набор данных из 300 сканированных изображений страниц размера A4 или Letter. В набор входят изображения типичных офисных документов на различных языках, включая изображения стандартных тестовых наборов для оценки производительности МФП [29], страницы журналов и фрагменты газетных статей. Пример размеченных данных показан на рисунке 3.8. Можно увидеть, что даже если брать блоки только из размеченных областей некоторого класса, то ряд блоков сложно однозначно отнести к данному классу, например, текст на диаграммах. Поскольку подобные неоднозначности в обучающей выборке могут ухудшить качество классификации, то такие блоки исключаются из выборки.

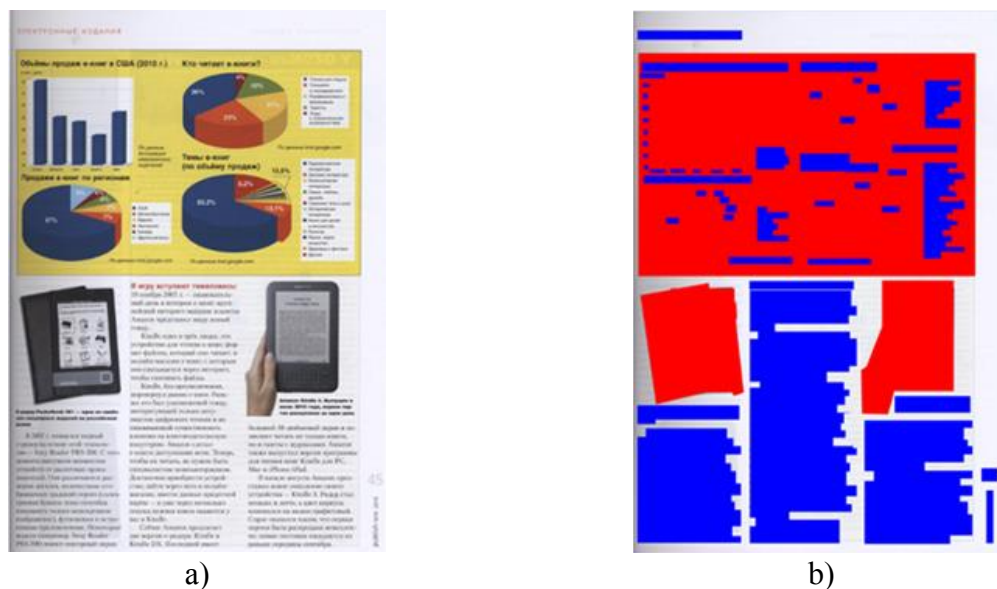


Рисунок 3.8 – Пример данных для обучения: а) сканированное изображение; б) разметка: синие – области текста; красные – картинки; светлые – фон.

### Вычисляемые признаки

При разработке алгоритма изначально для каждого блока рассматривалось вычисление около 30 текстурных признаков [57, 58, 96]. В результате выбора и обучения классификаторов в качестве информативных признаков квадратного блока изображения  $I_\Omega$  размера  $N$  на  $N$  пикселей, где пиксель с координатами  $(r, c)$  расположен в центре блока, было отобрано 12 текстурных признаков  $F_k(r, c), k \in [1, 12]$ . Причём, в соответствии с методологией НИР, описанной во второй главе, для первого прототипа используются признаки в форме, как они описаны в формулах (3.13–3.32) и вычисляемые в действительных числах в формате с плавающей точкой, а для второго прототипа эти признаки преобразуются в целочисленные или в числа с фиксированной точкой [97].

Введём величины, которые используются при вычислении признаков  $F_k(r, c)$ . Среднее значение яркости:

$$\bar{I}_\Omega(r, c) = \frac{1}{N^2} \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}-1} I_\Omega(r + i, c + j). \quad (3.13)$$

Горизонтальная и вертикальная конечные разности:

$$\Delta x(i, j) = \frac{|I_\Omega(i, j + 1) - I_\Omega(i, j)| + |I_\Omega(i, j - 1) - I_\Omega(i, j)|}{2}, \quad (3.14)$$

$$\Delta y(i, j) = \frac{|I_\Omega(i + 1, j) - I_\Omega(i, j)| + |I_\Omega(i - 1, j) - I_\Omega(i, j)|}{2}, \quad (3.15)$$

Максимальное и минимальное значение в  $i$ -й строке блока, максимальное и минимальное значение в  $j$ -м столбце блока:

$$V_{rmax}(i) = \max_{j \in [0, N-1]} I_\Omega(i, j), \quad (3.16)$$

$$V_{rmin}(i) = \min_{j \in [0, N-1]} I_\Omega(i, j), \quad (3.17)$$

$$V_{cmax}(j) = \max_{i \in [0, N-1]} I_\Omega(i, j), \quad (3.18)$$

$$V_{cmin}(j) = \min_{i \in [0, N-1]} I_\Omega(i, j). \quad (3.19)$$

Бинарный блок  $I_{1\Omega}$ , полученный в результате порогового отсечения  $I_\Omega$ :

$$I_{1\Omega}(i, j) = \begin{cases} 0, I_\Omega(i, j) < \frac{2^{D_l}}{2}, \\ 1, I_\Omega(i, j) \geq \frac{2^{D_l}}{2}. \end{cases} \quad (3.20)$$

Признаки  $F_k(r, c)$  вычисляются с помощью формул с (3.21) по (3.32).

Дисперсия:

$$F_1(r, c) = \frac{1}{N^2} \sum_{i=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{j=-\frac{N}{2}}^{\frac{N}{2}-1} (\bar{I}_\Omega(r, c) - I_\Omega(r + i, c + j))^2. \quad (3.21)$$

Среднее модулей разности средней яркости блока и средних яркостей 4-х соседних блоков:

$$F_2(r, c) = \frac{1}{4} (|\bar{I}_\Omega(r, c) - \bar{I}_\Omega(r, c + N)| + |\bar{I}_\Omega(r, c) - \bar{I}_\Omega(r, c - N)| + |\bar{I}_\Omega(r, c) - \bar{I}_\Omega(r + N, c)| + |\bar{I}_\Omega(r, c) - \bar{I}_\Omega(r - N, c)|). \quad (3.22)$$

Среднее из разностей максимальных  $V_{rmax}, V_{cmax}$  и минимальных  $V_{rmin}, V_{cmin}$  значений в строках и столбцах:

$$F_3(r, c) = \frac{1}{2N} \left( \sum_{i=r-\frac{N}{2}}^{r+\frac{N}{2}-1} (V_{rmax}(i) - V_{rmin}(i)) + \sum_{j=c-\frac{N}{2}}^{c+\frac{N}{2}-1} (V_{cmax}(j) - V_{cmin}(j)) \right). \quad (3.23)$$

Среднее сумм горизонтальной и вертикальной конечных разностей:

$$F_4(r, c) = \frac{1}{(N-2)^2} \sum_{i=r-\frac{N}{2}+1}^{r+\frac{N}{2}-2} \sum_{j=c-\frac{N}{2}+1}^{c+\frac{N}{2}-2} (\Delta x(i, j) + \Delta y(i, j)). \quad (3.24)$$

Среднее максимумов горизонтальной и вертикальной конечных разностей, вычисленных в каждом пикселе блока:

$$F_5(r, c) = \frac{1}{(N-2)^2} \sum_{i=r-\frac{N}{2}+1}^{r+\frac{N}{2}-2} \sum_{j=c-\frac{N}{2}+1}^{c+\frac{N}{2}-2} \max(\Delta x(i, j), \Delta y(i, j)). \quad (3.25)$$

Доля пикселей, в которых сумма горизонтальной и вертикальной конечных разностей превышает порог  $T$ :

$$F_6(r, c) = \frac{1}{(N-2)^2} \sum_{i=r-\frac{N}{2}+1}^{r+\frac{N}{2}-2} \sum_{j=c-\frac{N}{2}+1}^{c+\frac{N}{2}-2} 1 | (\Delta x(i, j) + \Delta y(i, j)) > T. \quad (3.26)$$

Доля черных пикселей в бинарном блоке  $I_{1\Omega}$ :

$$F_7(r, c) = \frac{1}{N^2} \sum_{i=r-\frac{N}{2}}^{r+\frac{N}{2}-1} \sum_{j=c-\frac{N}{2}}^{c+\frac{N}{2}-1} (1 - I_{1\Omega}(i, j)). \quad (3.27)$$

Кросс-корреляция бинарного блока  $I_{1\Omega}$  в вертикальном направлении:

$$F_8(r, c) = \frac{1}{N(N-1)} \sum_{i=r-\frac{N}{2}}^{r+\frac{N}{2}-2} \sum_{j=c-\frac{N}{2}}^{c+\frac{N}{2}-1} (I_{1\Omega}(i, j) \oplus I_{1\Omega}(i+1, j)), \quad (3.28)$$

где  $\oplus$  – логическая операция «исключающее или».

Доля пикселей бинарного блока  $I_{1\Omega}$ , изменившихся в результате применения к нему морфологической операции открытия [78], которая обозначена  $I_{1\Omega} \circ \alpha$ , где  $\alpha$  - структурный элемент:

$$F_9(r, c) = \frac{1}{N^2} \sum_{i=r-\frac{N}{2}}^{r+\frac{N}{2}-1} \sum_{j=c-\frac{N}{2}}^{c+\frac{N}{2}-1} 1 | (I_{1\Omega} \circ \alpha)(i, j) \neq I_{1\Omega}(i, j). \quad (3.29)$$

Энергия, однородность и контраст [98] вычисляются из нормализованной матрицы совместной встречаемости  $h_{2\Omega}'(i, j)$  со смещениями  $s_r = 1, s_c = 0$  (см. формулы (1.5) – (1.6)),  $h_{2\Omega}'(i, j)$  вычислена для блока  $I_{\Omega}$ :



$$F_{10}(r, c) = \sum_{i=0}^{2^{D_l}-1} \sum_{j=0}^{2^{D_l}-1} (h_{2\Omega}'(i, j))^2, \quad (3.30)$$

$$F_{11}(r, c) = \sum_{i=0}^{2^{D_l}-1} \sum_{j=0}^{2^{D_l}-1} \frac{h_{2\Omega}'(i, j)}{1 + |i - j|}, \quad (3.31)$$

$$F_{12}(r, c) = \sum_{i=0}^{2^{D_l}-1} \sum_{j=0}^{2^{D_l}-1} (i - j)^2 h_{2\Omega}'(i, j). \quad (3.32)$$

### Алгоритм классификации

При выборе алгоритма классификации для реализации во встроенном ПО важно, чтобы модель имела малый размер и работала быстро. По этой причине рассматриваются логистическая регрессия, дерево решений, метод опорных векторов [99] и адаптивный бустинг (AdaBoost) для деревьев решений [100]. Для выбора алгоритма решается задача классификации блоков текста (размер блока 30 на 30 пикселей). Качество классификации оценивается с помощью среднего гармонического точности (precision) и полноты (recall) – F1-меры [101] в ходе двойного скользящего контроля (cross-validation) по группам однотипных изображений [53]. К группе относятся страницы, набранные одинаковыми шрифтами с одинаковым форматированием и способом растеризации картинок. Если какие-то части таких схожих страниц попадут и в обучающую, и тестовую выборки, то это может привести к завышенной оценке точности. Поэтому вся группа рассматривается как один объект в скользящем контроле по отдельным объектам (leave-one-out) [102]. Для рассматриваемой выборки число групп составило 71. В работе [103] утверждается, что оценки качества работы классификаторов при использовании традиционного скользящего контроля в один цикл получаются завышенными, поэтому целесообразно использовать двойной (вложенный) скользящий контроль [104], который учитывает влияние настройки параметров классификатора.

Процедура двойного скользящего контроля по группам однотипных изображений следующая: вся выборка разбивается на  $U$  групп; параметры классификатора выбираются во внутреннем цикле с помощью скользящего контроля для  $U-1$  групп, т.е. обучение на  $U-2$  группах кроме  $j$ -ой, тестирование на  $j$ -ой, так для всех  $j$ , причем объекты классов каждый раз балансируются как в обучающей, так и в тестовой выборках; далее выполняется обучение (с параметрами классификатора, оптимизированными во внутреннем цикле) на  $U-1$  группах кроме  $i$ -ой, которая не использовалась во внутреннем цикле, тестирование на  $i$ -ой, так для всех  $i$ . Значения критерия качества (F1-мера), полученные при тестировании во внешнем цикле, составляют выборку, для которой вычисляются математическое ожидание и дисперсия.

Таблица 3.3. Математическое ожидание и дисперсия F1-меры, полученные в ходе двойного скользящего контроля по группам однотипных изображений, для различных алгоритмов классификации.

Алгоритм классификации	мат. ожидание F1-меры	дисперсия F1-меры
Логистическая регрессия	0,890	0,0019
Дерево решений	0,896	0,0015
Адаптивный бустинг	0,931	0,0004
Метод опорных векторов	0,929	0,0008

Таблица 3.3 содержит значения математического ожидания и дисперсии F1-меры, полученные в ходе двойного скользящего контроля по группам однотипных изображений, для рассматриваемых алгоритмов классификации. В ходе проведенного сравнения алгоритмов классификации был выбран адаптивный бустинг в модификации Gentle AdaBoost [105] выполненной в ПО, реализованном в лаборатории «Компьютерной графики и мультимедиа» ВМК МГУ [106]. Адаптивный бустинг выполнялся на деревьях малой глубины. Выбор алгоритма обусловлен высокой эффективностью бустинга на реальных задачах [107], низкими

вычислительными затратами, возможностями отбирать информативные признаки и возвращать в качестве результата уровень уверенности. Для сокращения времени выполнения классификации, веса и параметры пороговых классификаторов в узлах деревьев конвертируются из данных в исполняемый код.

В итоге разработан следующий алгоритм сегментации на области фона, текста и картинок. Классификация на первом этапе осуществляется с помощью трёх бинарных классификаторов AdaBoost, построенных по принципу «один против всех» (one-vs-all, one-vs-rest). В качестве критериев для настройки параметров классификаторов используется точность при детектировании фона, полнота при детектировании областей текста и F1-мера при детектировании областей картинок. Обучающие данные балансируются путём прореживания случайным образом преобладающего класса. Выходом каждого из классификаторов является уровень уверенности (confidence level) – число, характеризующее близость к положительному или отрицательному классу. На втором этапе выполняется взвешенное усреднение уровней уверенности по соседним блокам и принимается решение о классе каждого блока. Схема классификации приведена на рисунке 3.9. Далее выполняется объединение блоков картинок и текста в связанные области.

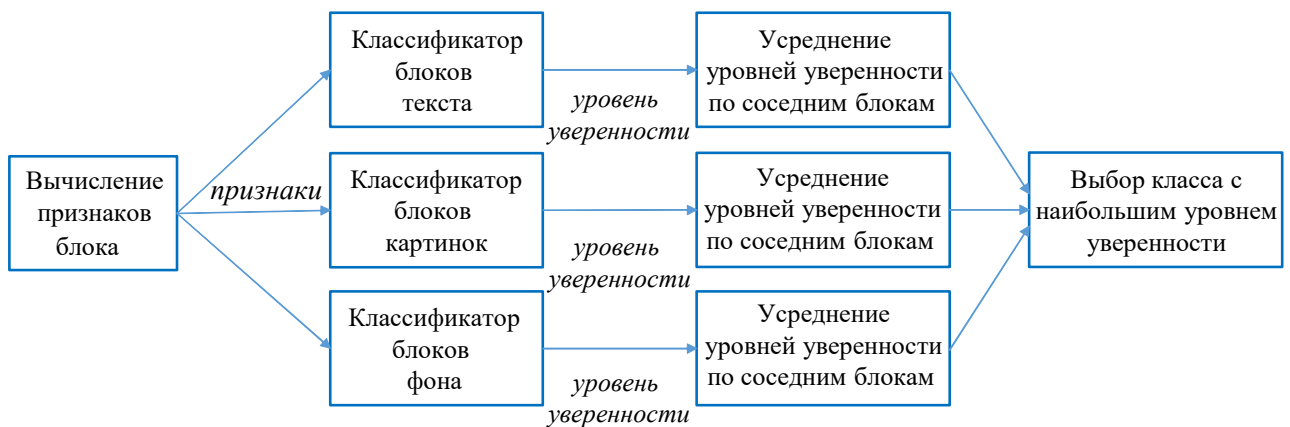


Рисунок 3.9 – Схема классификации блоков изображения на области текста, картинок и фона.

### Критерий качества сегментации

Для выбора размера блоков, размера величины перекрытия блоков, параметров взвешенного усреднения и правил объединения блоков используется критерий качества сегментации областей документа [108], идея которого основана на критерии, использовавшемся на соревновании по сегментации страниц документов в рамках конференции ICDAR [109]. Обозначим  $I_{g_i}$  – изображение, в котором пиксели областей, относящихся к  $i$ -му классу на эталонном изображении равны 1, а остальные равны 0. Связные области на  $I_{g_i}$  образуют множество  $\{g_{ik}\}, k \in [1, M_{g_i}]$ , где  $M_{g_i}$  – количество связных областей. Обозначим  $I_{s_i}$  – изображение, в котором пиксели областей, относящихся к  $i$ -му классу на сегментированном изображении равны 1, а остальные равны 0. Связные области на  $I_{s_i}$  образуют множество  $\{s_{it}\}, t \in [1, M_{s_i}]$ , где  $M_{s_i}$  – количество связных областей. Отношение пересечения к объединению для  $k$ -й связной области  $g_{ik}$  и связных областей из  $I_{s_i}$ , с которой данная область имеет ненулевое пересечение:

$$IOU_{gik} = \frac{|g_{ik} \cap I_{s_i}|}{|\{s_{it} | s_{it} \cap g_{ik} \neq \emptyset\} \cup g_{ik}|}, t \in [1, M_{s_i}]. \quad (3.33)$$

Отношение пересечения к объединению для  $t$ -й связной области  $s_{it}$  и связных областей из  $I_{g_i}$ , с которой данная область имеет ненулевое пересечение:

$$IOU_{sit} = \frac{|s_{it} \cap I_{g_i}|}{|\{g_{ik} | g_{ik} \cap s_{it} \neq \emptyset\} \cup s_{it}|}, k \in [1, M_{g_i}]. \quad (3.34)$$

Рассмотрим три случая соответствия друг другу связных областей эталонного (референсного) и сегментированного изображений:

- Пересечение  $t$ -й связной области сегментированного изображения с эталонным изображением  $I_{g_i}$  образует только одну связную область и  $IOU_{sit} > 0.8$ . Назовем это соответствие «один к одному» и обозначим количество таких областей  $n_{one-to-one}$ .

- Пересечение  $t$ -й связной области сегментированного изображения с эталонным изображением  $I_{g_i}$  образует несколько связных областей и  $IOU_{sit} > 0.8$ . Назовем это соответствие «один ко многим» и обозначим количество таких областей  $n_{one-to-many}$ .
- Пересечение  $k$ -й связной области эталонного изображения с сегментированным изображением  $I_{s_i}$  образует несколько связных областей и  $IOU_{gik} > 0.8$ . Назовем это соответствие «многие к одному» и обозначим количество таких областей  $n_{many-to-one}$ .

Для каждого  $i$ -го класса за исключением класса фона вычисляются смысловые аналоги доли распознанных областей  $R_i$  и точности распознавания областей  $A_i$ , а также их гармоническое среднее  $S_i$ :

$$R_i = \frac{w_1 n_{one-to-one} + w_2 n_{one-to-many} + w_3 n_{many-to-one}}{M_{g_i}}, \quad (3.35)$$

$$A_i = \frac{w_4 n_{one-to-one} + w_5 n_{one-to-many} + w_6 n_{many-to-one}}{M_{s_i}} \quad (3.36)$$

$$S_i = \frac{2R_i A_i}{R_i + A_i}, \quad (3.37)$$

где  $w_j$  – веса, с помощью которых настраивается относительная важность различных типов соответствия областей. Поскольку предпочтительным является соответствие «один к одному»  $w_1 = w_4 = 1,0$ ;  $w_2 = w_3 = w_5 = w_6 = 0,75$ .

Итоговый критерий качества сегментации  $S$  вычисляется, как арифметическое среднее  $S_i$  всех классов кроме фоновых для всех изображений из тестовой выборки.

### Настройка параметров

Поиск параметров для второго этапа сегментации выполнялся в дискретном пространстве набора параметров полным перебором. В таблице 3.4 в качестве примера показаны значения  $S$  для некоторого подмножества из рассмотренных

параметров: размер квадратного блока 10, 25 и 40 пикселей; усреднение уровней уверенности по соседним блокам с помощью равных весов и гауссиана (дисперсия равна 1); величина перекрытия блоков 0% (нет перекрытия, блоки идут встык), 20%, 40%.

Таблица 3.4. Значения критерия качества сегментации  $S$  в зависимости от размера блока, величины перекрытия и способа усреднения уровней уверенности соседних блоков

	Равные веса			гауссиан		
	Перекрытие блоков, %			Перекрытие блоков, %		
Размер блока, пикс.	0	20	40	0	20	40
10	0,40	0,39	0,35	0,40	0,39	0,34
25	0,60	0,68	0,64	0,61	0,69	0,65
40	0,49	0,55	0,54	0,50	0,55	0,54

Малый размер блока ведёт к разбиению больших областей на более мелкие, например, каждая строка текста может сегментироваться как отдельная область. Большой размер блока, наоборот, ведёт к слиянию областей, например колонки текста в журнальной статье объединятся в общую область, хотя при разметке они были указаны как отдельные области. Критерий качества сегментации  $S$  (формула 3.37) штрафует излишнее разбиение и объединение областей одного класса по сравнению с эталонной разметкой. Размер блока 25 пикселей для разрешения 75 dpi соответствует примерно 8 мм.

Процент перекрытия блоков задаёт шаг расположения блоков на изображении. Если перекрытие большое, то шаг уменьшается, и в сочетании с малым размером блока это ведёт к ещё большему разбиению областей на части. Наоборот, если перекрытия нет, то шаг большой, что в сочетании с большим размером блока ведёт к ещё большему объединению областей.

Разница при использовании равных весов или гауссиана для сглаживания уровней уверенности невелика. Использование гауссиана с дисперсией равной 1 даёт немного более высокие значения критерия сегментации. В целом невысокие значения критерия качества сегментации  $S$  объясняются большим разнообразием типов документов в обучающей выборке и неоднозначностью разметки, особенно в части объединения или разделения областей текста, а также обозначения текстовых областей внутри картинок (см. рисунок 3.8).

### 3.5. Подавление печатного растра

Как отмечалось в первой главе, для создания иллюзии присутствия большого количества оттенков цветов при печати с помощью лазерных, светодиодных и струйных принтеров используют печатный растр, состоящий из ячеек полутонирования (см. рисунок 1.1). Аналогичный подход используется в типографской печати. Причём растр применяется не только для печати картинок, но и для текста и элементов графики, цвета которых отличаются от максимально насыщенных красок. При сканировании линейкой датчиков сканера регулярной решётки растра в результате биения их периодов возможно возникновение муарового узора (муара) [52]. При последующей печати сканированного изображения возможно биение периодов двух решёток растра, что также приведёт к муару. Кроме того, после сканирования предпочтительнее иметь изображение с большим количеством оттенков вместо изображения, состоящего из точек только нескольких цветов. Методом решения является преобразование печатного растра (halftone), в непрерывное изменение оттенков (contone). Подобные операции называют подавлением печатного растра (descreen), или обратным полутонированием (inverse halftoning), или подавлением муара (что терминологически не вполне корректно, если речь идёт о предотвращении возникновения муара при последующей печати) [110].

### **Существующие способы подавления печатного растра**

Несколько десятков работ, посвящённых алгоритмам подавления печатного растра, можно разделить на две группы:

- использующие подавление пиков в частотной области;
- применяющие адаптивное сглаживание в пространственной области с последующим повышением резкости.

На амплитуде спектра Фурье изображения чёрно-белой печати можно наблюдать чёткие пики, образованные структурой печатного растра. Подавление таких пиков ведёт к удалению структуры растра с изображения [111, 112]. Однако для цветных изображений подавлять растр с помощью фильтрации в частотной области существенно сложнее по причине использования разных цветовых пространств при печати и сканировании. При печати используется цветовое пространство из четырёх или более цветов, например CMYK. Сканирование производится в трёхкомпонентном цветовом пространстве RGB. Пики на амплитудах спектра для каналов RGB образуются от растров всех цветов, использовавшихся при печати. Эти пики существенно менее заметны по сравнению с пиками для изображений чёрно-белой печати. Зачастую пики растра и периодических структур/текстур содержимого изображения сложно различить, особенно когда параметры печати (линеатура, цветовая система, углы растров) неизвестны. Один из наиболее известных программных продуктов для подавления растра – встраиваемый модуль Descreen [113] для Adobe Photoshop выполняет обработку в частотной области, но для успешной работы требует точного указания параметров печати.

В большинстве публикаций, посвящённых алгоритмам подавления растра, работающим в пространственной области, рассматривается коррекция изображений только чёрно-белой печати [114–117]. Как правило, они включают два этапа: сглаживание с помощью фильтра, предохраняющего границы, а затем повышение резкости. Иногда этим двум этапам предшествует этап определения



параметров печати (в частности, линеатуры) с целью настройки последующих этапов фильтрации. Отдельно стоит отметить способ [118], названный авторами *hardware-friendly descreening* (HDF), который, в отличие от большинства описанных в публикациях алгоритмов, рассматривает подавление раstra не только для картинок, но и для областей текста. HDF основан на идее итерационного фильтра анизотропной диффузии [119], но выполняет только одну итерацию. Он имеет низкую вычислительную сложность, выполняет все вычисления в целочисленной арифметике и предполагает эффективную реализацию в виде интегральной схемы или во встроенном ПО.

С учетом того, что способ подавления раstra должен работать во встроенном ПО МФП в автоматическом режиме без вмешательства пользователя как для изображений чёрно-белой, так и цветной печати, целесообразно реализовать его в виде адаптивного сглаживающего фильтра, степень размытия которого различается для похожих и непохожих соседних блоков изображения. Поскольку в результате применения сглаживающих фильтров подавляется не только печатный растр, но и размываются контурные перепады яркости (границы), а также уменьшается контраст изображения, требуется коррекция размытия и повышение контраста.

### **Способы повышения резкости**

Размытие границ можно устранить с помощью методов восстановления изображений (*image restoration*), к которым относятся различные подходы к инверсной фильтрации и обратной свёртке (*deconvolution*), или с помощью методов улучшения изображений (*image enhancement*), к которым относятся фильтры, повышающие резкость (*sharpening*) [78]. Методы восстановления изображений часто итерационные, имеют высокую вычислительную сложность, не позволяют обрабатывать изображение по частям, поэтому во встроенных системах МФП целесообразно применять фильтры повышения резкости.

Способы повышения резкости разделяются на две группы:

- увеличивающие высоту контурного перепада;
- уменьшающие длину контурного перепада.

Рисунок 3.10 иллюстрирует различие в работе способов данных групп на примере профиля яркости, построенного поперёк размытой границы. У контурного перепада есть высота – разница между постоянными уровнями яркости с двух сторон от перепада (см. рисунок 3.10a), и есть длина (протяжённость) – расстояние между крайними пикселями постоянных уровней яркости с двух сторон от перепада (см. рисунок 3.10b). Добавление к изображению части результата его фильтрации с помощью фильтра верхних частот (ФВЧ) приводит к увеличению высоты перепада, на рисунке 3.10a это показано синим цветом. Например, фильтр нерезкого маскирования (unsharp mask) [78] увеличивает высоту контурного перепада.

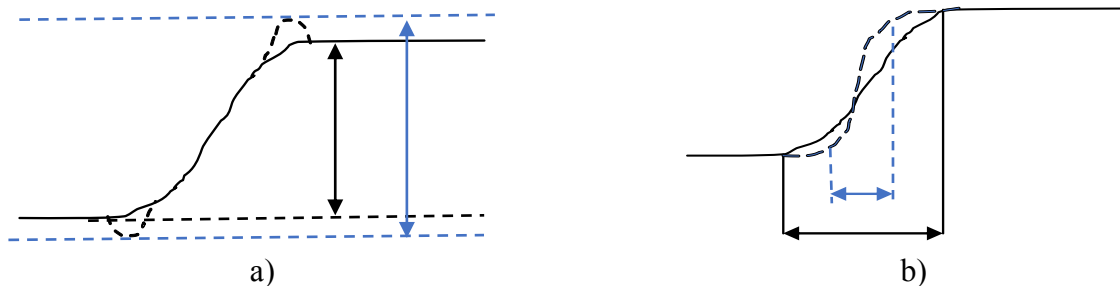


Рисунок 3.10 – Иллюстрация способов повышения резкости: а) увеличение высоты перепада яркости; б) уменьшение длины перепада яркости.

Значительно меньше освещены в литературе фильтры для уменьшения длины перепада яркости. Морфологическое повышение резкости [120] позволяет для одномерного случая преобразовать размытый протяжённый перепад в идеальную ступеньку. Однако применение морфологического повышения резкости к изображениям ведёт к возникновению эффекта постеризации (cartoon effect). Такие изображения выглядят неестественно, т.к. привычными для человека являются слегка размытые изображения с небольшим уровнем шума. В [121]

автором предложен алгоритм, позволяющий уменьшить длину перепада, как показано на рисунке 3.10b, и не приводящий к возникновению заметного эффекта постеризации. Целесообразно применять оба способа повышения резкости. Если сначала увеличить высоту перепада яркости, а затем уменьшить длину перепада, то добавленные вдоль перепада участки (см. рисунок 3.10a) расширятся, появится нежелательный ореол. Поэтому следует повышать резкость путём последовательного применения сначала уменьшения длины перепада яркости, а затем увеличения высоты перепада.

### **Последовательность шагов предлагаемого способа**

Схема предлагаемого способа подавления печатного растра [61, 62] показана на рисунке 3.11.

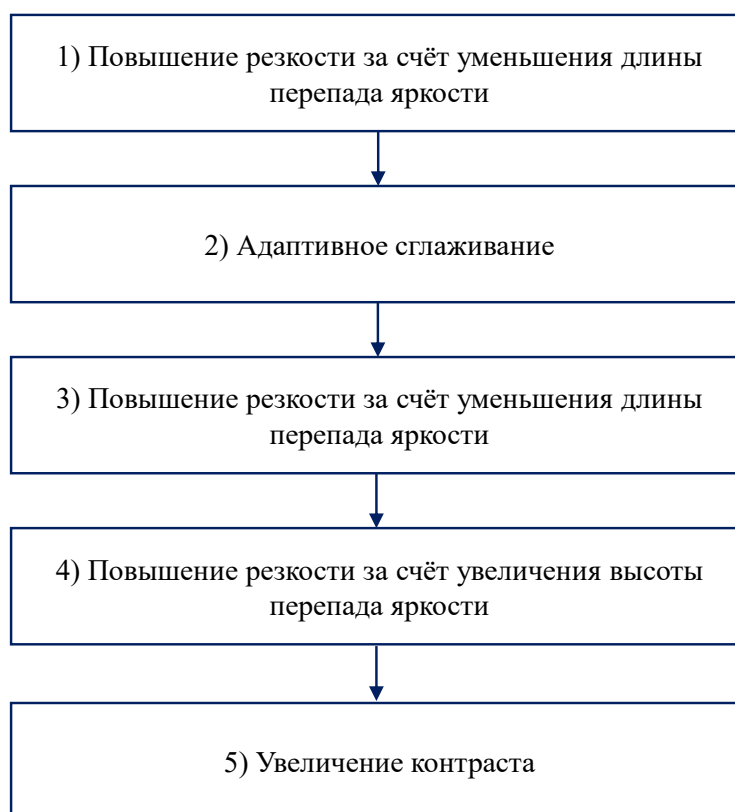


Рисунок 3.11 – Схема способа подавления печатного растра.

При сканировании растрованное изображение немного размывается. Это отрицательно влияет на определение степени похожести блоков в фильтре адаптивного сглаживания. Поэтому на шаге 1 выполняется повышение резкости за счёт уменьшения длины перепада яркости. Заметим, что при этом заодно корректируется размытие изображения, возникшее в ходе сканирования. На шаге 2 осуществляется адаптивное сглаживание цветовых каналов изображения. Далее повышается резкость сглаженного изображения, сначала на шаге 3 за счёт уменьшения длины перепада яркости, затем на шаге 4 за счёт увеличения высоты перепада. На шаге 5 происходит увеличение контраста изображения на 5%, т.к. приблизительно на эту величину контраст падает в результате сглаживания изображения. Подробно рассмотрим шаги 1 (и аналогичный ему 3), 2 и 4.

#### **Повышения резкости путём уменьшения длины перепада яркости**

Для повышения резкости за счёт уменьшения длины перепада яркости сканированное цветное изображение  $I$  конвертируется в полутоновое  $I_l$  по формуле (1.1). Квадратное окно размера  $N_o$  на  $N_o$  с центром в пикселе с координатами  $(r, c)$  перемещается по всем пикселям изображения  $I_l$ . Для каждого положения окна, обозначаемого  $\Omega$ , вычисляются  $T_1$  – среднее значение яркости пикселей меньше первого квартиля распределения яркостей пикселей в окне ( $h'_\Omega$ , см. формулы (1.3-1.4)) и  $T_2$  – среднее значение яркости пикселей больше третьего квартиля распределения яркостей пикселей в окне:

$$z_1 = \min \left( z \mid \sum_{k=0}^z h'_\Omega(k) \geq 0,25 \right), \quad (3.38)$$

$$z_2 = \max \left( z \mid \sum_{k=z}^{2^{D_l}-1} h'_\Omega(k) \geq 0,25 \right), \quad (3.39)$$

$$T_1 = \sum_{k=0}^{z_1} k h'_\Omega(k), \quad (3.40)$$

$$T_2 = \sum_{k=z_2}^{2^{D_l}-1} kh'_{\Omega}(k). \quad (3.41)$$

В области контурного перепада  $T_1$  соответствует уровню яркости с одной стороны от перепада, а  $T_2$  – уровню с другой стороны. Яркость пикселя в области контурного перепада  $I_I(r, c)$  преобразуются с помощью сигмовидной функции преобразования уровней  $s(x)$  (см. формулу (3.43)), где  $x \in [0, 1]$ . Предварительно  $I_I(r, c)$  нормализуется в диапазон  $[0, 1]$  с помощью функции  $f(x)$  (см. формулу (3.42)). В равномерной области разность  $T_2 - T_1$  не велика (меньше порога  $T_n$ ), в этом случае значение яркости пикселя с координатами  $(r, c)$  не изменяется (см. формулу (3.44)).

$$f(x) = \begin{cases} 0, x < T_1, \\ \frac{x - T_1}{T_2 - T_1}, x \in [T_1, T_2], \\ 1, x > T_2, \end{cases} \quad (3.42)$$

$$s(x) = \frac{x^2}{x^2 + (1 - x)^2}, \quad (3.43)$$

$$I'_I(r, c) = \begin{cases} I_I(r, c), & T_2 - T_1 < T_n, \\ T_1 + s(f(I_I(r, c)))(T_2 - T_1), & \text{иначе.} \end{cases} \quad (3.44)$$

В результате преобразования по формуле (3.44) контурные перепады на  $I'_I$  становятся более крутыми, длина перепадов сокращается (см. рисунок 3.10b), изображение выглядит более резким чем  $I_I$ . Эффект постеризации существенно менее выражен по сравнению с морфологическим повышением резкости [120], но может быть замечен. Для первого шага способа подавления печатного раstra данный эффект не имеет значения, но на третьем шаге его необходимо минимизировать. Этого можно достичь за счёт взвешенного суммирования  $I'_I(r, c)$  с исходным  $I_I(r, c)$ , где веса для  $I'_I(r, c)$  определяются функцией  $g(x)$ , веса для  $I_I(r, c)$  определяются как  $1 - g(x)$ , аргументом  $g(x)$  является разность  $T_2 - T_1$ .

$$g(x) = 1 - e^{-\frac{5(x-T_n)}{2^{D_I}}}, \quad (3.45)$$

$$I''_I(r, c) = I_I(r, c)(1 - g(T_2 - T_1)) + I'_I(r, c)g(T_2 - T_1). \quad (3.46)$$

### Коррекция для предотвращения изменения насыщенности

В литературе, как правило, не упоминается и в алгоритмах обработки изображений не учитывается тот факт, что изменение контраста цветного изображения ведёт к изменению его насыщенности. Покажем, что при повышении контраста насыщенность уменьшается. Насыщенность пикселя  $q = \{q_R, q_G, q_B\}$  в цветовом пространстве HSV равна [8]:

$$q_S = \frac{\max(q_R, q_G, q_B) - \min(q_R, q_G, q_B)}{\max(q_R, q_G, q_B)}. \quad (3.47)$$

Пусть  $q_R$  имеет максимальное значение, а  $q_G$  минимальное. После увеличения контраста, например  $q''_R = 1,1q_R$ , а  $q''_G = 0,9q_G$ . Тогда разница насыщенностей исходного и контрастированного изображения отрицательна:

$$\frac{q_R - q_G}{q_R} - \frac{1,1q_R - 0,9q_G}{1,1q_R} = -\frac{0,2q_G}{1,1q_R}. \quad (3.48)$$

Аналогичным образом показывается, что при понижении контраста насыщенность увеличивается.

Для предотвращения изменения насыщенности в обработанном изображении  $I'' = \{I''_R, I''_G, I''_B\}$ , для которого вычислен канал яркости  $I''_I$ , по сравнению с исходным изображением  $I = \{I_R, I_G, I_B\}$ , для которого вычислен канал яркости  $I_I$ , используется следующий способ коррекции:

$$I''_R(r, c) = \begin{cases} \frac{I_R(r, c)I''_I(r, c)}{I_I(r, c)}, & I_I(r, c) \neq 0, \\ I_R(r, c), & \text{иначе,} \end{cases} \quad (3.49)$$

$$I''_G(r, c) = \begin{cases} \frac{I_G(r, c)I''_I(r, c)}{I_I(r, c)}, & I_I(r, c) \neq 0, \\ I_G(r, c), & \text{иначе,} \end{cases} \quad (3.50)$$

$$I''_B(r, c) = \begin{cases} \frac{I_B(r, c)I''_I(r, c)}{I_I(r, c)}, & I_I(r, c) \neq 0, \\ I_B(r, c), & \text{иначе.} \end{cases} \quad (3.51)$$

Данный способ коррекции необходимо применять после каждой операции обработки цветного изображения, связанной с изменением канала яркости. Причём коррекция выполняется в обоих случаях:

1. Когда в процессе обработки цветного изображения изменяются цветовые каналы  $I''_R, I''_G, I''_B$ . В этом случае канал яркости  $I''_I$  вычисляется из значений цветовых каналов по формуле (1.1), а затем применяются формулы (3.49)-(3.51).
2. Когда в процессе обработки изображения изменяется канал яркости  $I''_I$ . В этом случае по формулам (3.49)-(3.51) вычисляются значения цветовых каналов  $I''_R, I''_G, I''_B$ .

Для уменьшения времени обработки можно применить данную коррекцию для группы подряд идущих операций. В тексте ниже в ряде случаев опущено упоминание коррекции для предотвращения изменения насыщенности или получения значений цветовых каналов, хотя фактически это выполняется после большинства операций улучшения качества изображений.

### Адаптивное сглаживание

На втором шаге способа подавления печатного раstra применяется алгоритм адаптивного сглаживания, в котором квадратное окно размера  $N_w$  на  $N_w$  с центром в пикселе с координатами  $(r, c)$  перемещается по всем пикселям цветного изображения  $I = \{I_R, I_G, I_B\}$ , для каждого  $(r, c)$  вычисляется разница  $S(r, c)$  между блоком размера  $N_b$  на  $N_b$  с центром в пикселе с координатами  $(r, c)$  и блоком такого же размера с центром в пикселе с координатами  $(r + u, c + v)$  (см. рисунок 3.12), причем значения  $u$  и  $v$  есть псевдослучайные числа [122] в диапазоне  $[-N_w/2, N_w/2]$  при условии  $u \neq 0, v \neq 0$ :

$$\begin{aligned}
S(r, c) = & \sum_{i=-\frac{N_b}{2}}^{\frac{N_b}{2}} \sum_{j=-\frac{N_b}{2}}^{\frac{N_b}{2}} \left( I_R(r+i, c+j) - I_R(r+u+i, c+v+j) \right)^2 + \\
& + \left( I_G(r+i, c+j) - I_G(r+u+i, c+v+j) \right)^2 + \\
& + \left( I_B(r+i, c+j) - I_B(r+u+i, c+v+j) \right)^2.
\end{aligned} \tag{3.52}$$

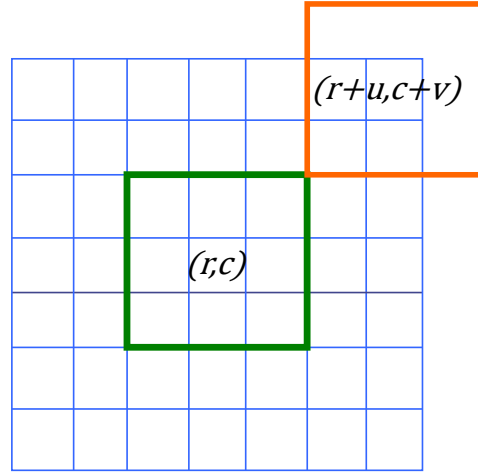


Рисунок 3.12 – Окно и блоки в окне при адаптивном размытии.

Если разница  $S(r, c)$  невелика (меньше порога  $T_s$ ), то предполагается, что пиксель с координатами  $(r, c)$  находится на однородной пространственно повторяющейся текстуре, образованной печатным растром, и результат сглаживания в данном пикселе  $I'(r, c)$  равен  $I_{\sigma_s}^*(r, c)$  – результату гауссова размытия с дисперсией  $\sigma_s$ .  $I_{\sigma_s}^*(r, c)$  получается с помощью свёртки каждого канала цветного изображения с одинаковым ядром в виде двумерного гауссиана. Если разница  $S(r, c)$ , больше порога  $T_s$ , то предполагается, что пиксель с координатами  $(r, c)$  находится в области заметного изменения текстуры, образованной печатным растром, и результат сглаживания в данном пикселе  $I'(r, c)$  равен  $I_{\sigma_m}^*(r, c)$  – результату гауссова размытия с дисперсией  $\sigma_m$ . Причем  $\sigma_s > \sigma_m$ , т.е. области однородных текстур размываются сильнее, чем области, где происходит резкое



изменение вида текстуры раstra, что, вероятно, соответствует границам на исходном изображении до растривания.

$$I'(r, c) = \begin{cases} I_{\sigma_s}^*(r, c), & S(r, c) < T_s, \\ I_{\sigma_m}^*(r, c), & \text{иначе.} \end{cases} \quad (3.53)$$

### **Повышение резкости путём увеличения высоты контурного перепада**

На четвёртом шаге способа подавления печатного раstra выполняется повышение резкости с помощью увеличения высоты контурного перепада. Одним из наиболее известных алгоритмов такого типа является фильтр нерезкого маскирования, в котором к изображению добавляется умноженная на коэффициент  $k$  разность изображения и его размытой фильтром Гаусса копии [78]. Для ряда сочетаний дисперсии фильтра Гаусса и  $k$  возможно возникновение видимых искажений в виде ореолов (эффект гало) вдоль контурных перепадов. В [121] автором предложен подход, позволяющий избежать возникновения ореолов за счёт использования для обработки канала яркости  $I_I$  вместо фильтра Гаусса модифицированного билатерального фильтра. В классическом билатеральном фильтре [123] в качестве пространственного ядра и функции весов по диапазону яркостей (называемой также функцией радиометрического расстояния или функцией предохранения границ, edge-stop function) используются двумерный и одномерный гауссианы соответственно. В модифицированном фильтре предлагается применять усредняющее («плоское», flat) ядро по пространству и весовую функцию по диапазону яркостей  $\Phi(x)$ , которая похожа на гауссиан, но с увеличением  $|x|$  уменьшается более плавно при значении параметра  $\mu$  больше двух [124]:

$$\Phi(x) = \frac{1}{\sqrt{1 + \left| \frac{x^2}{\sigma_r^2} \right|^\mu}}, \quad (3.54)$$

где  $\sigma_r^2$  – по смыслу аналогичен дисперсии гауссиана и устанавливает величину перепада, которая считается “полезной” границей между областями.

В модифицированном билатеральном фильтре квадратное окно размера  $N_o$  на  $N_o$  с центром в пикселе с координатами  $(r, c)$  перемещается по всем пикселям изображения  $I_I$ , результат фильтрации вычисляется следующим образом:

$$I'_I(r, c) = \frac{\sum_{i=-\frac{N_o}{2}}^{\frac{N_o}{2}} \sum_{j=-\frac{N_o}{2}}^{\frac{N_o}{2}} I_I(r, c) \Phi(I_I(r, c) - I_I(r + i, c + j))}{N_o^2 \sum_{i=-\frac{N_o}{2}}^{\frac{N_o}{2}} \sum_{j=-\frac{N_o}{2}}^{\frac{N_o}{2}} \Phi(I_I(r, c) - I_I(r + i, c + j))}. \quad (3.55)$$

Фильтр нерезкого маскирования с использованием модифицированного билатерального фильтра:

$$I''_I(r, c) = \begin{cases} I_I(r, c), & |I_I(r, c) - I'_I(r, c)| < T_n, \\ I_I(r, c) + k(I_I(r, c) - I'_I(r, c)), & \text{иначе.} \end{cases} \quad (3.56)$$

Значения в каналах цветного изображения  $I''$  вычисляются по формулам (3.49)-(3.51).

### Результаты

Параметры фильтров повышения резкости и адаптивного сглаживания настроены в зависимости от разрешения сканирования, например для разрешения 600 точек на дюйм и глубины цвета на канал  $D_l = 8$ :  $N_o = 11$ ;  $T_n = 25$ ;  $N_w = 7$ ;  $N_b = 3$ ;  $T_s = 25000$ ;  $\sigma_s = 3$ ;  $\sigma_m = 5$ ;  $\mu = 3$ ;  $\sigma_r = 10$ ;  $k = 0,5$ . На рисунке 3.13 показаны примеры результатов обработки при подавлении печатного раstra для увеличенного в 3 раза фрагмента изображения. Видно, что печатный растр успешно подавлен, а полученное в результате обработки изображение является достаточно чётким.

Качество разработанного способа подавления раstra по сравнению с альтернативными алгоритмами и ПО оценивалось на основе показателей, вычисляемых по одной из тестовых таблиц, используемых в стандарте ИСО для оценки качества копирования [32], а также на основе опроса пользователей. Сравнение выполняется для ПО Descreen [113] и SilverFast HDR [125], а также алгоритма HFD [118]. Следует отметить, что литературе HFD описан только для

изображений чёрно-белого растра. Данный алгоритм независимо применялся к каждому цветовому каналу сканированного изображения, что в ряде случаев приводило к возникновению цветовых искажений.

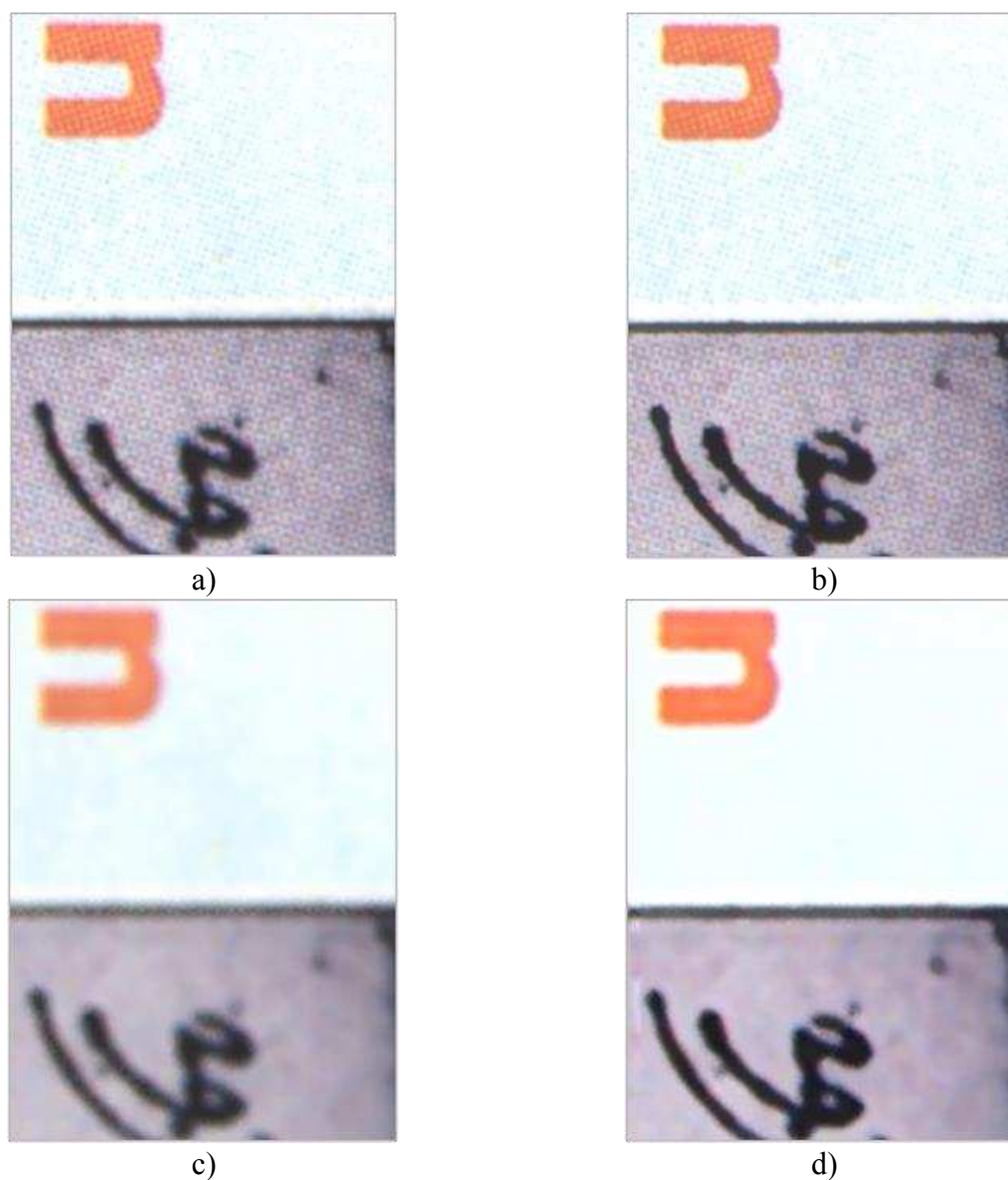


Рисунок 3.13 – Примеры результатов этапов обработки при подавлении печатного растра: а) исходное сканированное изображение; б) результат повышения резкости за счёт уменьшения длины перепада; в) результат адаптивного размытия; г) итоговый результат.

На рисунке 3.14 приведена напечатанная типографским способом тестовая таблица и обозначены три области, на которых вычисляются показатели качества.

Радиальные миры в области 1 (так называемые black и white Siemens star) используются для оценки степени размытия высокочастотных структур в ходе подавления раstra. Оценка резкости контурного перепада между контрастными областями с чёрно-белым растром выполняется для области 2. Оценка искажения цветов в результате работы способа осуществляется в области цветного раstra 3. Тестовая таблица отсканирована с разрешением 600 dpi и глубиной цвета 24 bpr, затем обработана различными способами подавления печатного раstra.

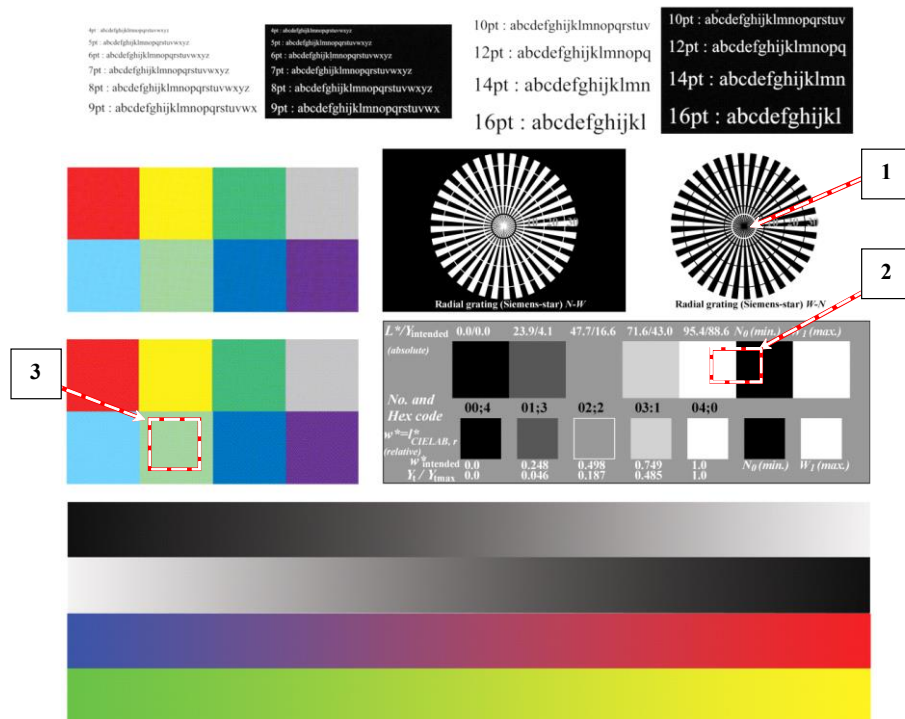


Рисунок 3.14 – Тестовая таблица с обозначением областей, используемых для количественного сравнения способов подавления печатного раstra.

Таблица 3.5. Увеличение диаметра круга в пикселях в области 1 (см. рисунок 3.14), в котором неразличимы линии радиальной миры

Элемент	Descreeen	HFD	SilverFast	Предлагаемый
Black star	8	9	54	13
White star	6	8	49	11

В таблице 3.5 приведены значения увеличения диаметра круга, в котором неразличимы линии радиальной миры, по сравнению с необработанным сканированным изображением. Предлагаемый способ сохраняет высокочастотные чёрно-белые детали чуть хуже, чем Descreen и HFD, но увеличение диаметра составляет лишь около половины мм, что незаметно невооружённым глазом.

Длина перепада между квадратными участками чёрно-белого раstra приведена в таблице 3.6. Вычисляются средние значения яркости светло-серой и тёмно-серой областей, между которыми есть контурный перепад. Усредняются значения яркости в столбцах, в результате чего получается усреднённый профиль яркости. На профиле краем перепада считаются ближайшие точки, значения яркости которых более чем на 10 градаций яркости отличаются от средних яркостей светло-серой и тёмно-серой областей. Длина перепада — это абсолютная разность между координатами краёв перепада. Предлагаемый способ обеспечивает границу между областями даже более резкую чем на исходном сканированном изображении.

Таблица 3.6. Оценка резкости границы в области 2 (см. рисунок 3.14)

Параметр	Исходный скан	Descreen	HFD	SilverFast	Предлагаемый
Длина перепада, пикс.	6	6	6	9	4

Таблица 3.7. Оценка искажений цвета в области 3 (см. рисунок 3.14).

Параметр	Descreen	HFD	SilverFast	Предлагаемый
$\Delta H$	1	-12	9	-1
$\Delta S$	-3	-3	12	0
$\Delta V$	-4	-4	0	3

Значения пикселей области 3 исходного сканированного и обработанного изображений конвертировано в цветовое пространство HSV [8], где  $H, S, V \in [0, 255]$ , затем для каждого канала обоих изображений вычислено среднее значение. Результаты поканальной разности средних значений исходного и обработанного изображений приведены в таблице 3.7. В отличие от альтернативных решений у предлагаемого способа подавления печатного растра практически отсутствует искажение цвета.

Для сравнительной оценки качества способов подавления печатного растра три журнальные страницы, напечатанные типографским способом с использованием четырёхкрасочной печати, отсканированы с разрешением 600 dpi, далее различными способами на изображениях подавлялся растр, затем обработанные изображения были напечатаны с помощью МФП Samsung CLX-6240 на одинаковой бумаге. Помимо обработанных изображений были напечатаны и исходные. Изображения показаны на рисунке 3.15. Их особенностью является присутствие как текста, причем не только чёрного цвета, так и картинок. В опросе участвовало 30 человек, которые были проинструктированы для каждого из трёх изображений ранжировать пять напечатанных страниц (4 способа подавления растра и необработанное) в порядке убывания качества, где под качеством понимались чёткость текста и картинок, отсутствие видимых артефактов. Интервьюируемые не знали способ обработки. Самое качественное изображение получало оценку  $E$  в 5 баллов, следующее 4, и так далее до 1.

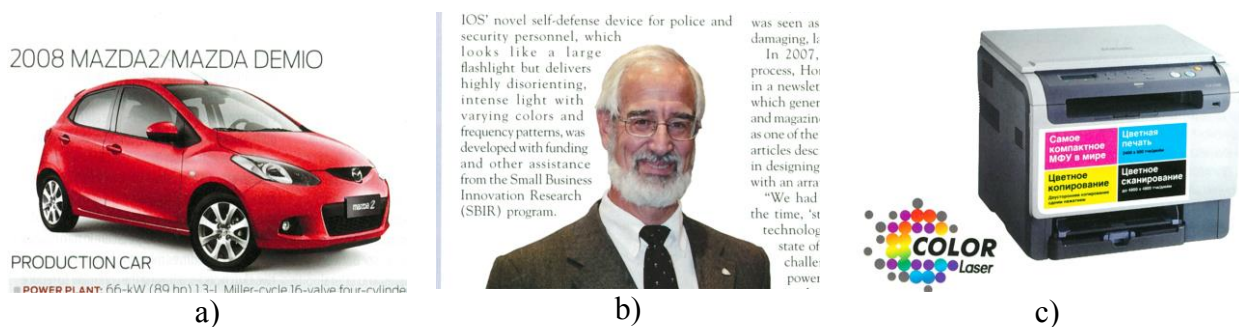


Рисунок 3.15 – Изображения, использовавшиеся при опросе

Таблица 3.8 содержит оценки  $\bar{E}$ , усреднённые по всем участвовавшим в опросе, для каждого из трёх рассматриваемых изображений. Предложенный способ имеет самые высокие оценки. Участвовавшими в опросе было отмечено, что он обеспечивает наиболее четкий текст без пятен и артефактов на буквах. Самое низкое качество имеют необработанное изображение и результат обработки HFD, но HFD не разрабатывался для работы с цветными изображениями. Descreen и SilverFast обеспечивают визуально хорошие результаты по сравнению с необработанным изображением, но чёткость текста на их результатах заметно хуже по сравнению предлагаемым способом, особенно у SilverFast, что видно по оценке для изображения с рисунка 3.15b.

Таблица 3.8. Средние оценки пользователей для изображений с рисунка 3.15.

Способ	$\bar{E}$ для а	$\bar{E}$ для b	$\bar{E}$ для с
Исходный скан	1,3	2,2	1,5
Descreen	3,9	4,2	3,3
HFD	2,2	1,9	1,8
SilverFast HDR	3,0	1,9	3,6
Предлагаемый	<b>4,6</b>	<b>4,8</b>	<b>4,8</b>

### 3.6. Улучшение глобального и локального контраста

Иногда встречаются документы, отдельные элементы которых плохо различимы в силу низкого глобального или локального контраста. Например, это может быть чёрный текст на тёмном фоне, или, наоборот, белый текст на светлом фоне. При сканировании и копировании видимость подобных элементов может ещё больше уменьшиться, поэтому целесообразно при сканировании умеренно повышать локальный контраст.

### Адаптивная функция преобразования уровней

В [64, 65, 126] автором предложен способ повышения контраста. Рисунок 3.16 иллюстрирует основную идею способа. Гистограмма фрагмента изображения с чёрным текстом на сером фоне смещена влево. Это можно использовать для регулировки положения сигмовидной функции преобразования уровней так, чтобы контраст между тёмными оттенками увеличился. Гистограмма фрагмента изображения со светло-зелёным текстом на светло-сером фоне смещена вправо. Требуется установить положение сигмовидной функции преобразования уровней для увеличения контраста светлых оттенков. Таким образом, ключевыми вопросами являются: какую функцию выбрать, чтобы менять её форму в зависимости от положения гистограммы, и как добиться плавного изменения формы между соседними пикселями для предотвращения возникновения артефактов.

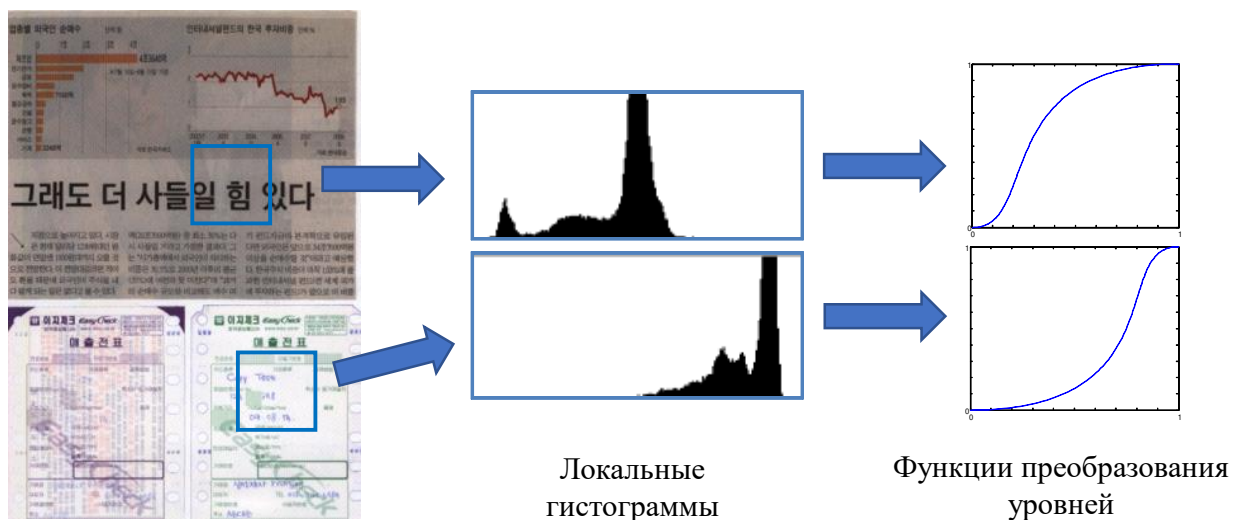


Рисунок 3.16 – Иллюстрация идеи повышения локального контраста.

В качестве функции, которой можно задать сигмовидную форму, предлагается использовать интерполяционный кубический сплайн Эрмита в параметрической форме [8]:



$$f(t) = (1 - t^2 + 2t^3)P_0 + (3t^2 - 2t^3)P_1 + (t - 2t^2 + t^3)Q_0 + (t^3 - t^2)Q_1, \quad (3.57)$$

где параметр  $t \in [0,1]$ ; сплайн начинается в точке  $P_0$  с координатами  $(P_{0x}, P_{0y})$ ; сплайн заканчивается в точке  $P_1$  с координатами  $(P_{1x}, P_{1y})$ ; форма сплайна задаётся векторами  $Q_0$  и  $Q_1$ . Вектор  $Q_0$  имеет координаты  $(Q_{0x}, Q_{0y})$ , начинается в  $P_0$  и в этой точке является касательным к кривой; вектор  $Q_1$  имеет координаты  $(Q_{1x}, Q_{1y})$ , начинается в  $P_1$  и в этой точке является касательным к кривой.

Значения  $y$  координат точек и векторов зафиксированы следующим образом:  $P_{0y} = 0$ ,  $P_{1y} = 1$ ,  $Q_{0y} = 0$ ,  $Q_{1y} = 0$ . Значения  $x$  координат точек находятся из гистограммы  $h'$  (см. (1.4)) всего канала яркости  $I_I$ , вычисляемого по формуле (1.1):

$$P_{0x} = \min(\min\{i | h'(i) \geq T_b\}, \min\{i | \sum_{k=0}^i h'(k) \geq T_a\}) / (2^{D_I} - 1), \quad (3.58)$$

$$P_{1x} = \max(\max\{i | h'(i) \geq T_b\}, \max\{i | \sum_{k=i}^{2^{D_I}-1} h'(k) \geq T_a\}) / (2^{D_I} - 1), \quad (3.59)$$

где  $T_a = 0,005$ ,  $T_b = 0,003$ . Также из  $h'$  вычисляется порог бинаризации  $T$ , например можно использовать алгоритм Оцу [85].

### Схема преобразования данных способа повышения контраста

Схема преобразования данных в методе повышения локального контраста показана на рисунке 3.17. Локальное окно  $\Omega$  размера  $N_l$  на  $N_l$  перемещается по  $I_I$  с шагом  $\Delta s$  по строкам и по столбцам. Для каждого окна вычисляется гистограмма  $h_{\Omega}'$ , затем из гистограммы определяются 1-й и 99-й перцентили  $L_1$  и  $L_{99}$ . Если разность  $L_{99} - L_1$  больше равно  $R$ , то из  $h_{\Omega}'$  вычисляется порог бинаризации  $T_{\Omega}$ . Формируется изображение  $I_U$ , в котором пиксели соответствуют положениям окна  $\Omega$ . Размер  $I_U$  меньше  $I_I$ , он тривиальным образом определяется из  $W$  и  $H$  (размеров  $I_I$ ), размера окна  $N_l$  и шага  $\Delta s$ . Значения пикселя  $I_U(r', c')$  для окна  $\Omega$ :

$$I_U(r', c') = \begin{cases} T, & L_{99} - L_1 < R, \\ T_{\Omega}, & \text{иначе.} \end{cases} \quad (3.60)$$

Далее изображение сглаживается фильтром низких частот (ФНЧ). Применяется усредняющий фильтр с размером ядра свёртки 5 на 5. Цель сглаживания – исключить резкие изменения соседних значений  $I_U$ , так как данные значения служат для регулировки векторов сплайна, т.е. формы функции преобразования уровней.

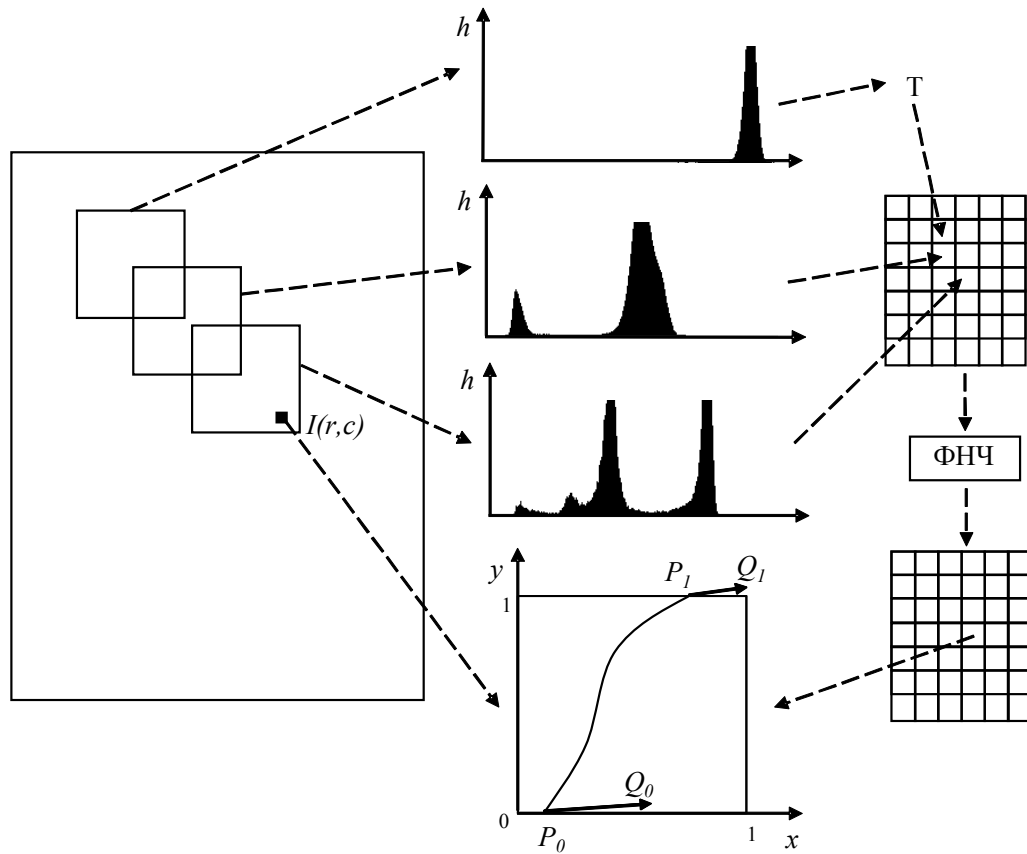


Рисунок 3.17 – Схема преобразования данных в методе повышения локального контраста.

Поскольку размеры  $I_I$  и  $I_U$  известны, то для каждого пикселя изображения  $I_I$  значения  $x$  координат векторов  $Q_0$  и  $Q_1$  находят, исходя из значения  $U$ , извлекаемого из  $I_U$  с помощью билинейной интерполяции [8]:

$$Q_{0x} = \begin{cases} 1 + 3 \left( \frac{U}{2^{D_I} - 1} - 0,5 \right), & \frac{U}{2^{D_I} - 1} > 0,5, \\ 1, & \text{иначе,} \end{cases} \quad (3.61)$$

$$Q_{1x} = \begin{cases} 1, & \frac{U}{2^{D_l} - 1} > 0,5, \\ 1 + 3 \left( 0,5 - \frac{U}{2^{D_l} - 1} \right), & \text{иначе.} \end{cases} \quad (3.62)$$

Далее вычисляется новое значение яркости пикселя из  $I_l$  по получившейся для данного пикселя функции преобразования уровней. Затем находятся значения цветовых каналов по формулам (3.49)-(3.51).

Заметим, что в формуле (3.57) кривая  $f(t)$  задана в параметрическом виде,  $x$  и  $y$ , соответствующие значениям яркости на входе и выходе функции преобразования уровней, лежат в диапазоне от 0 до 1. Для получения в таблице перекодировки целочисленных значений яркости на выходе в зависимости от яркости на входе,  $t$  в (3.57) изменяется с малым шагом  $\Delta t$ , по  $f(t)$  вычисляются значения  $x$  и  $y$ , которые умножаются на  $2^{D_l}$  и округляются до ближайшего целого, из множества пар целочисленных  $x$  и  $y$  оставляют пары с уникальным значением  $x$ , которое является номером столбца в таблице, куда помещается целочисленное значение  $y$ .

Если контраст всего сканированного изображения низкий, то предложенный способ повышает его за счёт установки  $P_0 > 0$  и  $P_1 < 1$ . Изменение формы кривой преобразования уровня в зависимости от распределения яркостей в локальной окрестности позволяет улучшить локальный контраст.

### Результаты

Оценивать эффект повышения локального контраста можно с помощью контраста Майкельсона, также известного как видимость [127], вычисляемого для  $\Psi_i$  областей канала яркости  $I_l$  изображения тестовой таблицы из ИСО стандарта [32]:

$$C_{m_i} = \frac{\max_{(r,c) \in \Psi_i} I_l(r,c) - \min_{(r,c) \in \Psi_i} I_l(r,c)}{\max_{(r,c) \in \Psi_i} I_l(r,c) + \min_{(r,c) \in \Psi_i} I_l(r,c)}. \quad (3.63)$$

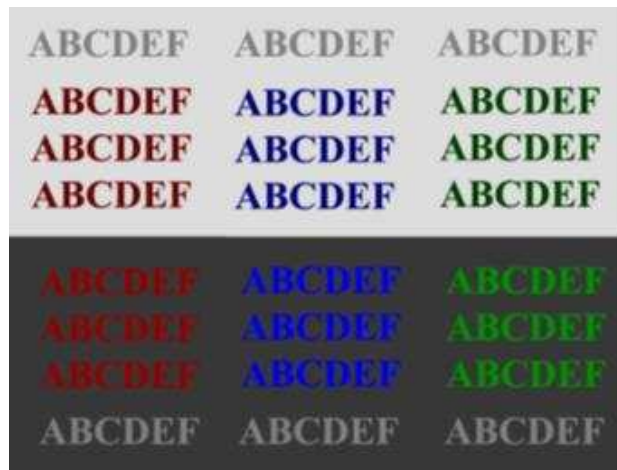


Рисунок 3.18 – Фрагмент тестовой таблицы для оценки методов улучшения локального контраста.

Фрагмент тестовой таблицы для оценки методов улучшения локального контраста показан на рисунке 3.18. На данном изображении рассматривается 8 областей  $\Psi_i$ : по 3 для цветного текста на светло-сером и тёмно-сером фоне, 2 для серого текста на светло-сером и тёмно-сером фоне. В таблице 3.9 для ряда способов коррекции глобального и локального контраста, а также исходного изображения, приведены: количество областей, в которых  $C_{m_i}$  увеличился;  $\bar{C}_m$  – среднее значение по восьми  $C_{m_i}$ ; минимальное из  $C_{m_i}$ ; присутствие видимых артефактов в обработанной изображении. В отличие от других способов, предлагаемый увеличил локальный контраст во всех восьми областях, причем увеличение контраста больше, чем у других способов, и видимых артефактов на изображении не возникает.

Пример работы метода улучшения локального контраста показан на рисунке 3.19. Текст на документе стал более различимым, обработанное изображение выглядит более насыщенным, что позитивно оценивается пользователями.

**범칙금영**  
Traffic Citation Fine

통고서번호 Report Number	0137-2007-1- 1118 011064
납부고지일 Payment Notification Date	2007년(Y) 8월(M) 24
(1차) 납부기한 Payment Deadline Date (1st)	2007년(Y) 9월(M) 3
(2차) 납부기한 Payment Deadline Date (2nd)	2007년(Y) 9월(M) 23
성명 Name	송수환

※ 1차 납부기한 다음 날부터 2차 납부기한 (20일) 이내에는 20/100%  
위 금액을 정히 영수함.  
Receipt of payment acknowledged  
20 년(Y) 월(M) 일(D)  
은행 본 · 지점(국고대리점 및 국고 수납대리점), 우체국 또  
은행 지점

※인터넷 납부는 고지일로부터 3일 경과후 종합 납부 서

**범칙금납**  
Notice of Penalt

통고서번호(Report Number) 0137-2007-1- 111

a)

**범칙금영**  
Traffic Citation Fine

통고서번호 Report Number	0137-2007-1- 1118 011064
납부고지일 Payment Notification Date	2007년(Y) 8월(M) 24
(1차) 납부기한 Payment Deadline Date (1st)	2007년(Y) 9월(M) 3
(2차) 납부기한 Payment Deadline Date (2nd)	2007년(Y) 9월(M) 23
성명 Name	송수환

※ 1차 납부기한 다음 날부터 2차 납부기한 (20일) 이내에는 20/100%  
위 금액을 정히 영수함.  
Receipt of payment acknowledged  
20 년(Y) 월(M) 일(D)  
은행 본 · 지점(국고대리점 및 국고 수납대리점), 우체국 또  
은행 지점

※인터넷 납부는 고지일로부터 3일 경과후 종합 납부 서

**범칙금납**  
Notice of Penalt

통고서번호(Report Number) 0137-2007-1- 111

b)

Рисунок 3.19 – Пример работы метода улучшения локального контраста: а) исходное сканированное изображение; б) результат обработки предлагаемым методом.

Таблица 3.9. Сравнение способов улучшения контраста изображений

Метод обработки	Количество областей, где $C_{m_i}$ увеличился	$\bar{C}_m$	$C_{m_{\min}}$	Заметные артефакты
Исходное изображение	-	0.40	0.05	-
Photoshop Auto Contrast	0	0.40	0.05	Нет
Photoshop Auto Levels	2	0.40	0.06	Нет
Multiscale Retinex [128]	4	0.42	0.17	Да, ореол
Moroney [129]	6	0.43	0.05	Да, ореол
Коррекция теней [130]	6	0.43	0.01	Нет
CLANE [131]	4	0.40	0.09	Да, пятна
Предлагаемый	8	0.54	0.26	Нет

### **3.7. Примеры применения методологии НИР и методики режимов сканирования**

Создание новых или совершенствование существующих режимов работы МФП выполняется в ходе НИР. В данном разделе приведены два примера применения методологии НИР, предложенной во второй главе, и методики режимов сканирования и копирования.

#### **3.7.1. Сканирование и сохранение документа в PDF с векторизацией контуров символов**

Как отмечено в первой главе, режим сохранения сканированного изображения в метафайловом формате, в первую очередь PDF, является обязательным для большинства МФП. При сохранении в метафайл изображения текста, картинок и фона сохраняются отдельно. Для уменьшения размера файла текст и картинки сжимаются. В результате качество текста деградирует: форма и цвет символов изменяются, что особенно заметно при повторной печати. При оптическом распознавании символов для сканированного изображения такой копии возникает заметно больше ошибок по сравнению с обработкой оригинала.

#### **Этапы методологии НИР при разработке режима сохранения документов**

Для НИР была сформулирована следующая тема: разработка такого режима сканирования и сохранения в PDF, который обеспечит, как минимум, два цикла повторной печати и сканирования без существенной потери качества распознавания текста. Размер файла для страницы А4 сканированной с разрешением 300 точек на дюйм не должен превышать 700 kb, время обработки такой страницы на типичном офисном ПК менее 2 с.

В таблице 3.10 приведены основное содержание и результаты этапов НИР по разработке данного режима сканирования в соответствии с предложенной во второй главе методологией. Выполнение НИР по данной методологии с применением методики режимов сканирования позволило за несколько месяцев разработать алгоритмы и ПО нового режима сохранения сканированного документа в PDF.

Таблица 3.10. Этапы НИР по разработке режима сканирования и преобразования документа в PDF в соответствии с предложенной методологией

Этап НИР	Содержание
1. Поиск идеи	<p>Целесообразно применить методику режимов сканирования. Ключевой вопрос: как обрабатывать и сохранять области текста? Возможные подходы для обработки областей текста:</p> <ul style="list-style-type: none"> <li>а) сохранять текст в виде качественного растрового изображения, но это ведет к большому размеру файла;</li> <li>б) распознавать текст и шрифт, но это нереализуемо для большого количества языков и шрифтов, особенно во встроенном ПО;</li> <li>с) конвертировать изображения символов текста в векторную графику, идея выглядит перспективной, подобная функция (“Trace&amp;Smooth PDF”) используется, например в МФП Canon iR C2550 [132].</li> </ul> <p>При JPEG сжатии картинок целесообразно настраивать таблицы квантования, чтобы обеспечить наилучшее качество для заданного размера.</p>
2. Макет	<p>Вручную написан код по рисованию контура одного символа прямыми и кривыми, скриптом код продублирован для рисования 2000 символов на странице размера А4, код вставлен в PDF без сжатия, документ напечатан для оценки визуального качества, PDF сжат zip для оценки размера PDF со сжатием.</p>
3. Предварительная оценка пользователем	<p>Визуальное качество напечатанного изображения высокое. Размер сжатого PDF существенно меньше метафайла, в котором</p>

	текст хранится в виде растрового изображения сравнимого качества.
4. Анализ патентов	Существуют патенты, описывающие векторизацию символов текста, например [133], однако они не рассматривают обработку картинок и фона, а также сохранение метафайла.
5. Выбор критериев	Сформулированы критерии качества векторизации контуров символов [53], из которых с помощью взвешивания по методу анализа иерархий [45] получен общий критерий качества. Заданы требования на размер файла PDF и время обработки для встроенной платформы и на ПК.
6. Обзор литературы	Проведен обзор публикаций по методам векторизации бинарных областей растрового изображения [134–139].
7. Первый прототип	Реализован на MATLAB только для областей текста.
8. Оценка качества	Выявлено низкое качество векторизации символов маленького размера (меньше 11 типографских пунктов). Потребовалась доработка алгоритма и прототипа.
9. Идентификация этапов в конвейере обработки	Предпочтительна реализация во встроенном ПО. Альтернативные варианты: TWAIN драйвер и прикладное ПО (см. рисунок 1.3).
10. Второй прототип	Реализован на C++ для всех этапов обработки.
11. Анализ лицензий используемого ПО	Для сжатия в PDF используются библиотеки zlib [140] и libjpeg [141], которые допускают коммерческое использование при условии указания применения библиотеки.
12. Оценка качества и производительности	При копировании качество деградирует медленнее, чем при сохранении текста в виде изображения. При сравнимом качестве данный способ обеспечивает минимальный размер файла. Страница А4 сканированная с разрешением 300 dpi обрабатывается около 1 с на ПК и около 1.5 с на встроенной платформе.
13. Защита интеллектуальной собственности	Подготовлены две заявки на изобретение, по которым впоследствии выданы патенты [67, 68].



14. Анализ требований к описанию	Код на C++ оформлен в соответствии с требованиями производственных подразделений.
15. Документация и тесты	Подготовлены отчет по НИР и набор тестовых изображений с результатами, получающимися на каждом этапе обработки.

### **Последовательность шагов режима сохранения документа в PDF**

Поскольку для реализации данного режима требуется устранять скос, подавлять печатный растр, сегментировать изображение на области фона, текста и картинок, последовательность обработки почти полностью соответствует рекомендованной в методике режимов сканирования (рисунок 3.2) со следующими изменениями: на этапе 6 оценка параметров по изображению низкого разрешения не производится; на этапе 11 для областей текста выполняется векторизация контуров и оценка цвета символов, а для картинок выполняется JPEG сжатие с настройкой таблиц квантования с целью обеспечить наилучшее качество для заданного размера сжатого изображения [53]; на этапе 12 символы текста сохраняются в PDF в виде заполненных замкнутых путей (path) – последовательности отрезков прямых и кубических кривых Безье [8], затем потоки (stream) PDF, содержащие пути, сжимаются без потерь алгоритмом deflate [10].

Кратко рассмотрим обработку областей текста [55]. Каждая область текста, полученная в результате сегментации на этапе 5 (рисунок 3.2), рассматривается отдельно. Области текста на изображении низкого разрешения соответствует область на изображении заданного разрешения, для которой и осуществляется обработка. Для яркостного канала изображения выполняется бинаризация области. Порог может определяться, например, с помощью алгоритма Оцу [85]. Поскольку в рассматриваемых областях пикселей фона больше, чем пикселей символов, то делается вывод о присутствии в данной области тёмных символов на светлом фоне или наоборот. Далее на бинарном изображении определяются связные области, и каждая область обрабатывается отдельно. Например, для буквы “i” существует две связные области, каждая из которых обрабатывается независимо друг от друга.

Последовательность шагов обработки для каждой связной области показана на рисунке 3.20. На шаге 1 выполняется оценка цвета области по тем пикселям сканированного цветного изображения, которые соответствуют пикселям связной области бинарного изображения. В первоначальной версии алгоритма далее выполнялся поиск пикселей контура связной области с последующим преобразованием контура в путь. Тесты на восьмом этапе НИР (см. таблицу 3.10) показали, что для символов размера 10 типографских пунктов и менее при разрешении сканирования 300 dpi контур получается неестественным и непохожим на действительный вид символов (см. рис 3.21a), причём повышение разрешения сканирования улучшает внешний вид контура. Поэтому в первоначальный алгоритм было добавлено увеличение изображения области символа, если размеры области меньше заданного порога. Также было замечено, что поиск контура и его векторизация для символов большого размера (больше 26 пунктов при разрешении сканирования 300 dpi) занимает значительно больше времени, чем для символов меньшего размера, причём уменьшение количества строк и столбцов в 2 раза не влияет на визуальное качество контура. Поэтому принято решение масштабировать изображения символов в predetermined диапазон размеров с сохранением соотношения сторон. Целесообразно масштабировать изображение области символа не для бинарного изображения, а для исходного сканированного (шаг 2 на рисунке 3.20), а затем снова выполнять бинаризацию (шаг 3). На рис 3.21b показано улучшение формы контура за счет применения шагов 2 и 3.

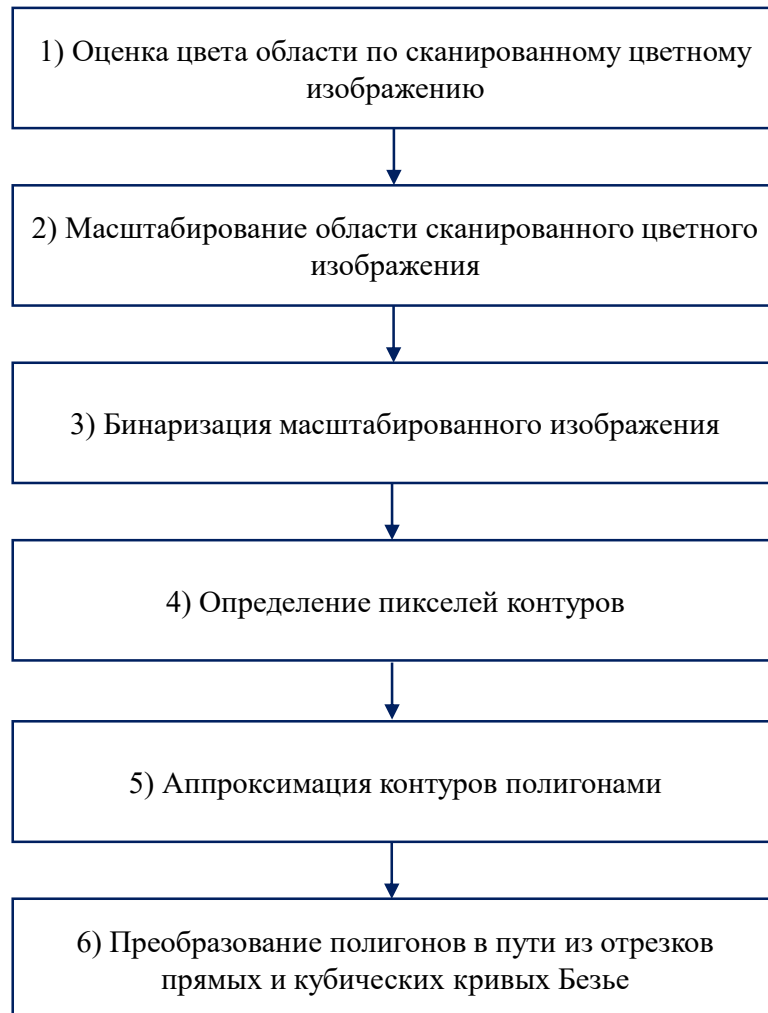


Рисунок 3.20 – Схема обработки при векторизации контуров связных областей.

На шаге 4 выполняется определение пикселей внешнего и, если таковые имеются, внутренних контуров связной области [142]. Исходя из того, что команда заполнения пути цветом в формате PDF [10] по правилу «ненулевого числа витков» (non-zero winding) [8] имеет длину один символ, а по правилу «нечётности» (even-odd) – два символа, с точки зрения сокращения размера файла имеет смысл использовать более короткую команду. Поэтому пиксели внутренних и внешнего контура обходятся в противоположных направлениях: для определённости внешний контур трассируется по часовой стрелке, а внутренние – против часовой стрелки.



Рисунок 3.21 – Влияние шагов 2 и 3 (см. рисунок 3.20) на форму контура: а) шаги 2 и 3 не применялись; б) шаги 2 и 3 выполнялись.

На шаге 5 пиксели контуров аппроксимируются полигонами. При аппроксимации решается задача минимизации количества вершин полигона (так называемая, *min-#* проблема) при заданных величинах максимального и суммарного отклонения евклидова расстояния от пикселей контура до полигона. Решение, близкое к оптимальному, ищется с помощью алгоритма Колесникова [136]. Регулировка величин отклонений позволяет искать компромисс между размером PDF файла, который зависит от количества вершин полигонов, и качеством аппроксимации контуров. Потенциально можно выполнять аппроксимацию не полигонами, а сегментами кубической кривой Безье, но это нецелесообразно, т.к. потребует существенно больших вычислительных затрат, при этом часть кривых редуцируется в отрезки прямых, например при аппроксимации острых углов контуров, а хранение таких редуцированных кривых ведёт к увеличению размера файла.

На шаге 6 выполняется преобразование полигонов в пути, состоящие из отрезков прямых и кубических кривых Безье [53]: для каждой вершины полигона, прилегающие стороны которой образуют тупой угол, выполняется аппроксимация кубической кривой Безье двух отрезков от середин прилегающих сторон до данной вершины; если же прилегающие стороны образуют острый угол, то в путь добавляются отрезки от середин прилегающих сторон до данной вершины.

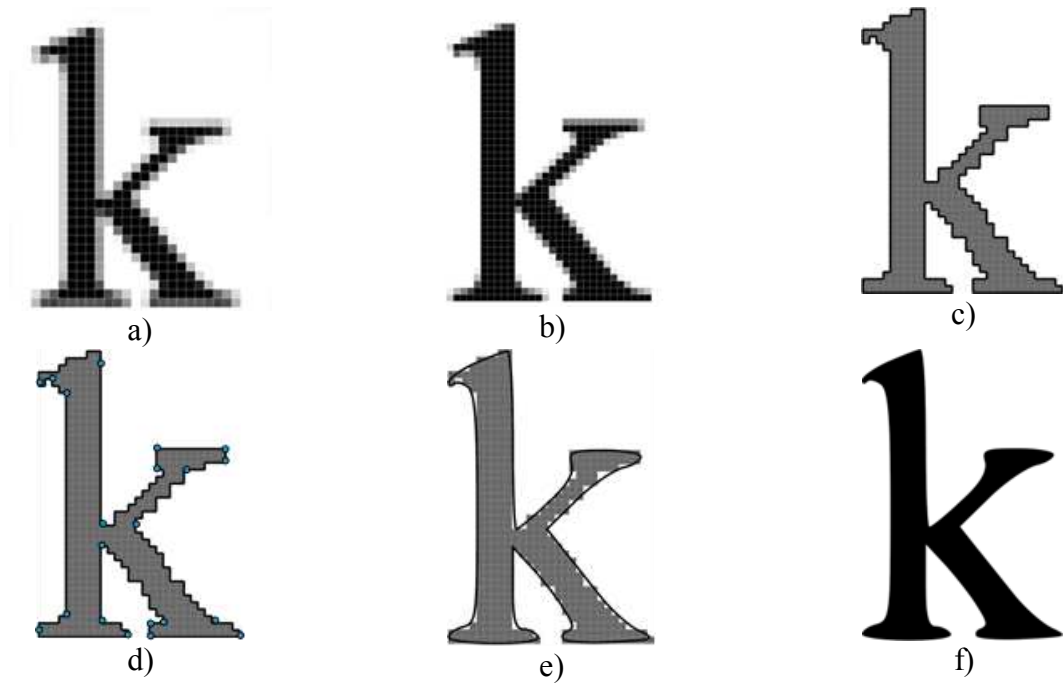


Рисунок 3.22 – Иллюстрация шагов обработки изображения символа: а) фрагмент сканированного изображения; б) масштабированный фрагмент; в) нахождение пикселей контура; г) аппроксимация контура полигоном; д) преобразование полигонов в пути; е) путь, залитый цветом символа.

Рисунок 3.22 иллюстрирует шаги обработки при векторизации контура буквы «к». Данный способ векторизации контуров символов также применен в ПО для печати снимков экрана [143].

### Результаты

Рассмотрим, как размер PDF файла зависит от способа сохранения информации. В таблице 3.11 приведены размеры PDF файлов при сходном визуальном качестве напечатанного текста для двух документов, показанных на рисунке 3.23. При сохранении текстовых областей и картинок в виде изображений, сжатых без потерь информации (сжатие deflate) размер файлов большой. При сжатии с потерей информации (JPEG) для обеспечения приемлемого качества изображения текста не может быть выбрана высокая степень сжатия, и размеры файлов значительно превышают заданное ограничение в 700 kb. При векторизации контуров символов текста размер файлов удовлетворяет заданному ограничению.



Рисунок 3.23 – Документы для сравнения размера файлов PDF: а) только текст; б) текст и картинки.

Сравним визуальное качество векторизации контуров символов предлагаемым способом и Canon Trace&Smooth PDF [132]. На рисунке 3.24 показаны увеличенные примерно в 10 раз фрагменты страницы PDF для текста размером 6 пунктов. Рисунок 3.24а демонстрирует фрагмент сканированного изображения. Аналогично будет выглядеть такой фрагмент при сохранении в PDF изображения со сжатием без потери информации. Использование функции Canon Trace&Smooth PDF создаёт символы с угловатым неестественным контуром (рисунок 3.24б). Внешний вид контуров символов, полученных предлагаемым способом, гораздо лучше (рисунок 3.24с). Можно заметить, что форма некоторых букв немного искажается, например пропадает засечка (serif) у «b». Заметим, что это не является ограничением предложенного алгоритма векторизации, а есть следствие установки на шаге 5 (рисунок 3.20) таких параметров (максимального и суммарного отклонения евклидова расстояния от пикселей контура до полигона), которые обеспечивают размер PDF файла в заданном диапазоне. Уменьшение величин отклонений позволяет получить контур, повторяющий форму символов с незаметными невооруженным глазом отличиями.

Таблица 3.11. Размер PDF файла для различных способов сохранения при сходном визуальном качестве напечатанного текста

Способ сохранения	Размер файла, kb	
	Только текст	Текст и картинки
Растровое изображение, сжатое без потери информации	3882	11027
Растровое изображение, сжатое с потерей информации	1257	1490
Предлагаемый	<b>538</b>	<b>625</b>

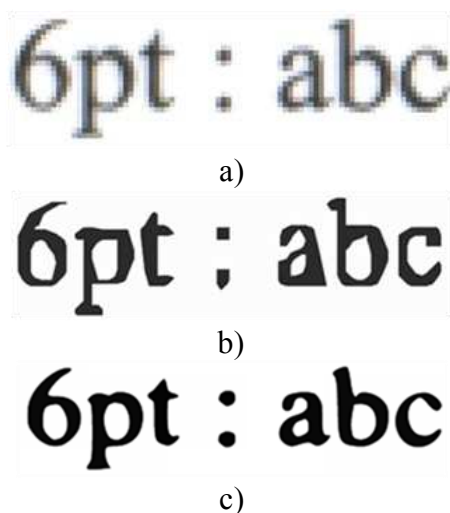


Рисунок 3.24 – Сравнение векторизации контуров символов: а) фрагмент сканированного изображения; б) Canon Trace&Smooth PDF; в) предлагаемый способ.

Оценка цвета символов и заливка всего тела символа одинаковым цветом позволяют избежать цветовых искажений. Во-первых, цветной и серый текст в силу использования печатного раstra на сканированном изображении является смесью пикселей разных цветов и оттенков. В результате применения способа подавления печатного раstra (этап 9 на рисунок 3.2) пиксели символов принимают близкие значения, но возможны небольшие вариации на краях символов. Во-вторых, даже при сохранении чёрного текста в виде цветного изображения с JPEG сжатием на краях символов образуются цветовые искажения. На рисунках 3.25а и 3.25с

показаны увеличенные цветной и чёрный символы, сохраненные в PDF с JPEG сжатием. Неравномерный цвет по телу символа при повторной печати ведёт к ухудшению его читаемости. Сохранение символов предложенным способом (рисунки 3.25b и 3.25d) позволяет избежать искажений цвета.

Важным показателем качества работы данного режима сканирования является точность распознавания символов для повторных копий документа, когда документ сканируется, затем преобразуется в PDF, снова печатается и так несколько раз. В таблице 3.12 приведено сравнение по количеству ошибок распознавания для предлагаемого способа и для сохранения в PDF растрового изображения, сжатого с помощью JPEG. Размеры файлов приблизительно равны. Исходный документ взят из тестовых изображений ИСО стандарта [29] и содержит примерно 1500 символов латинского алфавита, арабских цифр и знаков пунктуации, размер шрифта 10 и 12 пунктов. Оптическое распознавание символов выполнялось ПО ReadIRIS Pro 11 [144] с отключенным режимом коррекции синтаксиса. Сканирование осуществлялось с разрешением 300 dpi, PDF в цвете печатался с разрешением 600 dpi с помощью МФП Samsung CLX-6240.

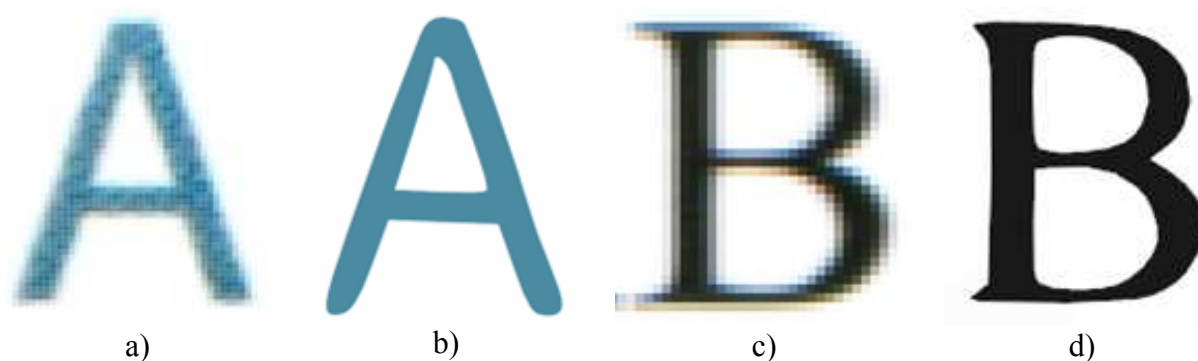


Рисунок 3.25 – Цвет символов в PDF в зависимости от способа сохранения: а) цветной символ, сжатый JPEG; б) цветной символ в виде залитого цветом контура; в) чёрный символ, сжатый JPEG; г) чёрный символ в виде залитого цветом контура.



Таблица 3.12. Отношение суммы ошибочно распознанных и нераспознанных символов к общему количеству символов

Способ сохранения	Скан оригинала	1-я копия	2-я копия	3-я копия
Изображение, сжатое JPEG	0,002	0,007	0,104	0,251
Предлагаемый		<b>0,002</b>	<b>0,015</b>	<b>0,084</b>

В результате распознавания скана оригинала возникло несколько не критичных для визуального восприятия ошибок, например, запятая заменилась на точку, а цифра “0” на букву “О”. При распознавании первой копии с использованием предлагаемого способа эти же ошибки повторились, что ожидаемо. Для изображения, сжимаемого JPEG, появилось несколько новых ошибок, связанных, в основном, с пропуском или заменой символов пунктуации. В результате распознавания второй копии изображения, сжимаемого JPEG, количество ошибок распознавания резко возрастает, причем много ошибок связано с неверным распознаванием букв, например, “i” распознается как “l”. Количество ошибок для предлагаемого способа значительно меньше, и резкий рост их числа начинается только с третьей копии.

### 3.7.2. Сканирование и копирование двухсторонних карточек

В ряде МФП компании Самсунг, например в модели CLX-6240 [145], реализован режим копирования «ID сору», который позволяет скопировать две стороны идентификационной карточки (identification card, ID card), например водительских прав, на одну страницу. В ИСО стандарте 7810 [146] приведены размеры типовых ID карточек. На страницу A4 помещаются обе стороны ID карточки. В данном режиме пользователь помещает карточку на стекло сканера лицевой стороной и сканирует, затем переворачивает карточку на стекле и сканирует её обратную сторону, изображения обеих сторон карточки печатаются

на одной странице. Эксплуатация режима «ID сору» выявила следующие недостатки:

- отсутствует коррекция скоса изображений, в результате копия выглядит неаккуратно;
- если пользователь неверно ориентировал одну из сторон карточки, например, перевернул вверх ногами, то на напечатанной странице стороны карточки ориентированы различным образом, что неудобно;
- невозможно сохранить PDF со сформированной страницей.

Задачей НИР являлась разработка режима сканирования и копирования двухсторонних карточек, в котором будут устранены перечисленные недостатки, при этом время копирования увеличится не более чем на 1 с по сравнению с существующим режимом на идентичной встроенной платформе. В качестве объектов для сканирования могут выступать не только идентификационные, но и любые двухсторонние карты, содержащие цифробуквенную информацию, например, банковские и дисконтные карты. Рисунок 3.26 иллюстрирует работу режима сканирования двухсторонних карточек.

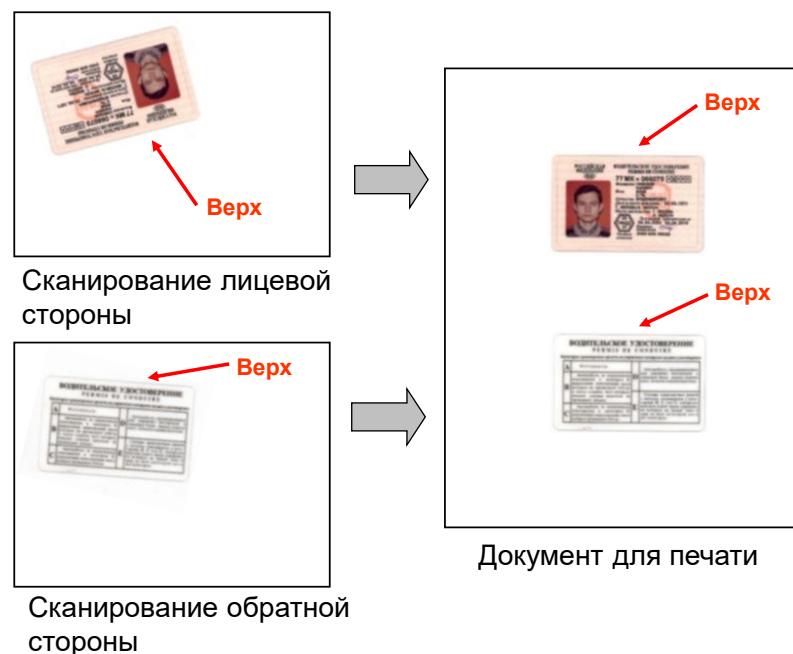


Рисунок 3.26 – Иллюстрация работы режима сканирования двухсторонних карточек.

### **Этапы методологии НИР при разработке режима копирования двухсторонних карточек**

В таблице 3.13 приведены основное содержание и результаты этапов НИР по разработке данного режима в соответствии с предложенной во второй главе методологией. Использование методологии и методики режимов сканирования позволило формализовать разработку режима сканирования двухсторонних карточек и выполнить её в кратчайшие сроки.

Таблица 3.13. Этапы НИР по разработке способа копирования двухсторонних карточек в соответствии с предложенной методологией

Этап НИР	Содержание
1. Поиск идеи	<p>Целесообразно с небольшими изменениями применить методику режимов сканирования для каждой из сторон карточки. Ключевой вопрос: как определять ориентацию изображения?</p> <p>Возможные подходы определения ориентации:</p> <p>а) детектирования лица, но это применимо только для одной из сторон для нескольких типов карточек, где есть фотографии;</p> <p>б) распознавание символов, но сложно реализовать во встроенном ПО МФП распознавание для большого количества языков;</p> <p>с) распознавание только арабских цифр, учитывая, что карточки, как правило, содержат цифровую информацию, данный подход является перспективным, несмотря на то, что для определения переворота на 180° можно использовать только цифры 1, 2, 3, 4, 5 и 7. Принято решение, воспользоваться подходом с).</p>
2. Макет	<p>Предложен пример интерфейса пользователя для выбора варианта размещения сторон карточки на странице (рисунок 3.27). Вручную повернуты сканированные изображения двух сторон карточки, размещены на странице (несколько вариантов размещения) и</p>

	сохранены в PDF. Документы напечатаны для оценки визуального качества.
3. Предварительная оценка пользователем	Визуальное качество напечатанного и сохраненного в PDF изображений высокое.
4. Анализ патентов	Патент [147] описывает различные режимы копирования, в том числе копирование двух страниц оригинала на одну страницу копии, для определения ориентации изображений с целью переворачивания неверно ориентированных применяется распознавание символов, однако данный патент не рассматривает коррекцию скоса и сохранения результата сканирования в метафайл.
5. Выбор критериев	Ориентация изображений сторон карточек для 4-х возможных ориентаций ( $0^\circ$ , $90^\circ$ , $180^\circ$ , $270^\circ$ ) должна определяться с точностью не менее 95%. Задано ограничение на время работы на встроенной платформе.
6. Обзор литературы	Существует большое количество публикаций, описывающих способы оптического распознавания символов, их глубокий обзор требует большего времени, чем отведено для НИР, поэтому анализ существующих подходов заимствован из книги [148].
7. Первый прототип	Алгоритм определения ориентации карточки, для которой уже скорректирован скос, реализован на MATLAB. Рассмотрены случаи поворота на $180^\circ$ и корректной ориентации ( $0^\circ$ ).
8. Оценка качества	Обнаружено, что детектируются только тёмные цифры на светлом фоне, а светлые на тёмном фоне не детектируются. Потребовалась доработка алгоритма и прототипа.
9. Идентификация этапов в конвейере обработки	Предпочтительна реализация во встроенном ПО. Альтернативный вариант – TWAIN драйвер (см. рисунок 1.3).
10. Второй прототип	Реализован на C++ для всех этапов обработки. Рассмотрены случаи поворота на $90^\circ$ , $180^\circ$ , $270^\circ$ и корректной ориентации ( $0^\circ$ ).
11. Анализ лицензий используемого ПО	Для сжатия в PDF используется библиотека libjpeg [141].

12. Оценка качества и производительности	Для 75 изображений карточек (на каждом из которых изображена, как минимум, одна цифра из набора 1, 2, 3, 4, 5, 7), ориентированных всеми четырьмя возможными способами, имело место только 4 случая неверного определения ориентации. Время обработки одной стороны карточки при сканировании с разрешением 300 dpi на встроенной платформе менее 1 с. Учитывая то, что изображение лицевой стороны карточки обрабатывается во время сканирования обратной стороны, требования по производительности удовлетворены.
13. Защита интеллектуальной собственности	Подготовлена заявка на изобретение, по которой был выдан патент в РФ [70] и патенты в США [71, 72].
14. Анализ требований к описанию	Код на C++ оформлен в соответствии с требованиями производственных подразделений. Проведено автоматическое сканирование статическим анализатором кода для поиска ошибок и уязвимостей.
15. Документация и тесты	Подготовлены отчёт по НИР, рекомендации по параллельной реализации алгоритма на встроенной платформе, набор тестовых изображений.

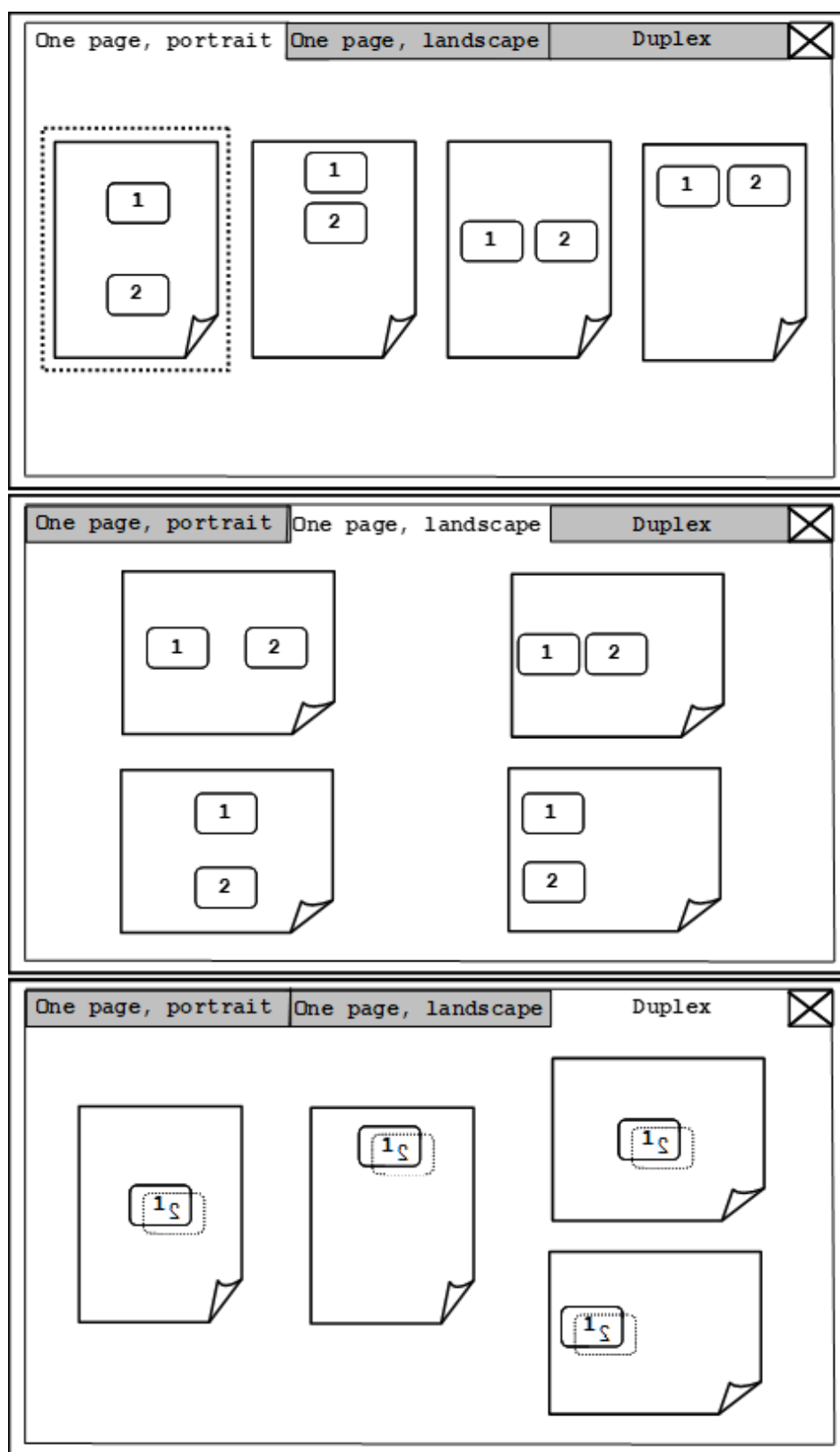


Рисунок 3.27 – Пример интерфейса пользователя для выбора варианта размещения сторон карточки на странице.

### Последовательность этапов режима копирования двусторонних карточек



Рисунок 3.28 – Схема этапов обработки в режиме копирования двухсторонних карточек.

На основе методики режимов сканирования предложена последовательность этапов обработки, показанная на рисунке 3.28. Левый столбец (этапы с 1 по 9) отражает обработку изображения лицевой стороны карточки, правый (этапы с 10 по 18) – обратной стороны. На этапе 19 правильно повернутые изображения обеих сторон карточки размещаются на странице заданным образом и сохраняются в метафайл. Этапы, рекомендованные для режимов сканирования (см. рисунок 3.2), показаны на голубом фоне. Задача детектирования текстовых областей не ставится, поэтому классификация на области текста и картинок не производится, выполняется классификация только фоновых областей.

### **Определения ориентации изображения стороны карточки**

Отличительной особенностью данного режима является этап определения ориентации изображения стороны карточки (9 и 18 на рисунке 3.28) и поворот, если необходимо исправить ориентацию. Шаги данного этапа приведены на 3.29. Как указано в таблице 3.13, ориентация изображения определяется путём распознавания арабских цифр “1”, “2”, “3”, “4”, “5” и “7” для четырёх возможных ориентаций: 0°, 90°, 180°, 270°. Часто на карточках цифры располагаются на сложном цветном фоне (см. пример на рисунке 3.32), поэтому применение для их сегментации отсечения по глобальному и локальному порогу яркости неэффективно. Целесообразно определять границы символов с помощью порогового отсечения результата фильтра высоких частот (ФВЧ). На шаге 1 (рисунок 3.29) цветное сканированное изображение конвертируется в полутоновое  $I_I$  с помощью формулы 1.1. На шаге 2 изображение  $I_I$  фильтруется с помощью ФВЧ. При разрешении сканирования 300 dpi используется ядро свёртки, показанное в формуле (3.64).

$$I_f = I_I * \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -8 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{bmatrix}. \quad (3.64)$$



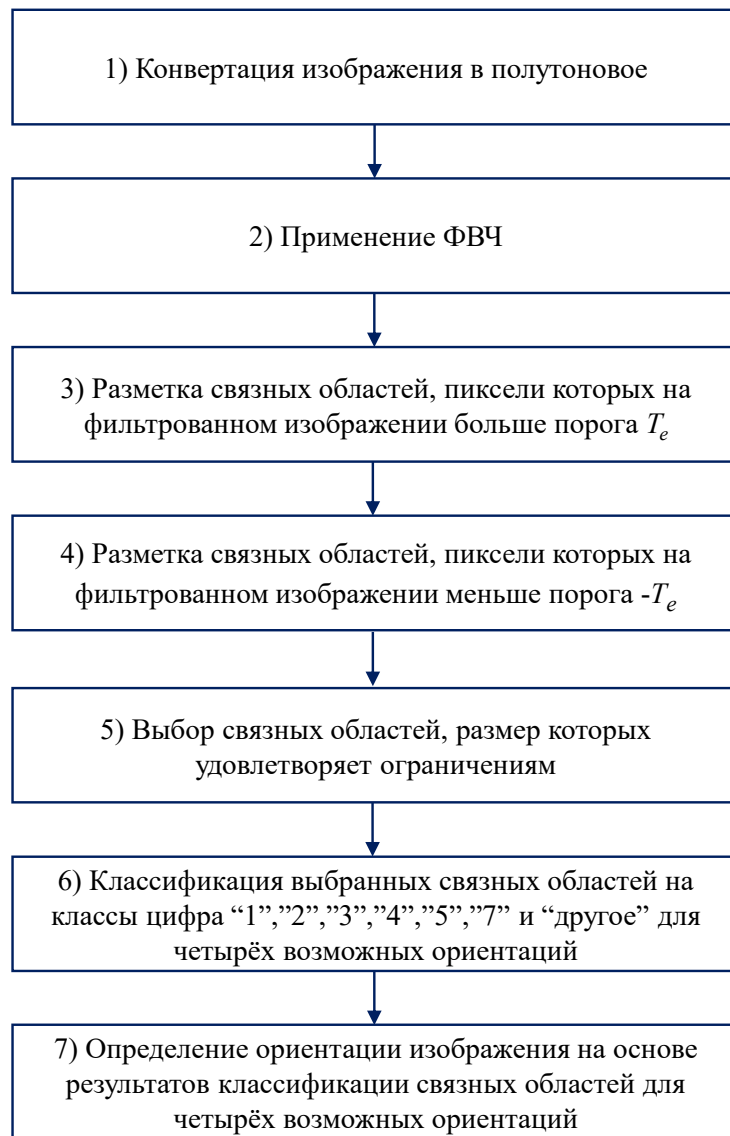


Рисунок 3.29 – Шаги способа определения ориентации изображения стороны карточки.

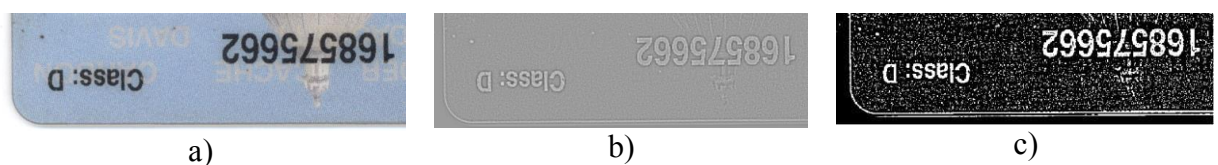


Рисунок 3.30 – Пример обработки фрагмента стороны карточки: а) исходное изображение; б) абсолютное значение результата ФВЧ; с) пороговое отсечение результата ФВЧ на шаге 3.

Разметка связных областей, пиксели которых на фильтрованном изображении больше порога  $T_e$ , выполняется на шаге 3. Данные пиксели расположены в районе границ тёмных символов на светлом фоне. Разметка связных

областей, пиксели которых на отфильтрованном изображении меньше порога  $-T_e$  выполняется на шаге 4. Данные пиксели расположены в районе границ светлых символов на тёмном фоне. На рисунке 3.30 показан пример обработки фрагмента стороны карточки. В результате порогового отсечения на шагах 3 и 4 получаются связные области, только немногие из которых соответствуют цифрам. Связные области, соответствующие цифрам, как правило, имеют множество отверстий и неравномерностей внешнего контура (см. рисунок 3.30с).

Обозначим бинарное изображение, содержащее одну из связных областей полученных на шаге 3 или 4, как  $I_\Omega$ . Количество столбцов в  $I_\Omega$  обозначим  $W_\Omega$ , а количество строк –  $H_\Omega$ . Большинство связных областей, не являющихся цифрами, можно отбросить на основе их размера. Связная область выбирается для дальнейшего анализа, если её размеры удовлетворяют следующему условию:

$$H_\Omega \geq 14 \text{ AND } H_\Omega \leq 70 \text{ AND } W_\Omega \geq 14 \text{ AND } W_\Omega \leq 70 \text{ AND } \left( \frac{H_\Omega}{W_\Omega} > 1,2 \text{ OR } \frac{W_\Omega}{H_\Omega} > 1,2 \right). \quad (3.65)$$

Данное условие проверяется на шаге 5. Оно получено исходя из размеров цифр от 6 до 20 пунктов при разрешении сканирования 300 dpi. Для разрешения 600 dpi значения 14 и 70 увеличиваются в 2 раза. При реализации во встроенном ПО шаги с 1 по 5 выполняются в один проход по сканированному цветному изображению без отведения буферов памяти размером с целое изображение для хранения результатов ФВЧ и пороговых отсечений, хранятся данные только нескольких строк.

На шаге 6 выполняется классификация выбранных связных областей на классы “1”, “2”, “3”, “4”, “5”, “7” и “другое” для четырёх возможных ориентаций. Для печати цифр на идентификационных и банковских карточках обычно используются легко-читаемые шрифты. Узкие, декоративные и имитирующие рукописный шрифты применяются редко. На начальном этапе разработки классификатора рассматривалось два подхода для создания классификаторов: дедуктивный на основе априорных знаний о начертании цифр и индуктивный с помощью

машинного обучения. Для обучения классификаторов, обладающих высокой обобщающей способностью, требуется либо сбор представительной обучающей выборки, либо разработка алгоритма генерации таких синтетических данных, которые идентичны реальным. И то и другое требует по меньшей мере нескольких недель. С другой стороны, внешний вид цифр даже после применения ФВЧ известен и может быть описан с помощью аналитически заданных правил. На разработку набора классификаторов с заданными правилами требуется несколько дней. Целесообразно начать с дедуктивного подхода, а в случае, если разработанные классификаторы не обеспечат высокое качество распознавания на имеющейся выборке, переключиться на подготовку представительной обучающей выборки и использование методов машинного обучения.

Для каждой из рассматриваемых цифр сформулированы признаки и правила классификации. Список признаков и правил для всех рассматриваемых цифр можно найти в [53]. В качестве примера рассмотрим признаки и правила для распознавания цифры “2”. Для связной области  $I_\Omega$  вычисляются 7 признаков  $F_i$ :

$$F_1 = \frac{H_\Omega}{W_\Omega}, \quad (3.66)$$

$$F_2 = \max \left( \sum_{r=H_\Omega-2}^{H_\Omega-1} \sum_{c=0}^{W_\Omega-1} I_\Omega(r, c), \sum_{r=H_\Omega-3}^{H_\Omega-2} \sum_{c=0}^{W_\Omega-1} I_\Omega(r, c) \right) / 2W_\Omega, \quad (3.67)$$

$$F_3 = \sum_{r=\frac{3H_\Omega-1}{5}}^{\frac{3H_\Omega-1}{4}-1} \sum_{c=0}^{r-\frac{3H_\Omega}{5}+1} I_\Omega(r, W_\Omega - c - 1), \quad (3.68)$$

$$F_4 = \sum_{r=0}^1 \sum_{c=0}^1 I_\Omega(r, c) + \sum_{r=0}^1 \sum_{c=W_\Omega-2}^{W_\Omega-1} I_\Omega(r, c), \quad (3.69)$$

$$F_5 = \max \left( \sum_{r=0}^1 \sum_{c=\frac{W_\Omega}{3}-1}^{\frac{2W_\Omega}{3}-1} I_\Omega(r, c), \sum_{r=1}^2 \sum_{c=\frac{W_\Omega}{3}-1}^{\frac{2W_\Omega}{3}-1} I_\Omega(r, c) \right) / 2 \left( \frac{W_\Omega}{3} + 1 \right), \quad (3.70)$$

$$F_6 = \max \left( \sum_{r=\frac{H_\Omega}{5}-1}^{\frac{2H_\Omega}{5}-1} \sum_{c=W_\Omega-2}^{W_\Omega-1} I_\Omega(r, c), \sum_{r=\frac{H_\Omega}{5}-1}^{\frac{2H_\Omega}{5}-1} \sum_{c=W_\Omega-3}^{W_\Omega-2} I_\Omega(r, c) \right) / 2 \left( \frac{H_\Omega}{5} + 1 \right), \quad (3.71)$$

$$F_7 = \sum_{r=\frac{H_\Omega}{2}-1}^{\frac{2H_\Omega}{3}-1} \sum_{c=0}^{\frac{2H_\Omega}{3}-1-r} I_\Omega(r, c). \quad (3.72)$$

Признак  $F_1$  – отношение высоты  $I_\Omega$  к ширине. Остальные  $F_i$  используются для определения наличия или отсутствия пикселей связной области на определённых участках изображения  $I_\Omega$  (см. рисунок 3.31).

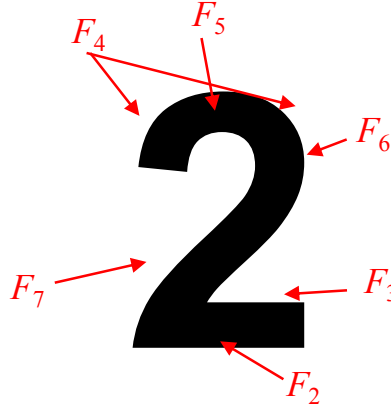


Рисунок 3.31 – Участки изображения, в которых вычисляются признаки  $F_i$ .

Если следующее логическое правило истинно, то область  $I_\Omega$  считается цифрой “2”:

$$F_1 \geq 1,1 \text{ AND } F_1 \leq 2,2 \text{ AND } F_2 > 0,65 \text{ AND } F_3 = 0 \text{ AND } F_4 = 0 \text{ AND } F_5 > 0,65 \text{ AND } F_6 > 0,55 \text{ AND } F_7 = 0. \quad (3.73)$$

Для каждой связной области применяется шесть классификаторов для рассматриваемых цифр:  $k \in \{1,2,3,4,5,7\}$ . Каждая связная область рассматривается

для четырёх ориентаций:  $i \in \{0, 90, 180, 270\}$ . Вычисляется  $N_{ki}$  – количество распознанных цифр  $k$  для ориентации  $i$ . На шаге 7 (рисунок 3.29) выполняется определение ориентации:

$$o = \underset{i \in \{0, 90, 180, 270\}}{\operatorname{argmax}} \left( \sum_{k \in \{1, 2, 3, 4, 5, 7\}} w_k N_{ki} \sum_{k \in \{1, 2, 3, 4, 5, 7\}} \operatorname{sign}(N_{ki}) \right), \quad (3.74)$$

где  $w_k$  – веса, причем веса для цифр “1” и “7” в два раза ниже, чем для остальных, так как для них вероятность ложного обнаружения выше. Второй множитель в формуле (3.74) учитывает то, что на определение ориентации более сильное влияние оказывает распознавание нескольких разных цифр, чем одинаковых.

### Результаты

Пример коррекции скоса и переворота изображения лицевой стороны карточки показан на рисунке 3.32.

Для оценки качества работы способа была сформирована выборка из 75 изображений различным образом ориентированных и скошенных сторон карточек. Каждое изображение содержало как минимум одну из шести рассматриваемых цифр. Всего на карточках было 1137 цифр из числа шести рассматриваемых. После шага 5 (рисунок 3.29) общее число связных областей составляет около 50000. В результате порогового отсечения результата ФВЧ 147 цифр распались на несколько связных областей и не могут быть классифицированы верно. Это происходит для цифр на неравномерном цветном фоне. Например, на рисунке 3.32 несколько цифр тяжело распознать даже человеку. Из оставшихся для классификации 990 областей цифр 982 были классифицированы верно, 8 отнесены к классу “другие”. Произошло несколько ложных срабатываний. Три раза буква “з” классифицировалась как цифра “3”, что положительно влияет на определение ориентации. Одиннадцать раз области от рисунков классифицированы как “1” и четыре раза как “7”. Только для одного изображения это привело к неверному определению ориентации. По итогам данного тестирования принято решение, что

разработанные дедуктивным (“сверху-вниз”) подходом классификаторы работают достаточно хорошо, и нет необходимости тратить время на создание альтернативного варианта с помощью машинного обучения (подход “снизу-вверх”).



Рисунок 3.32 – Пример коррекции скоса и переворота изображения лицевой стороны карточки.

Работа созданного режима сравнивалась с ПО OCR ReadIRIS Pro 11 [144] и FineReader 10 [149], которые имеют функции автоматической коррекции скоса и поворота изображения, повернутого на 180°. Таблица 3.14 содержит результаты сравнения. Предложенный способ позволил в 99% случаях корректно определить ориентацию карточек, что существенно больше, чем обеспечили программы OCR, хотя они для определения ориентации распознавали не только цифры, но и буквы.

Таблица 3.14. Результаты распознавания цифр и определения ориентации для повернутых на 180° скошенных карточек

ПО	Распознано цифр, %	Определена ориентация карточек, %
ReadIRIS Pro 11	74	72
Fine Reader 10	63	69
Предлагаемый способ	<b>86</b>	<b>99</b>

### **Выводы по третьей главе**

Представлена систематизация искажений изображений, возникающих при копировании и сканировании с сохранением информации в метафайл. Выделены наиболее критичные дефекты сканированного изображения, устранять которые рекомендуется во встроенном ПО.

Предложена методика (совокупность методов и подходов) режимов сканирования, которая включает этапы устранения основных дефектов сканированных изображений и проведение сегментации, необходимой для сохранения изображения в метафайл в соответствии с концепцией модели смешанного растрового содержимого. Методика предполагает предварительную оценку параметров обработки и сегментации по изображению низкого разрешения, а затем обработку изображения с разрешением, заданным пользователем. Предложены способы оценки и коррекции скоса без использования дополнительной памяти, подавления печатного раstra, улучшения локального контраста, сегментации изображения на области фона, текста и картинок. Для перечисленных способов выбраны параметры обработки в зависимости от разрешения сканирования, таким образом от пользователя не требуется настройки каких-либо параметров. Методика рекомендует последовательность применения предложенных способов обработки сканированного изображения.

Приведены примеры применения методологии НИР при создании решений для МФП и методики режимов сканирования для создания таких новых функций как сканирование и сохранение документа в PDF с векторизацией контуров символов, сканирование и копирование двухсторонних карточек.

Режим сканирования и сохранения документа в PDF с векторизацией контуров символов обеспечивает два цикла повторной печати и сканирования без существенной потери качества распознавания текста, при сходном визуальном качестве режим создаёт файл наименьшего размера, по сравнению с иными,

используемыми в МФП, методами. Режим копирования двухсторонних карточек на одну страницу позволяет скорректировать скос и исправить ориентацию 99 % карточек, стороны которых развёрнуты относительно друг друга на 180 градусов.



## **Глава 4. Подход к реализации специальных режимов печати**

В данной главе рассматриваются специальные режимы печати для экономии расходных материалов и добавления в напечатанный документ защитных элементов. Как правило, подобные режимы предназначены для устройств, поставляемых на крупные предприятия и в государственные организации, при этом сервисное обслуживание и обновление ПО, как правило, являются частью контракта. Срок эксплуатации устройств для данных сегментов рынка превышает 10 лет. В ряде случаев возникает необходимость добавления новых функций в существующие системы сканирования и печати путём обновления системного и/или встроенного ПО без модификации аппаратной части.

При модификации встроенного ПО объём добавляемого кода ограничен размером доступного пространства флеш-памяти, и влияние нового кода на существующий должно быть минимизировано, чтобы не внести ошибки в имеющиеся функции и не ухудшить характеристики производительности. Требуется сформировать подход, т.е. совокупность способов и приёмов [3], который позволит в короткие сроки разработать и реализовывать специальные режимы печати путём модификации встроенного ПО с учётом имеющихся требований и ограничений.

### **4.1. Специальные режимы печати**

К специальным режимам печати можно отнести такие, которые необходимо включить с помощью интерфейса пользователя, и работа которых чем-либо отличается от обычной печати. Большинство существующих специальных режимов печати можно отнести к одной из двух групп: печать для экономии расходных материалов (economical printing) и добавление в напечатанный документ защитных элементов (security printing).

#### 4.1.1. Экономичная печать

Экономия расходных материалов подразумевает сокращение расхода бумаги и краски, то есть чернил струйных и тонера лазерных принтеров. Для уменьшения количества используемых листов бумаги существуют режимы печати на обеих сторонах листа (duplex) и печати нескольких уменьшенных страниц документа на листе (n-up). Для сокращения расхода краски различают режимы черновой (draft) и экономичной (eco) печати. В первом случае подразумевается, что печатается черновик для оценки расположения напечатанных элементов на странице и не предполагается пускать напечатанный документ в документооборот. Во втором случае немного снижается расход краски, при этом качество документа незначительно отличается от печати с высоким качеством, возможным для данного устройства. Зачастую режимы черновой и экономичной печати используют одинаковый способ подготовки изображений для печати, различаются лишь величина сокращения расхода краски. Иногда алгоритмы обработки для этих режимов кардинально различаются. Поскольку экономичные режимы не только удешевляют печать, но и ведут к уменьшению количества отходов, иногда позволяют сократить потребление электроэнергии и углеродные выбросы, такие режимы называют экологичными или режимами «зелёной» печати [150].

Одним из наиболее распространённых методов сокращения расхода краски является уменьшение оптической плотности печатаемых элементов, что соответствует повышению яркости, при этом чёрный текст становится серым [151, 152]. Такой способ позволяет достаточно точно регулировать величину экономии краски, но визуальное качество графики и текста ухудшается. Изображение становится менее контрастным, на краях символов и линий появляется видимая неравномерность в силу присутствия на границах не полностью заполненных краской ячеек полутонирования.

Для предотвращения искажений краёв существует несколько методов, которые делают светлее только внутренние области символов текста и графических элементов [153, 154]. В этом случае качество отпечатка существенно выше, хотя регулярная текстура, образуемая печатным растром во внутренних областях, хорошо заметна. Количество сэкономленной краски зависит от содержимого изображения. В [155] показано, что если до растривания выполнять увеличение яркости внутренних областей графических элементов не на константное значение для всех пикселей области, а прибавлять синий шум, то наблюдатели оценивают визуальное качество напечатанного изображения выше, чем для других перечисленных способов.

Поскольку часто большую часть документов занимает текст, то целесообразно уменьшать расход краски при печати символов шрифтов. Для этой цели было создано несколько, так называемых экономичных шрифтов, например EcoFont [156] и Ryman Eco [157]. Пример текста для шрифта EcoFont показан на рисунке 4.1. В данном шрифте внутри символов располагаются круглые области, которые при печати не заполняются краской.



Рисунок 4.1 – Пример текста для шрифта EcoFont Vera Sans.

Авторы EcoFont утверждают [156], что применение экономичных шрифтов в документах позволяет уменьшить количество используемого тонера почти на 50%. Однако доля экономии краски сильно зависит от размера и семейства (гарнитуры) шрифта. Исследование [158] показывает, что применение экономических шрифтов экономит от 10 до 30% тонера для наиболее распространённых размеров текста (от 12 до 14 пунктов). К недостаткам экономии краски с помощью использования экономичных шрифтов следует отнести то, что они существуют только для

ограниченного набора языков, ухудшается качество OCR текста, напечатанного с использованием таких шрифтов.

#### **4.1.2. Печать защитных элементов**

Большое количество полиграфических технологий предназначено для защиты банкнот, ценных бумаг и документов, удостоверяющих личность, права собственности и т.п. от подделки и фальсификации [159]. Офисные принтеры не способны печатать защитные элементы столь же высокого качества как специализированные полиграфические устройства. Однако ряд защитных элементов можно создать и на офисной технике, что применяется для предотвращения копирования и несанкционированного распространения напечатанных документов. Например, при типографской печати ценных бумаг типичная высота символов микротекста от 15 до 90 мкм [160]. При печати на принтере высота символов микротекста составляет около одного типографского пункта (1/72 дюйма, около 350 мкм) [171]. Тем не менее, даже такой текст человек не может прочитать невооружённым глазом. При копировании с помощью офисной техники микротекст искажается, что позволяет использовать данный элемент для защиты от копирования. Разумеется, затратив значительные усилия или с помощью специализированных устройств, можно подделать защитные элементы, созданные на офисном принтере, но предполагается, что это не имеет смысла для защищаемого типа документов.

Для предотвращения несанкционированного распространения напечатанных документов в них встраивается так называемый цифровой водяной знак (ЦВЗ), который может быть виден или скрыт. Если ЦВЗ скрыт, то говорят о применении принтерной стеганографии [161], которая предполагает извлечение скрытого сообщения путём анализа сканированного изображения бумажного документа. Большинство описанных в литературе методов принтерной стеганографии для

работы с текстовыми документами основано на одном из следующих подходов для кодирования скрытой информации:

- сдвиг строк текста, слов или символов [162–164];
- модификация формы или цвета символов [165–167];
- добавление меток или изменение вида ячеек полутонирования в фоновых областях [168–170].

Определяющий вклад в разработку и практическую реализацию методов печати защитных элементов с помощью офисных лазерных принтеров внесли исследователи компании Xerox [171]. К наиболее известным способам относятся следующие:

1. Для английского языка, арабских цифр и символов пунктуации спроектирован специальный матричный шрифт, который позволяет печатать микротекст с высотой символов немного меньше одного типографского пункта. Символы хранятся в формате шрифта Type 3 Postscript [172]. На рисунке 4.2 показан пример напечатанного микротекста, сканированного с оптическим разрешением 2400 dpi.
2. Печать на равномерном фоне бинарного изображения или текста, которые не видно при просмотре документа под углом близким к 90 градусам, но видно при просмотре документа под острым углом. Такой эффект достигается за счёт использования в документе различных ячеек полутонирования (используются разные функции для ориентации пятна краски в ячейке) для фона и латентного текста или изображения [173].
3. Подобно предыдущему способу используется два вида ячеек полутонирования, дающих одинаковое распределение цвета, но точки внутри одной ячейки смещены относительно точек в другой, что при просмотре через полупрозрачную плёнку с напечатанной на ней сеткой позволяет увидеть латентное изображение по причине образования муарового узора [174].

4. Как правило, поглощение инфракрасного излучения чёрным и цветными тонерами различаются. Создаются две разные комбинации CMYK цветов, которые при обычном освещении воспринимаются человеком одинаково, но отличаются при освещении инфракрасным источником [175].
5. Печать в фоновых областях скрытого сообщения в виде шаблона из жёлтых точек, так как жёлтый наименее видимый на бумаге из CMYK цветов. Сообщение содержит информацию о серийном номере устройства, дате и времени печати. Обычно подобное сообщение называют идентификационным кодом устройства (Machine Identification Code, MIC) [176].

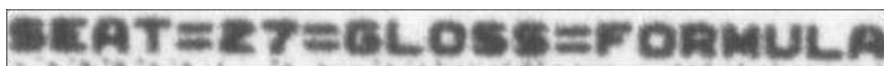


Рисунок 4.2 – Пример микротекста Хегох, высота символов около 250 мкм (1/100 дюйма).

Перечисленные существующие методы печати защитных элементов имеют ряд ограничений и недостатков. Методы, изменяющие вид ячейки полутонирования, требуют присутствия в документе достаточно больших по площади областей серого цвета. Печать микротекста возможна только на нескольких языках из специализированного системного ПО. Идентификационный код устройства часто невозможно корректно извлечь из копии или части документа.

#### **4.2. Локальная модификация кода процессора растровых изображений**

Обычно процессор растровых изображений (RIP) реализуется во встроенном ПО устройства печати, иногда в системном ПО ПК. При добавлении нового специального режима во встроенное ПО существующих устройств возникают следующие требования:

- а) Не должно быть влияния на время и результат работы других режимов печати.

- b) Время печати в специальном режиме не должно превышать время печати в обычном режиме на заметную для пользователя величину.
- c) Необходимо, насколько это возможно, минимизировать размер добавляемого программного кода.

Дабы удовлетворить этим требованиям, сформулирован следующий подход, при котором выполняется лишь локальная модификация кода RIP:

- определить в существующем коде такие несколько мест, где за счёт ветвления можно выполнить переход на процедуру реализации специального режима, а затем вернуться к основной последовательности обработки;
- выбрать вычислительно простой алгоритм, имеющий компактную реализацию для используемой встроенной платформы;
- при реализации переиспользовать существующие в коде RIP функции.

Высокоуровневая схема работы RIP показана на рисунке 4.3. На схеме опущено много важных деталей, например, преобразование цветов, преобразования систем координат, обрезка (clipping) и т.д. Тем не менее схема иллюстрирует основные этапы обработки и потоки данных.

На вход RIP поступает метафайл с графическими элементами для формирования изображения страницы. RIP выполняет три блока операций:

- Интерпретация формата метафайла, извлечение графических и управляющих записей.
- Отрисовка (рендеринг) графических элементов в координатах печатаемого изображения.
- Преобразование отрисованного изображения в печатный растр.

На рисунке 4.3 цифрами показаны три модуля, в которых целесообразно выполнить передачу управления на функции новых режимов:

1. Модуль растеризации шрифтов, где можно изменять отображение отдельных символов текста.

2. Модуль декодирования последовательности пикселей растровых изображений, на выходе которого можно изменять вид растровых изображений, переданных в метафайле.
3. На выходе из блока преобразования в печатный растр можно вносить изменения в цветовые каналы подготовленного для печати изображения, например для цветного лазерного принтера это бинарные каналы СМΥК.



Рисунок 4.3 – Схема работы RIP с указанием модулей для передачи управления в специальные режимы печати.



Без ограничения общности предложенный подход применялся при создании новых специальных режимов печати для лазерного принтера с RIP, предназначенном для печати метафайла Postscript [21]. В результате локальной модификации модуля растеризации символов реализованы режим защиты от копирования за счёт генерации “на лету” микросимволов и печати микротекста [177–180] и режим экономичной печати текста [181, 182]. Путём модификации модуля декодирования последовательности пикселей растровых изображений созданы режим черновой печати [183–188] и режим защиты от копирования с помощью генерации микроизображений [53]. Добавление обработки на выходе блока преобразования в печатный растр позволило реализовать два режима встраивания скрытых ЦВЗ в печатаемый документ [189–191].

#### **4.3. Примеры применения подхода к реализации режимов печати**

В данном разделе приведены четыре примера применения подхода к реализации новых специальных режимов печати за счёт локальной модификации кода процессора растровых изображений. Режимы печати микротекста и экономичной печати встраиваются в модуль растеризации шрифтов. В режиме черновой печати обработка выполняется одновременно на выходе из блока декодирования последовательности пикселей растровых изображений и в модуле растеризации шрифтов. Режим добавления в документ скрытых ЦВЗ встраивается на выходе из блока преобразования в печатный растр.

##### **4.3.1. Генерация микросимволов и печать микротекста**

К недостаткам спроектированных для микропечати микрошрифтов, таких как MicroText Specialty Imaging Fonts компании Xerox [172], относится то, что они существуют для ограниченного количества языков, и для встраивания этих шрифтов в документ требуется установка на ПК специального ПО. Для

пользователя предпочтительно иметь возможность печатать микротекст на любом языке и из произвольного ПО.

### **Способ генерации микросимволов «на лету»**

На первый взгляд представляется, что можно использовать в документе, созданном в любом прикладном ПО, например MS Word, текст с размером шрифта по вертикали (кеглем) равным одному типографскому пункту (~350 мкм), напечатать этот документ, и таким образом получить микротекст. Однако при печати текста такого размера различимы только отдельные символы некоторых шрифтов, например Arial. Для большинства семейств шрифтов [192], особенно декоративных и имитирующих рукописные, почти все символы становятся неразличимыми. Под различимостью подразумевается возможность человека распознать напечатанный на бумаге символ при использовании увеличительного стекла.

Предлагаемая локальная модификация модуля растеризации шрифтов позволяет генерировать микросимволы “на лету” [177–180]. На рисунке 4.4 показан фрагмент схемы обработки в модуле растеризации шрифтов в RIP Postscript [21]. Чтобы избежать повторной растеризации, т.е. трансформации символа в соответствии с размером шрифта и разрешением печати из векторного описания в карту бит (бинарное изображение символа), используется кэш шрифтов (font cache). Для сокращения объёма используемой оперативной памяти карты бит для больших шрифтов не сохраняются в font cache. В эту схему добавляется два блока, обведённых на рисунке 4.4 пунктирной линией: изменение параметров шрифта до рендеринга символа в карту бит и изменение карты бит после.

Если для печати поступает символ шрифта размера 2 пункта или менее, то в блоке изменения параметров шрифта название шрифта заменяется на Helvetica, а размер устанавливается равным 6 пунктам. Выбор Helvetica обусловлен тем, что это один из стандартных шрифтов Postscript, в этом шрифте существуют начертания (glyph) для многих языков, форма начертаний проста и не содержит

декоративных элементов, например засечек (serif). Размер шрифта в 6 пунктов позволяет выполнить рендеринг символа так, что он хорошо различим, при этом при растеризации применяется хинтинг [193] – набор правил для улучшения вида относительно маленьких (меньше 10 пунктов) букв. Кроме того, чем больше размер шрифта, тем большего размера получается карта бит, что ведёт к увеличению времени, необходимого для её обработки.

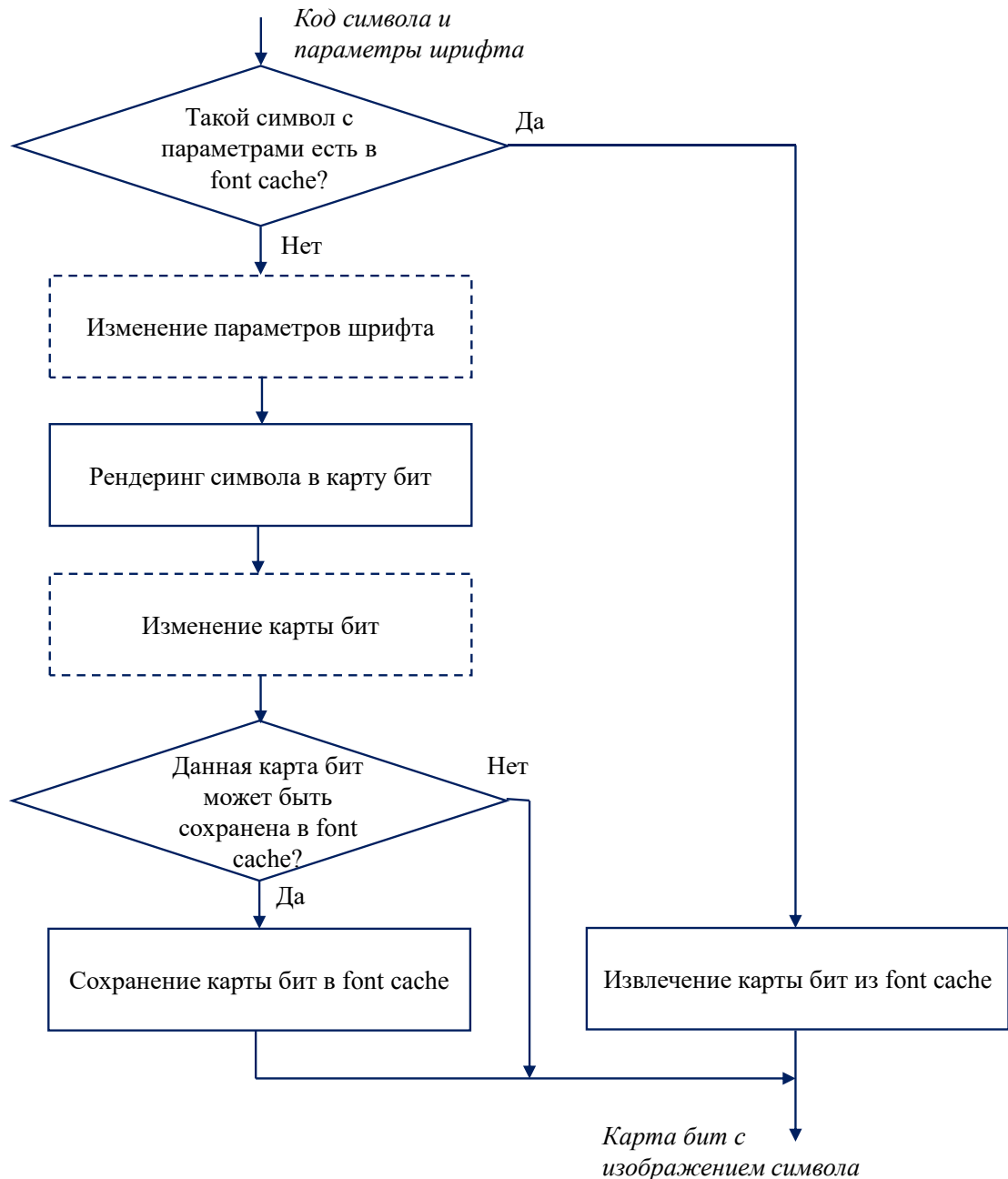


Рисунок 4.4 – Фрагмент схемы обработки в модуле растеризации шрифтов в RIP Postscript.

Далее существующий блок рендеринга из кода символа и векторного описания для шрифта Helvetica размера 6 пунктов создаёт карту бит с изображением символа. В карте бит единица соответствует печатаемой точке, а ноль – непечатаемой. Для изображения символа строится остов (скелет) с помощью операции утоньшения. По результатам анализа методов утоньшения, представленных в обзорах [194, 195], выбран алгоритм Ahmed–Ward [196], который даёт остовы, похожие на исходные символы, и реализуется с помощью компактного программного кода.

Изображение с остовом необходимо уменьшить до исходного размера шрифта. Для этого изображение разбивается на ячейки по числу пикселей целевого изображения. Если хотя бы один пиксель скелета попадает в ячейку, то соответствующий пиксель устанавливается в 1, в противном случае пиксель равен 0.

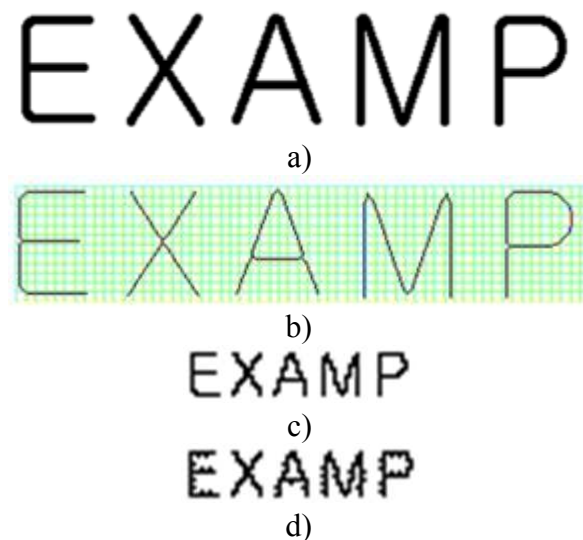


Рисунок 4.5 – Иллюстрация этапов генерации микросимволов: а) Растрированные символы для кегля 6; б) Остовы символов, показанные на сетке из ячеек; в) остовы, уменьшенные до целевого размера; г) итоговые битовые карты с микросимволами.

Получаемые таким образом микросимволы различимы при печати с помощью лазерного принтера. Однако для разрешения печати 1200 dpi и выше отрезки толщиной в один пиксель могут плохо пропечатываться из-за недостатка заряда в этом месте на фотобарабане. Это зависит от качества тонера и бумаги, износа фотобарабана и ещё ряда факторов. Если для отрезка из трёх подряд идущих по прямой пикселей сбоку от среднего пикселя “приклеить” ещё один пиксель (не важно, с какой стороны), то отрезки, как правило, пропечатываются полностью. Аналогично, если элемент символа «точка», например в букве «i», в результате утоньшения трансформируется единственный пиксель, то он, вероятно, не пропечатается. Поэтому к отдельно расположенному пикселю остова «приклеивается» ещё два так, чтобы эти три пикселя формировали между собой угол. Модифицированная карта бит сохраняется в кэше шрифтов.

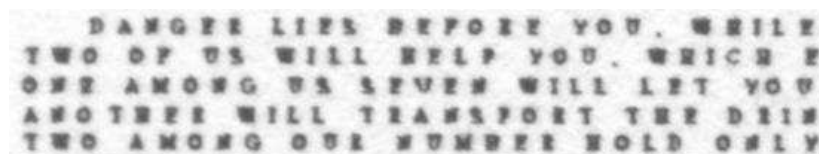
На рисунке 4.5 показан пример генерации микросимволов. Для растрированных символов размера шрифта 6 (рисунок 4.5a) выполняется построение остова (рисунок 4.5b). Зелёным на рисунке 4.5b показана сетка из 12 ячеек в высоту, что при разрешении 1200 dpi соответствует высоте микросимвола в 0.01 дюйма. Рисунок 4.5c отображает результаты уменьшения символов до 12 пикселей в высоту. Итоговые битовые карты с микросимволами, где к остову “приклеены” дополнительные пиксели показан на рисунке 4.5d.

### Результаты

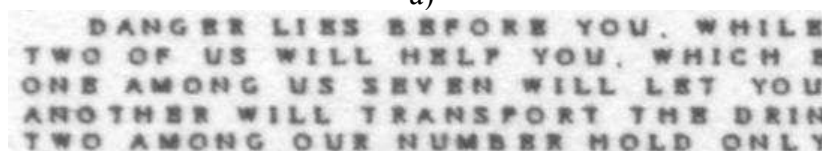
На рисунке 4.6 приведено два примера текста, напечатанного шрифтом кегля 1 с разрешением печати 1200 dpi. В первом случае использовался моноширинный шрифт Courier (рисунок 4.6a), символы которого имеют, пожалуй, наилучшую различимость для маленьких размеров шрифта. Второй пример напечатан предлагаемым способом с помощью генерации микросимволов «на лету» (рисунок 4.6b). Для этого и последующих в этой главе примеров оригинал сканирован с оптическим разрешением 2400 dpi. Многие символы, напечатанные шрифтом Courier, неразличимы, хотя текст можно прочесть за счёт угадывания целых слов.

Заметим, что для большинства других шрифтов различимость символов текста будет значительно хуже. При этом все символы, напечатанные предлагаемым способом, легко узнаваемы.

Печать микросимволов для 9 различных языков (русский, английский, немецкий, арабский, корейский, китайский, японский, иврит, греческий) показала, что предлагаемый способ позволяет получать на бумаге различимый с помощью увеличительного стекла микротекст для всех рассмотренных языков, кроме арабского. В арабской письменности используются лигатуры (символы пишутся слитно) [197]. Предлагаемый способ не гарантирует плавного перехода начертания одного символа в другой. Это ограничение можно преодолеть путём модификации алгоритма, но такая задача не решалась. Для японских и корейских иероглифов, а также символов корейского письма, тесты показали, что для уверенного чтения напечатанного предлагаемым способом микротекста следует увеличить высоту микросимволов до 450 мкм. Пример микротекста на корейском языке показан на рисунке 4.7.



a)



b)

Рисунок 4.6 – Текст, напечатанный шрифтом кегля 1: а) шрифт Courier; б) предлагаемый способ.



Рисунок 4.7 – Пример напечатанного микротекста на корейском языке.

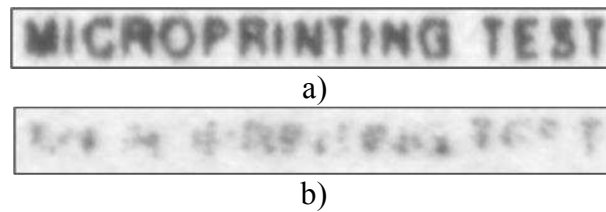


Рисунок 4.8 – Иллюстрация искажения микротекста при копировании: а) микротекст на оригинале; б) микротекст на копии.

Рисунок 4.8 демонстрирует искажения микротекста при копировании. На напечатанном оригинале тест различим (рисунок 4.8а), а на копии (рисунок 4.8б) полностью исказился. Печать и копирование выполнены с помощью МФП Samsung CLX-6240. Высота символов микротекста на рисунке 4.8а составляет 250 мкм, то есть равна высоте микротекста, напечатанного шрифтом Xerox MicroText Specialty Imaging Fonts (рисунок 4.2). Качество микротекста сравнимо, хотя шрифт Xerox специально спроектирован для микропечати, а в предлагаемом способе битовые карты с микросимволами генерируются «на лету».

Реализованный во встроенном ПО режим микропечати оказывает незначительное влияние на время печати по сравнению с обычным режимом, так как в модуль растеризации шрифтов добавлено всего несколько условных операторов. Кроме того, за счёт использования кэша шрифтов для хранения карты бит каждого кода символа, размера и названия шрифта растеризация выполняется только один раз. Добавление к странице текста (2000 букв английского алфавита, размер шрифта 12 пунктов) 100 символов микротекста увеличивает время печати менее чем на 0,01 с, что незаметно для пользователя.

#### **4.3.2. Способ экономичной печати**

Экономичный способ предполагает печать документов достаточно высокого качества, но с заметной экономии тонера по сравнению с обычным режимом. Как отмечено в разделе 4.1.1, обычно такой цели служит небольшое повышение

яркости внутренних областей символов, но при этом заметна текстура, образуемая печатным растром во внутренних областях. Предлагается уменьшить количество печатаемых точек во внутренних областях символов путём переключения элементов в карте бит растриванного символа, причём положение точек выбирается случайным образом, но зависит от ранее переключенных бит [181, 182]. Этот способ встраивается в модуль растеризации шрифтов (рисунок 4.4). Изменения параметров шрифта не происходит, изменяется только карта бит. Аналогично генерации микросимволов за счёт использования кэша шрифтов нет необходимости повторно обрабатывать карты бит тех символов и шрифтов, которые для этого задания на печать/метафайла встречались ранее.

### **Схема переключения бит растриванного символа**

Схема на рисунке 4.9 поясняет работу алгоритма переключения бит в случайных координатах. Первый прототип экономичного режима печати был итерационным. Количество итераций  $N$  устанавливается равным произведению количества строк  $H$  карты бит символа, количества столбцов  $W$  и коэффициента  $k$ ,  $k > 0$ . На каждой итерации генерируется псевдослучайное число в диапазоне от 0 до  $WH-1$  и из него тривиальным образом вычисляются координаты  $(r, c)$  в карте бит символа. Для того чтобы найти полностью заполненный краской участок в теле символа, фрагмент карты бит с центром в  $(r, c)$  сравнивается с матрицей  $B_1$  (корректнее говорить о семействе матриц, так как несколько элементов в ней могут быть как нулём, так и единицей). Если элементы карты бит с центром в  $(r, c)$  и  $B_1$  совпадают, то фрагмент битовой карты заменяется на матрицу  $B_2$ , в центре которой расположено несколько нулевых бит, что при печати ведёт к появлению крошечного «пятна», незаполненного тонером. Заметим, что необходимо обнулить несколько соседних бит, так как единичный нулевой бит, окружённый единицами, не ведёт к появлению «пятна». Далее выполняется переход к следующей итерации.

Матрицы  $B_1$  и  $B_2$  различаются в зависимости от разрешения печати [53]. Например, при разрешении 600 dpi используются матрицы:



$$B_1 = \begin{bmatrix} \times & 1 & 1 & 1 & \times \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ \times & 1 & 1 & 1 & \times \end{bmatrix}, \quad (4.1)$$

$$B_2 = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \quad (4.2)$$

где  $\times$  означает, что этот элемент может быть как 0, так и 1.

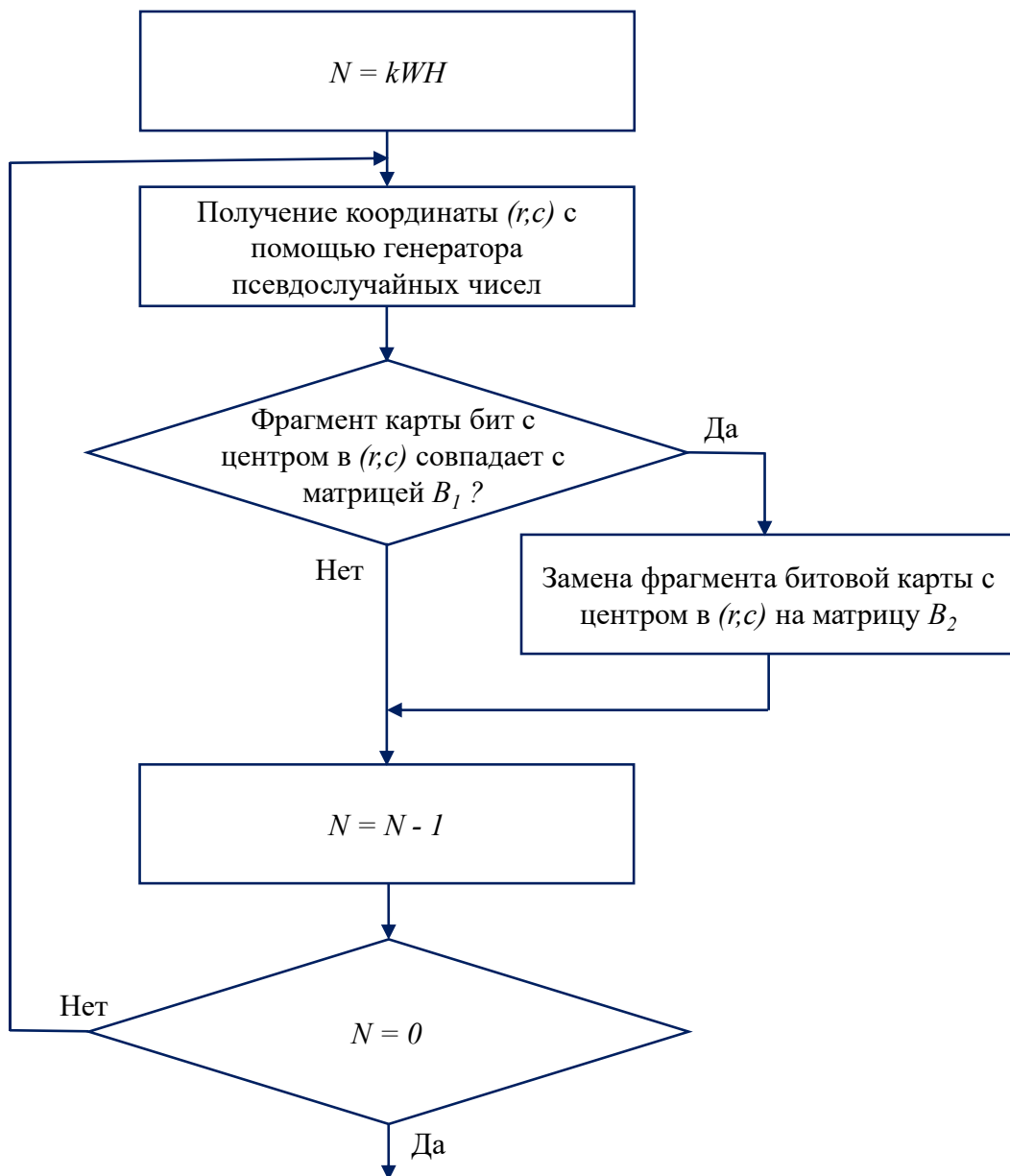


Рисунок 4.9 – Схема алгоритма переключения единичных бит в случайных координатах.

### Оценка экономии тонера

Качественно влияние параметра  $k$  на количество переключенных бит проиллюстрировано на рисунке 4.10. При  $k = 0,2$  переключенных бит немного, и экономия тонера не высока. При  $k = 1,4$  почти вся внутренняя область символа покрыта областями, которые не будут заполнены тонером, и экономия тонера существенно выше.



Рисунок 4.10 – Иллюстрация количества переключенных элементов в карте бит символа в зависимости от параметра  $k$ : а)  $k = 0,2$ ; б)  $k = 1,4$ .

Предварительная оценка экономии тонера выполнялась с помощью программного эмулятора путём подсчета количества печатаемых точек. Процент экономии тонера равен:

$$P_e = \frac{N_b - N_s}{N_b} 100\%, \quad (4.3)$$

где  $N_b$  – количество печатаемых точек для нормального режима печати,  $N_s$  – количество точек для экономичного режима.

Таблица 4.1 содержит величины процента экономии тонера в зависимости от значения параметра  $k$  для документов, используемых в стандарте ISO 24734 [29]. Использовалось константное значение порождающего элемента (seed) последовательности псевдослучайных чисел, поэтому процент экономии тонера для одинаковых  $k$  не изменяется от запуска к запуску. Заметим, что увеличение  $k$  больше 1,0 не приводит к существенной экономии тонера, поэтому дальнейшие исследования проводились для  $k=1,0$ .

Таблица 4.1 Процент экономии тонера  $P_e$  в зависимости от параметра  $k$ .

$k$	0,1	0,2	0,5	1,0	2,0
$P_e, \%$	8,6	11,8	14,3	15,8	16,0

Для более точной оценки величины экономии тонера проводился эксперимент, в котором с помощью взвешивания картриджа с тонером определялся вес краски, израсходованной при печати в обычном и предлагаемом режиме. Выполнялась печать ста копий документов из стандарта ISO 24734 [29] на МФП Samsung CLX-6240 [145]. При печати в экономичном режиме затрачено на 13,4 % меньше тонера.

За счет использования кэша шрифтов и эффективной реализации алгоритма на DSP время печати на Samsung CLX-6240 в экономичном режиме увеличилось лишь на 0,1% по сравнению с обычным режимом, что удовлетворяет требованиям к производительности МФП.

#### **Качество печати в экономичном режиме**

Для оценки качества напечатанного различными способами одинакового текста проводился опрос пользователей. Рассматривались: печать в нормальном режиме чёрного текста, печать предлагаемым способом, повышение яркости во внутренних областях символов, повышение яркости всего символа, использование шрифта Ecofont. При проведении эксперимента стремились обеспечить одинаковый процент экономии тонера  $P_e = 15,8 \%$  для предлагаемого способа и повышение яркости во внутренних областях символов. При повышении яркости всего символа  $P_e = 19\%$ . При использовании Ecofont  $P_e = 29\%$ . Сканированные с разрешением 600 dpi фрагменты текста для пяти способов печати показаны на рисунке 4.11.

В опросе участвовало 30 человек, которые были проинструктированы ранжировать пять напечатанных страниц в порядке убывания качества, где под качеством понималось удобство чтения текста. Интервьюируемые не знали

использованный режим печати. Самое качественное изображение получало оценку  $E$  в 5 баллов, следующее 4, и так далее до 1.

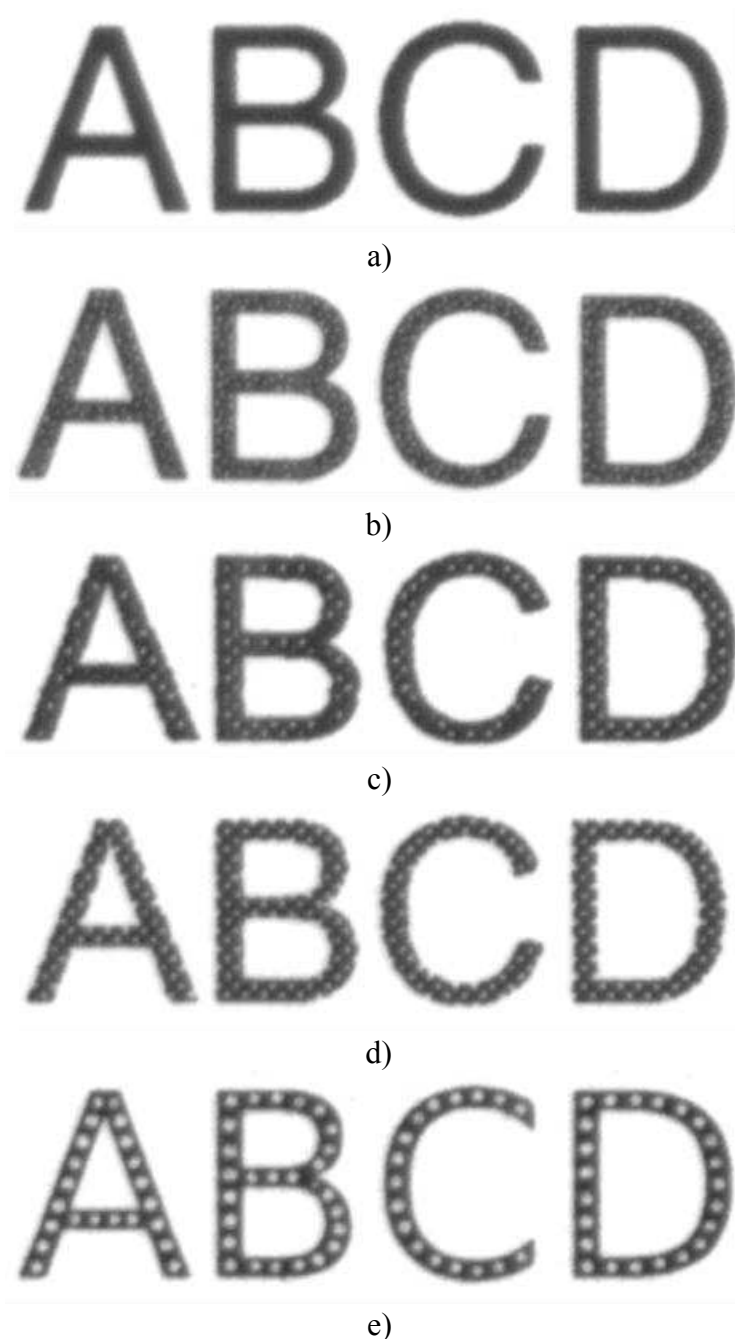


Рисунок 4.11 – Сканированный с разрешением 600 dpi фрагмент текста: а) сплошной чёрный цвет; б) предлагаемый метод,  $k = 1,0$ ; в) повышение яркости во внутренних областях символов; г) повышение яркости всего символа; д) использование шрифта Ecofont.

Таблица 4.2 содержит средние оценки пользователей  $\bar{E}$  для изображений с рисунка 4.11. Самую высокую оценку получил сплошной чёрный текст, напечатанный в нормальном режиме. Второе место занял предлагаемый способ, существенно опередив печать с повышением яркости внутри символов. Наблюдатели отмечали, что напечатанный предлагаемым способом текст легко читается, выглядит контрастным и резким. При печати на лазерных и светодиодных принтерах вокруг засвеченного на фотобарабане пятна возникает нежелательное рассеяние частичек тонера, причём чем больше точек печатается, тем больше частичек тонера рассеивается. В работах [198, 199] демонстрируется, что рассеивание тонера ведёт к падению чёткости границ для наблюдателя, а повышение яркости во внутренних областях графических элементов уменьшает количество краски, в том числе рассеиваемых частиц, и, таким образом, повышает чёткость напечатанного изображения. Данный эффект можно увидеть при внимательном сравнении увеличенных фрагментов сплошного чёрного текста (рисунок 4.11a) и добавления светлых «пятен» внутри символа (рисунок 4.11b). Тем не менее, за счёт более высокого контраста и равномерного заполнения символов краской, результат нормального режима печати оценивается выше. Печать с повышением яркости всего текста и с использованием шрифта Ecofont получили низкие оценки. В первом случае символы имеют рваные и нечёткие края. Шрифт с видимыми отверстиями в теле букв выглядит непривычно для пользователей.

Таблица 4.2 Средние оценки пользователей  $\bar{E}$  для изображений с рисунка 4.11.

Метод печати	$P_e$ , %	$\bar{E}$
Сплошной чёрный цвет	0	4,9
Повышение яркости во внутренних областях символов	14	2,6
Повышение яркости всего символа	19	2,2
Использование шрифта Ecofont	29	1,5
Предлагаемый метод	14	<b>3,8</b>

Предлагаемый способ позволяет экономить тонер не только при печати символов чёрного цвета, но и при печати серого и даже цветного текста (см. пример на рисунке 4.12).

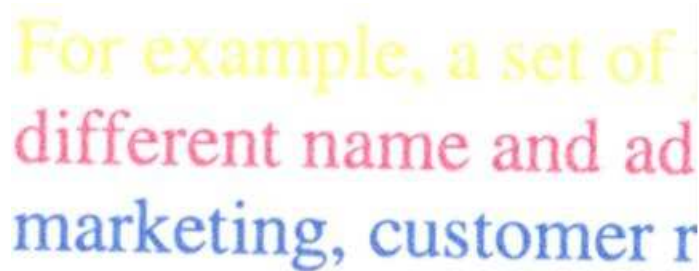


Рисунок 4.12 – Цветной текст, напечатанный в экономичном режиме.

### 4.3.3. Режим черновой печати

Документ, являющийся результатом черновой печати, служит для оценки расположения напечатанных элементов на странице. При этом все элементы должны быть легко узнаваемы, текст читаем, а экономия краски значительной по сравнению с нормальным режимом. Наиболее распространённым способом черновой печати является повышение яркости всего документа с помощью гамма-коррекции [78], что, как показано в предыдущем разделе, ведёт к искажению краёв символов, а иллюстрации становятся блёклыми и слабоконтрастными.

#### Обработка записей метафайла в зависимости от их типа

Предлагается реализовать режим черновой печати, в котором записи входного метафайла  $R_i \in \{S, G, I\}$  в зависимости от типа обрабатываются следующим образом [183–186]:

- текст  $S$  обрабатывается аналогично режиму экономичной печати с  $k = 1,0$ ;
- иллюстрации, передаваемые в виде растровых изображений  $I$ , преобразуются в цветные или полутоновые эскизы (sketches) в виде «карандашного рисунка»;
- при отрисовке последовательности команд векторной графики  $G$  на 30% повышается яркость.

Заметим, что ряд иллюстраций в документах существует в виде последовательности команд векторной графики. Такие картинки также могут быть конвертированы в эскизы, но в RIP это затруднительно сделать только за счёт локальной модификации кода. Кроме того, это требует существенных вычислительных ресурсов. Способ конвертирования картинок, созданных операторами векторной графики, в растровое изображение реализован в драйвере печати операционной системы [187, 188], что позволяет при печати с ПК в экономичном режиме все иллюстрации конвертировать в эскизы.

### **Способ преобразования растровых изображений в эскизы**

Рассмотрим способ преобразования растровых изображений в эскизы в RIP Postscript [21]. При интерпретации метафайла Postscript растровые изображения поступают на вход операторов языка Postscript `image` или `colorimage`. Как правило, изображение закодировано, например сжато, с помощью одного или нескольких так называемых фильтров. В коде обработки операторов `image` и `colorimage` выполняется вызов модуля декодирования последовательности пикселей растровых изображений. На выходе этого модуля растровое изображение *I* хранится как прямоугольная матрица пикселей и может быть преобразовано в эскиз.

Сформулированы следующие требования для метода преобразования иллюстрации в эскиз:

1. основная информация, содержащаяся в изображении, в том числе передаваемая цветом, на эскизе должна быть узнаваемой;
2. эскиз должен содержать как можно больше белых пикселей, что и обеспечивает существенное сокращение использования краски;
3. метод должен быть вычислительно прост и иметь реализацию в несколько килобайт (Кб) кода, чтобы была возможность встроить его в существующий RIP.

В литературе описано несколько десятков методов преобразования изображений в эскизы в виде «карандашного рисунка», которые можно разделить на три группы:

- использующие адаптивную бинаризацию [200–203];
- детектирующие контурные перепады [204, 205];
- основанные на «переносе стиля» с помощью глубоких нейронных сетей [206, 207].

Модели глубоких нейронных сетей занимают существенно больше нескольких Кб. Методы адаптивной бинаризации, как правило, дают эскизы с меньшим количеством белых пикселей, чем детектирующие контурные перепады. Исходя из анализа существующих алгоритмов, исследования сосредоточились на подходе, где сначала детектируются контурные перепады в канале яркости изображения с помощью порогового отсечения результата фильтра «разница гауссиан» (Difference of Gaussian, DOG) [78], затем цветовые каналы изображения маскируются картой контурных перепадов, т.е. умножаются на маску.

Пусть растровое изображение  $I$  представлено в цветовом пространстве RGB. Канал яркости  $I_I$  вычисляется по формуле (1.1).

Ядро свёртки DOG вычисляется как:

$$g(i, j) = \frac{1}{\sqrt{2\pi}} \left( \frac{1}{\sigma_1} e^{-\frac{i+j}{2\sigma_1^2}} - \frac{1}{\sigma_2} e^{-\frac{i+j}{2\sigma_2^2}} \right), \quad (4.4)$$

где  $i, j \in [-N_c/2, N_c/2]$ ;  $N_c$  – размер ядра свёртки;  $\sigma_1^2$  и  $\sigma_2^2$  – дисперсии гауссиан.

Маска  $M$  вычисляется путём порогового отсечения результата DOG фильтра:

$$M = \begin{cases} 1, & I_I * g > T_e, \\ 0, & \text{иначе,} \end{cases} \quad (4.5)$$

где  $T_e$  – предопределённый порог.

Цветовые каналы изображения  $I$  маскируются  $M$ :

$$I'_R(r, c) = M(r, c)I_R(r, c) + (1 - M(r, c))(2^{D_I} - 1), \quad (4.6)$$



$$I'_G(r, c) = M(r, c)I_G(r, c) + (1 - M(r, c))(2^{D_l} - 1), \quad (4.7)$$

$$I'_B(r, c) = M(r, c)I_B(r, c) + (1 - M(r, c))(2^{D_l} - 1), \quad (4.8)$$

где  $D_l$  – глубина цвета на канал.



a)



b)



c)



d)

Рисунок 4.13 – Примеры эскизов: а) исходное изображение; б) эскиз предлагаемым способом; в) эскиз путём адаптивной бинаризации [203]; д) эскиз с помощью последовательного рисования линий [204].

На рисунке 4.13 показаны исходное изображение и три эскиза: предлагаемым способом, с помощью адаптивной бинаризации [203] и итеративного детектирования границ [204]. Эскиз, полученный предлагаемым способом, позволяет увидеть все важные детали фотографии в цвете и содержит наибольшее количество белых пикселей, т.е. обеспечивает наилучшую экономию краски, по сравнению с альтернативными вариантами.

### Результаты

Заметим, что в Postscript цветное изображение может быть представлено в различных цветовых пространствах, например RGB, CMYK, HSB. В RIP Postscript существуют функции для конвертирования изображения из всех поддерживаемых цветовых пространств в одноканальное в градациях серого и процедуры маскирования каналов изображения. Эти функции переиспользуются при реализации режима черновой печати, что позволяет значительно сократить количество добавляемого в RIP кода. Временные затраты на конвертирование между цветовыми пространствами и маскирование обычно невелики, так как в RIP они выполняются с помощью DSP или ASIC. Фильтр DOG также реализуется на DSP, что позволяет обеспечить незначительное увеличение времени работы в режиме черновой печати по сравнению с нормальным режимом. Время печати документов из стандарта ISO 24734 [29] на МФП Samsung CLX-6240 [145] в черновом режиме увеличилось лишь на 0,5%.

Сравнение предлагаемого способа черновой печати и повышения яркости изображения с помощью гамма-коррекции показано на рисунке 4.14. В обоих случаях процент экономии тонера  $P_e$ , оцениваемый с помощью эмулятора по формуле (4.3) суммарно для четырёх цветовых каналов CMYK, равен 83 %. Оба способа позволяют легко узнать содержимое изображения, но цветной эскиз выглядит живо и необычно.



a)



b)



c)

Рисунок 4.14 – Сравнение способов черновой печати иллюстраций: а) исходное изображение; б) повышение яркости; с) замена изображения на его эскиз.

Среди 23 пользователей был проведён опрос. Им демонстрировалось десять троек напечатанных изображений как на рисунке 4.14: исходное изображение и два способа черновой печати картинок. Требовалось выбрать один из трёх вариантов ответа: «предпочитаю повышение яркости картинок», «предпочитаю преобразование картинок в эскизы» или «предпочитаю иметь возможность

выбирать из двух способов». Результаты опроса приведены в таблице 4.3. Большинство пользователей предпочитают иметь возможность выбирать между двумя способами черновой печати иллюстраций.

Таблица 4.3 Результаты опроса пользователей о способе черновой печати иллюстраций.

Вариант ответа	% пользователей
Предпочитаю повышение яркости картинок	8,7
Предпочитаю преобразование картинок в эскизы	13,0
Предпочитаю иметь возможность выбирать из двух способов	78,3

#### **4.3.4. Добавление в документ скрытых цифровых водяных знаков**

Для контроля оборота документов с ограничением прав доступа в них при печати добавляются явные и/или скрытые ЦВЗ.

##### **Требования к способу добавления ЦВЗ**

Сформулированы следующие требования к способу добавления и считывания скрытого ЦВЗ, содержащего идентификационный код устройства (MIC):

1. независимость от содержимого документа;
2. незаметность ЦВЗ при беглом просмотре документа;
3. считывание ЦВЗ с помощью сканирования на МФП и анализа сканированного изображения;
4. возможность идентификации документа даже по его отрезанной/оторванной части (не менее половины страницы) и первой копии;
5. информационная ёмкость не менее 8 байт;
6. отсутствие заметной задержки времени печати по сравнению с нормальным режимом.

Первое требование ведёт к необходимости печатать ЦВЗ в фоновых областях документа в виде шаблона чёрных (при чёрно-белой печати) или жёлтых (при цветной печати) точек. Отдельные точки на напечатанном документе похожи на случайные частички тонера и не бросаются в глаза. Однако при внимательном рассмотрении с помощью увеличительного стекла пользователь может заметить регулярную структуру точек и догадаться о существовании ЦВЗ [176].

Необходимо отметить, что в данном разделе под точкой понимается маленькое пятно тонера на бумаге, а не минимальный адресуемый бит в канале изображения, подготовленного для печати. Во-первых, для ряда разрешений печати единичный установленный бит, полностью окружённый нулевыми битами, часто не ведёт к прилипанию частичек тонера в соответствующей ему позиции. Во-вторых, размер пятна должен быть достаточным для считывания скрытого ЦВЗ с помощью сканера МФП с оптическим разрешением 600 dpi. Исходя из перечисленного, размер точки (диаметр пятна) устанавливается около 0,2 мм. Заметим, что точка такого размера практически незаметна невооружённым глазом.

### **Способ встраивания ЦВЗ**

Встраивание ЦВЗ в фоновые области предлагается реализовать на выходе из блока преобразования в печатный растр (см. рисунок 4.3), где можно анализировать цветовые каналы подготовленного для печати изображения и вносить в них изменения. Четвёртое требование в сочетании с тем, что часть точек может не пропечататься, или частички тонера отвалятся в процессе использования документа, ведёт к необходимости многократного дублирования информации (использования мажоритарного кодирования).

Предлагается способ встраивания скрытого ЦВЗ на основе следующих принципов:

- встраиваемые данные разбиваются на фрагменты;

- квадратные шаблоны из точек, где присутствие точки в определённой позиции означает единичный бит, а отсутствие – нулевой бит, содержат несколько дублей фрагментов данных;
- для обнаружения шаблонов около их углов печатаются вспомогательные метки;
- шаблоны с фрагментами данных многократно дублируются в таких случайных местах документа, где точки шаблона и вспомогательных меток попадают в фоновые области.

Без ограничения общности рассмотрим пример, когда размер встраиваемого в документ МПС составляет 12 байт. Разобьём его на 8 фрагментов данных. Каждый фрагмент, как показано на рисунке 4.15, содержит 16 бит, из которых три – это номер фрагмента, один – бит чётности, двенадцать – одна из восьми частей МПС. Бит чётности устанавливается в единицу, если номер фрагмента – нечётное число, и устанавливается в 0 в противном случае. Бит чётности используется для проверки при чтении номера фрагмента.

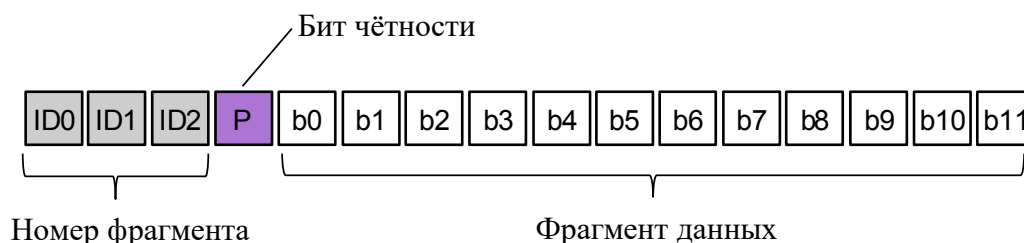


Рисунок 4.15 – Распределение бит фрагмента данных.

Фрагмент четырежды повторяется в шаблоне 8 на 8 точек. Существует множество вариантов дублирования бит фрагмента. На рисунке 4.16 показано два возможных варианта. Одинаковые биты обозначены одним и тем же цветом.

Точки в шаблоне расположены в узлах сетки с шагом в  $m$  мм (см. рисунок 4.17). Для обнаружения на странице шаблона около его четырёх углов печатаются вспомогательные метки. Каждая метка состоит из трёх точек, образующих прямоугольный треугольник с длинами катетов  $a$  и  $b$  мм как показано на рисунке 4.17. Расстояние по вертикали и горизонтали от угловой точки шаблона до

вершины треугольника, лежащей напротив гипотенузы, равно  $d$  мм. Все указанные размеры много больше размера точки.

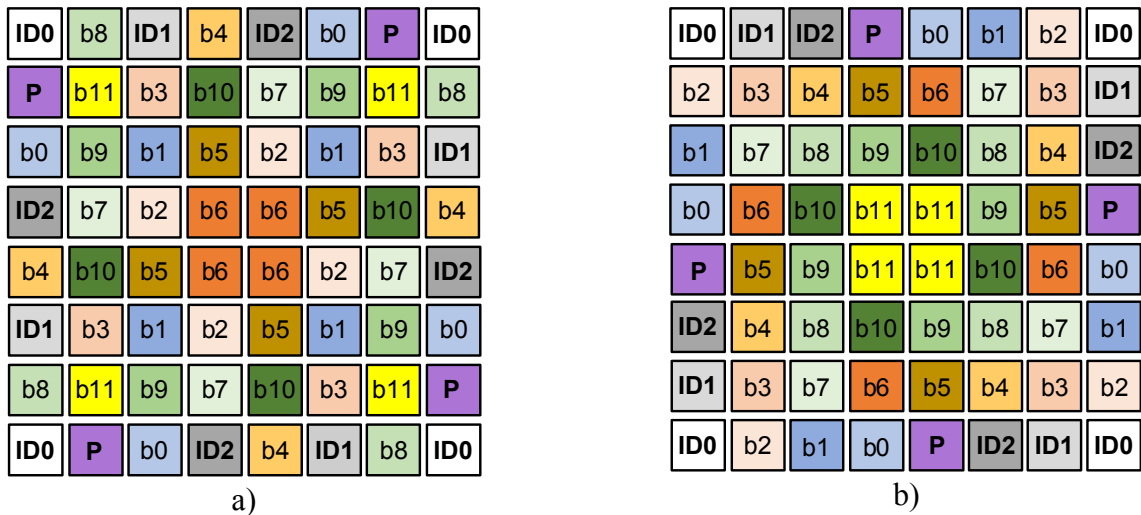


Рисунок 4.16 – Варианты дублирования бит фрагмента данных в шаблоне.

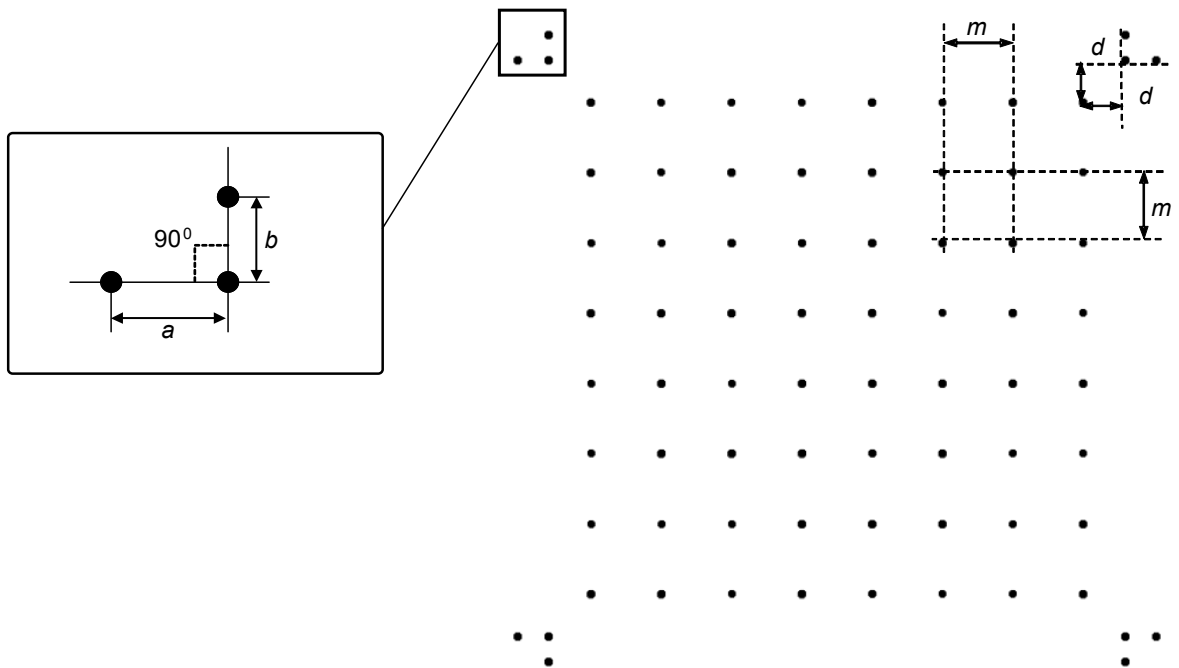


Рисунок 4.17 – Шаблон и вспомогательные метки.

С помощью генератора псевдослучайных чисел генерируется координата положения шаблона на странице. Далее проверяется, попадают ли точки шаблона

и вспомогательных меток в фоновые области, то есть равны ли нулю соответствующие окрестности этих точек элементы в каналах изображения, подготовленного для печати. Если это так, то точки шаблона и меток добавляются в канал чёрного при чёрно-белой печати или канал жёлтого при цветной. Если же хотя бы одна из точек шаблона попадает в печатаемые элементы или примыкает к ним, то генерируется новая случайная координата. Процесс продолжается до тех пор, пока 8 раз не будут добавлены в соответствующий канал печатаемого изображения шаблоны для всех фрагментов. Также итерации прерываются, если 30 попыток подряд найти свободную позицию для печати шаблона были неудачными. Для документов с тёмным фоном и при печати без полей возможно, что не удастся найти по 8 позиций для добавления шаблонов всех фрагментов МПС, но такой случай является нехарактерным. Для обычных текстовых документов не трудно найти несколько сотен подходящих позиций для добавления шаблона в печатаемый документ. Рисунок 4.18 демонстрирует два примера добавления шаблона в документ. В одном случае шаблон и метки расположены в фоновой области, в другом область частично занята линиями.

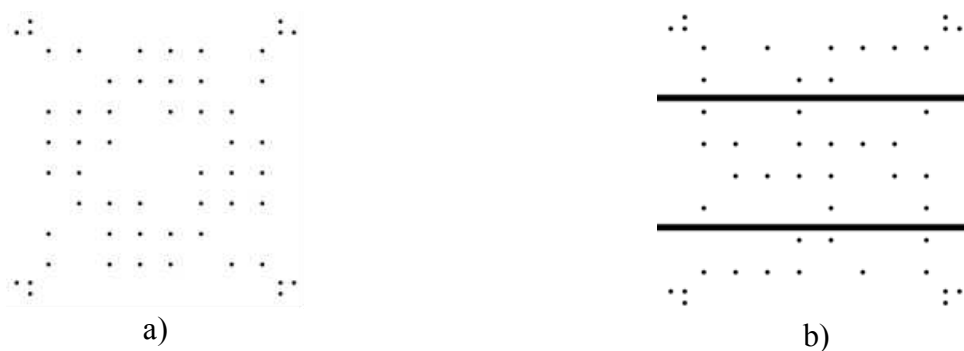


Рисунок 4.18 – Примеры добавления шаблонов в документ: а) шаблон и метки расположены в фоновой области; в этом примере номер фрагмента 5, данные F9E<sub>16</sub>; б) шаблон и метки расположены в области, частично занятой прямыми линиями; в этом примере номер фрагмента 7, данные 872<sub>16</sub>.



### Извлечение ЦВЗ

При извлечении ЦВЗ из сканированного изображения на первом этапе обнаруживаются вспомогательные метки. Значения параметров  $a$ ,  $b$ ,  $d$  и  $t$  известны. Если хотя бы две метки обнаружены, то проверяется наличие точек в требуемых позициях шаблона, таким образом считываются биты шаблона. Номер фрагмента повторяется в шаблоне четыре раза. Если номер проходит проверку на чётность/нечётность и совпадет по меньшей мере в трёх случаях, то считается, что шаблон извлечён, в противном случае данные этого шаблона исключаются из рассмотрения.

Несколько бит данных в шаблоне могут быть искажены. Точка, кодирующая единицу, может не пропечататься на бумаге или исчезнуть в процессе копирования. Отсутствие точки, кодирующее ноль, которое за счёт случайно прилипшей частички тонера или грязи на стекле сканера считывается как единица. Вторая ситуация существенно менее вероятна по сравнению с первой, поэтому оценим вероятность искажения отдельного единичного бита при извлечении ЦВЗ. Пусть значение бита было считано  $n$  раз,  $p$  – вероятность ошибки в бите, и искажения битов независимы. Результатом извлечения бита является наиболее часто встречающееся значение (мода) из всех повторов. Поскольку для чётного  $n$  возможна ситуация, когда 0 и 1 встретятся одинаковое количество раз, поэтому один из дублей отбрасывается, и  $n$  уменьшается на 1. Вероятность искажения извлеченного бита ЦВЗ  $p_D$ :

$$p_D = 1 - \sum_{l=0}^{(n-1)/2} C_n^l p^l (1-p)^{n-l}, \quad (4.4)$$

где  $C_n^l$  – число сочетаний из  $n$  по  $l$ .

На рисунке 4.19 приведены графики функции (4.4) для  $n = 3$  (встраивается 1 шаблон),  $n = 15$  (шаблон дублируется 4 раза),  $n = 31$  (шаблон дублируется 8 раз). Для каждой конкретной модели МФП и сценария использования документа можно оценить значение  $p$ . Например, при печати, копировании напечатанного документа

и последующем сканировании на МФП Samsung CLX-6240 неправильно считываются до 6 % точек, кодирующих единицы, и менее 1 % позиций, кодирующих нули, т.е.  $p=0,06$ . Благодаря тому, что каждый бит данных четыре раза дублируется в пределах шаблона и шаблоны несколько раз повторяются на странице, удаётся корректно извлекать данные МПС, т.к. вероятность искажения бита  $p_D$  близка к нулю. Кроме того, дополнительно осуществляется контроль валидности частей МПС, каждая из которых имеет диапазон допустимых значений.

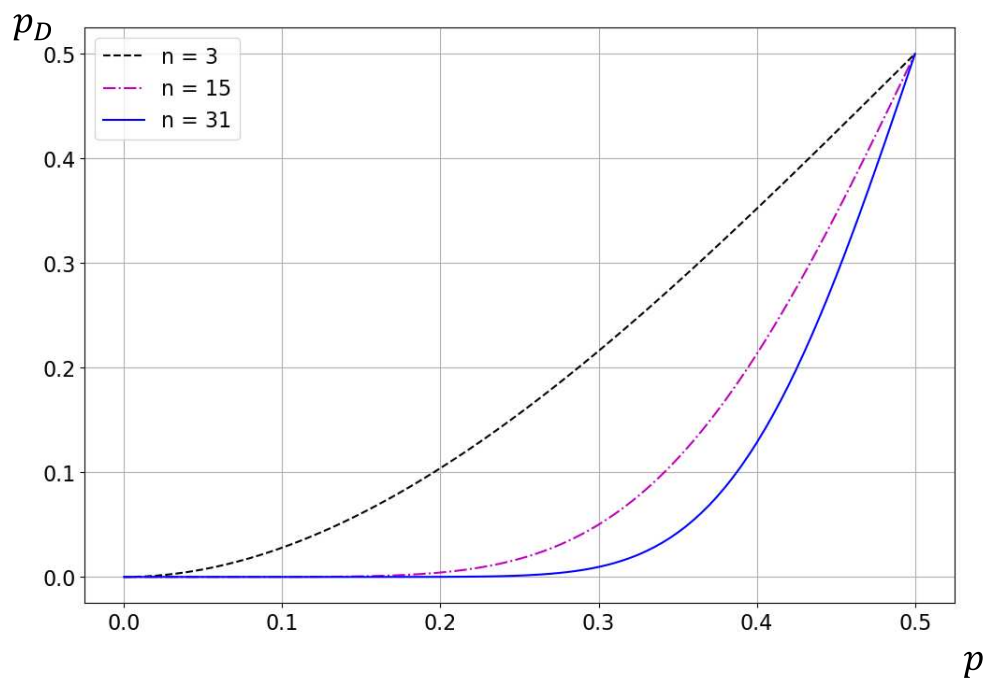


Рисунок 4.19 – Графики вероятности искажения бита в зависимости от числа дублирований  $n$ .

## Результаты

Для проверки возможности извлечения ЦВЗ из напечатанных искажённых документов был проведён следующий эксперимент. На лазерном чёрно-белом принтере Samsung ML-4450 [208] печатался обычный текстовый документ (чёрный текст на белом фоне, примерно 2000 символов английского текста на страницу, шрифт Times New Roman размером 12 пунктов, полуторный межстрочный интервал). При печати в документ добавлялся скрытый ЦВЗ, содержащий МПС.

Весь МІС разбивался на 8 фрагментов и каждый фрагмент повторялся в документе 8 раз. Таким образом в документе присутствует 64 шаблона. Исходный документ был отсканирован. Затем была получена бумажная копия документа и также отсканирована. Далее исходный документ был разорван пополам, и сканировались две половины документа. Для сканирования и копирования документа использовался МФП Samsung CLX-6240 [145]. Из сканированных изображений целой страницы напечатанного документа, двух его половин и копии извлекались шаблоны и считывался МІС. Результаты приведены в таблице 4.4. ЦВЗ корректно извлекается из половины страницы и копии документа.

Таблица 4.4 Результаты чтения идентификационного кода устройства (МІС).

Тест	Количество извлечённых шаблонов из 64	Минимальное количество дублей для одного бита	МІС прочитан верно?
Страница документа	63	28	Да
Верхняя половина страницы	28	12	Да
Нижняя половина страницы	26	12	Да
Первая копия	33	16	Да

Время печати документов из стандарта ISO 24734 [29] с добавлением ЦВЗ на чёрно-белом принтере Samsung ML-4450 [208] увеличилось по сравнению с обычным режимом на 1,1%, что практически незаметно для пользователя.

### Выводы по четвёртой главе

Рассмотрены специальные режимы печати, предназначенные для экономии расходных материалов и добавления в напечатанный документ защитных элементов. Сформулированы требования к режимам печати, реализуемым во встроенном программном обеспечении.

Предложен подход к реализации режимов специальной печати путём локальной модификации кода процессора растровых изображений, включающий:

1. определение в существующем коде модулей, где за счёт ветвления можно выполнить переход на процедуру реализации специального режима, а затем вернуться к существующей последовательности обработки;
2. выбор вычислительно простого алгоритма, имеющего компактную реализацию для используемой встроенной платформы;
3. переиспользование функций, существующих в коде процессора растровых изображений.

Определены следующие модули в процессоре растровых изображений для встраивания новых режимов печати:

1. модуль растеризации шрифтов, где можно изменять отображение отдельных символов текста;
2. модуль декодирования последовательности пикселей растровых изображений, на выходе которого можно изменять вид растровых изображений, переданных в метафайле;
3. модуль на выходе блока преобразования в печатный растр, где можно вносить изменения в каналы цвета подготовленного для печати изображения.

Предложенный подход позволил за короткое время добавить во встроенное программное обеспечение систем сканирования и печати следующие новые режимы, удовлетворяющие требованиям по времени и качеству работы:

- режим печати микротекста, позволяющий автоматически «на лету» генерировать микросимволы размером около 1 типографского пункта для широкого набора языков без использования специальных шрифтов; добавление на страницу микротекста увеличивает время печати на 0,01 с, что незаметно для пользователя;

- режим экономичной печати, который сокращает использование краски на 13,4% и обеспечивает лучшее визуальное качество символов по сравнению с альтернативными способами;
- режим черновой печати, в котором картинки преобразуются в цветные эскизы, при этом экономится до 80 % тонера, а время работы режима превышает время обычной печати лишь на 0,5 %;
- режим добавления на страницу скрытых цифровых водяных знаков, содержащих идентификационный код устройства; за счёт многократного дублирования информации способ позволяет извлечь скрытую информацию из копии и половины страницы.

## Глава 5. Подход к оценке и улучшению качества печатаемых фотографий

Цифровыми фотокамерами и камерами смартфонов ежедневно снимаются миллионы фотографий. Большинство фотографий остаётся в электронном виде, публикуется в облачных сервисах или распространяется в социальных сетях. Тем не менее, ряд пользователей печатают фотографии на фотопринтерах или универсальных цветных МФП. Можно выделить три группы пользователей:

- первые хотят получить напечатанные фотографии как можно быстрее и не желают тратить время и усилия на отбор фотографий;
- вторые готовы быстро отобрать качественные фотографии, для этого на ПК или мобильном устройстве использовать несколько простых способов устранения типичных дефектов;
- третьи – эксперты, которые могут заниматься подготовкой фотографий к печати часами, вручную редактируя изображения.

У пользователей первой группы печатаются в том числе испорченные изображения, что ведёт к неэффективному использованию расходных материалов. Часть напечатанных фотографий имеет посредственное качество. Для пользователей второй группы важно сократить время на исключение испорченных изображений и улучшение фотографий, нуждающихся в коррекции. Поэтому разработка функции автоматической оценки и повышения качества печатаемых фотографий является актуальной задачей. Требованиям экспертов трудно удовлетворить, но хорошо работающая автоматическая коррекция находит положительный отклик и у большинства пользователей данной группы

Файлы с фотографиями часто поступают в принтер непосредственно с камеры по проводному или беспроводному интерфейсу, или считываются с карт памяти. Поэтому способы оценки и улучшения качества изображения необходимо реализовать во встроенном ПО устройств печати. Вычислительные платформы компактных фотопринтеров обладают скудными ресурсами по сравнению с

платформами МФП для индустриального применения и, тем более, ПК, а время на подготовку фотографии к печати, включая коррекцию, не должно превышать нескольких секунд.

Большинство известных способов оценки и коррекции качества фотографий ориентированы на их просмотр на дисплее. Очевидно, что напечатанные фотографии выглядят иначе, и их вид зависит от используемой технологии и параметров печати. Требуется сформулировать подход к оценке и улучшению качества фотографий, учитывающий то, как выглядят именно напечатанные изображения, и позволяющий достичь высокой скорости обработки на целевой встроенной платформе.

### **5.1. Типичные дефекты фотографий**

В результате анализа большого количества фотографий, полученных с помощью различных камер, сделан вывод, что к наиболее часто встречающимся, раздражающим пользователей и, соответственно, требующим коррекции дефектам относятся [209]:

- размытие изображения;
- присутствие эффекта «красных глаз» при съёмке со вспышкой;
- затемнение целой фотографии или её отдельных частей;
- артефакты JPEG [23] сжатия;
- высокий уровень шума.

#### **Размытие**

Существует несколько возможных причин размытия или отсутствия чёткости/резкости фотографий. В [210] выделены четыре класса размытых (blur) изображений:

1. ошибка фокусировки камеры (out-of-focus);
2. смаз по причине движения фотографируемого объекта или камеры (motion blur);

3. хаотичные движения камеры из-за тряски или дрожания рук;
4. сложный характер размытия в силу нескольких причин.



a)



b)

Рисунок 5.1 – Иллюстрация восприятия резкости изображения в зависимости от его размера: а) фотография выглядит слегка нерезкой; б) та же фотография меньшего размера воспринимается чёткой.

Оценка резкости изображения наблюдателем сильно зависит от условий просмотра, в частности от размера изображения. Часто размытое изображение при



просмотре в уменьшенном виде выглядит чётким. Фотография на рисунке 5.1a выглядит слегка нерезкой, но та же фотография, будучи уменьшенной, на рисунке 5.1b воспринимается чёткой. Надо отметить, что обычной является ситуация, когда только часть изображения «в фокусе», а другая часть размыта. Такое изображение нельзя считать некачественным с точки зрения его резкости, хотя камера может сфокусироваться на ином объекте, чем планировал фотограф.

### **Эффект «красных глаз»**

Окрашивание на фотографии зрачка человека в красный цвет при использовании вспышки называют эффектом «красных глаз». Когда свет от вспышки падает на человеческий глаз под углом, не сильно отличающимся от оптической оси, он может проходить через зрачок, отражаться обратно от глазного дна примерно под тем же углом, под которым вошёл. Мелкие кровеносные сосуды на глазном дне дают отражённый свет красного цвета. На задней стенке сетчатки содержится меланин. Толщина слоя меланина коррелирует с пигментацией кожи и волос, поэтому у темнокожих и темноволосых людей обычно высокая концентрация на задней стенке сетчатки. Меланин ослабляет свет, проходящий через зрачок, и более высокое содержание меланина приводит к уменьшению выраженности эффекта «красных глаз». Пример изображения с эффектом «красных глаз» показан на рисунке 5.2.



Рисунок 5.2 – Изображение с эффектом «красных глаз».

Исследователи компании Kodak изучали частоту возникновения эффекта «красных глаз» для различных этнических групп в США (см. таблицу 5.1) [211]. Хотя внутри каждой национальности существуют некоторые различия в пигментации, для рас с более тёмным цветом лица в среднем эффект «красных глаз» возникает реже. Для европеоидов данный эффект возникает достаточно часто. Причём, процент детей европеоидов с заметным эффектом «красных глаз» достигает 82%, так как размер зрачка у детей больше, чем у взрослых. Поэтому коррекция «красных глаз» является необходимой функцией устройств, работающих с фотографиями.

Таблица 5.1 Процент людей с заметным эффектом «красных глаз» для различных этнических групп в США.

Этническая группа	% людей с заметным эффектом «красных глаз»
Афроамериканцы	9
Азиаты	15
Латиноамериканцы	41
Европеоиды	70

### Артефакты сжатия

Несмотря на существующие недостатки и имеющиеся лучшие альтернативы по отношению степень сжатия к качеству [212], формат JPEG [23] остаётся промышленным стандартом по работе с цифровыми фотографиями. Прореживание цветовых каналов, разбиение изображения на блоки и квантование на небольшое количество уровней, особенно в высоких частотах, ведут к образованию артефактов JPEG сжатия [213], среди которых наиболее заметны:

- блочные артефакты, при которых изображение выглядит разбитым на квадратные блоки, слегка отличающиеся оттенком;

- «звон» (ringing) или эффект Гиббса, выражающийся в ложном оконтуривании вдоль границ.

Пример изображения с JPEG артефактами показан на рисунке 5.4.



Рисунок 5.4 – Изображение с артефактами JPEG сжатия.

### **Затемнение целой фотографии или её отдельных частей**

При недостаточном времени экспозиции вся фотография получается недоэкспонированной, тёмной. Пропорциональное повышение контраста и применение гамма-коррекции, как правило, позволяет улучшить такие изображения [78].



Рисунок 5.3 – Изображение, снятое в контровом свете.

Существенно сложнее корректировать фотографии, на которых только часть получилась значительно более тёмной, чем хотелось. Такие изображения возникают, в частности, при фотографировании в контровом свете, когда на заднем плане расположен яркий источник света. В англоязычной литературе используется термин «задний свет» (backlight, backlit). Фотография, снятая в контровом свете, показана на рисунке 5.3. Из-за съёмки на фоне окна лицо человека получилось слишком тёмным.

### Высокий уровень шума

В плёночных фотоаппаратах для съёмки в условиях низкой освещённости использовалась плёнка с высокой ISO чувствительностью [214]. В цифровых камерах используется настройка с таким же названием «ISO», которая позволяет регулировать коэффициент усиления сигнала, считываемый с сенсора. При съёмке в сумерках выбирается большое значение «ISO», что ведёт к росту уровня шума на фотографии. Пример изображения с высоким уровнем шума показан на рисунке 5.5.

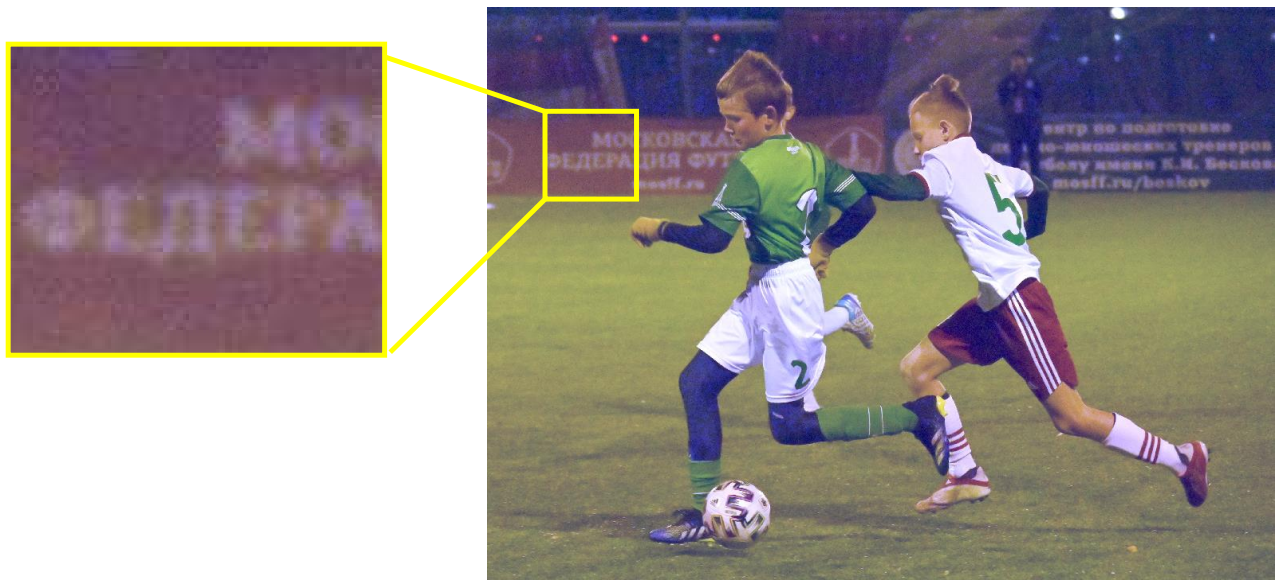


Рисунок 5.5 – Изображение с высоким уровнем шума.

## 5.2. Подход к автоматической коррекции фотографий

Поскольку фотопринтеры и МФП имеют несколько интерфейсов для получения фотографий, минуя ПК, алгоритмы оценки качества изображений и коррекции наиболее распространённых дефектов должны быть реализованы во встроенном ПО устройств печати. К разрабатываемым алгоритмам предъявляются следующие требования:

- a) Важно добиться высокого визуального качества напечатанных фотокарточек, а не только изображений на экране ПК.
- b) На основе результатов безреференсных («слепых») алгоритмов оценки качества [215] изображения без дефектов не должны изменяться, фотографии с дефектами улучшаются или же, если коррекция невозможна, то генерируется предупреждение для предотвращения печати некачественных фотокарточек.
- c) Высокая скорость работы на целевой встроенной платформе.

Чтобы удовлетворить перечисленным требованиям, сформулирован подход к автоматическому улучшению качества печатаемых фотографий:

1. Учитывать детали изображения, видимые для данного размера фотокарточки, разрешения и технологии печати.
2. Выполнять оценку характерных дефектов фотографий на основе машинного обучения по визуальным данным изображений и метаинформации.
3. Выполнять обработку в цветовом пространстве входных данных без лишней конвертации в промежуточные представления.

Современные фотокамеры снимают фотографии, содержащие десятки мегапикселей. При печати фотокарточки размера 10 на 15 см на термосублимационном фотопринтере с разрешением 300 dpi используется примерно 3 мегапикселя. Целесообразно изменять размер фотографии в соответствии с числом пикселей, используемых при печати. В большинстве случаев изменение размера – это уменьшение изображения, что позволяет не

анализировать мелкие детали, которые не видны на фотокарточке. Это существенно сокращает время обработки. Некоторые оттенки цвета отличаются на экране монитора и на напечатанной фотокарточке. В алгоритмах оценки и улучшения качества изображений следует ориентироваться на цвета, получаемые при печати.

Файлы с фотографиями, как правило, содержат, метаданные, которая хранится в формате EXIF (exchangeable image file format) [216] и в заголовке используемого графического формата. В EXIF-данных указываются использовавшиеся при съёмке параметры камеры, такие как время экспозиции, размер диафрагмы, чувствительность ISO, применялась ли вспышка и так далее. На основе параметров съёмки можно прогнозировать вероятность присутствия на изображении того или иного дефекта [217]. Очевидным примером является запись об использовании вспышки: если вспышка не применялась, то на изображении нет «красных глаз». Набор сохраняемых EXIF-данных значительно отличается для разных моделей камер, что затрудняет создание на их основе универсального алгоритма прогнозирования дефектов для фотопринтера, на вход которого могут поступать изображения, снятые различными устройствами. Тем не менее, удалось сформировать набор простых правил [218–220], которые позволяют для ряда фотографий не выполнять оценку дефектов на основе визуальной информации изображения, что сокращает общее время обработки и уменьшает количество ложных срабатываний.

Большинство фотографий, поступающих в фотопринтер, представлены в цветовом пространстве YCbCr формата JPEG [23], поэтому целесообразно выполнять обработку в этом пространстве.

Обсуждаемые в данной главе способы оценки и улучшения качества печатаемых фотографий изначально предназначались для термосублимационных фотопринтеров, но позднее были адаптированы для лазерных и струйных МФП. На основе предложенного подхода разработаны способы предотвращения печати

размытых фотографий и повышения резкости печатаемого изображения [121, 221, 222], алгоритм детектирования и коррекции эффекта «красных глаз» [223–227], метод оценки и улучшения затемнённых областей [228–233], способ печати анаглифных изображений из стереопары [234–236], где 3D эффект достигается за счёт цветового кодирования изображений, предназначенных для левого и правого глаза, и использования очков с двумя разными цветными фильтрами.

### **5.3. Примеры применения предложенного подхода**

В данном разделе показаны три примера применения подхода к оценке и улучшению качества печатаемых фотографий: способ оценки и повышения резкости изображения; алгоритм детектирования и исправления эффекта «красных глаз»; метод оценки и улучшения затемнённых областей на фотографиях, снятых в контровом свете.

#### **5.3.1. Способ оценки и повышения резкости**

##### **Схема способа**

Схема способа оценки и повышения резкости приведена на рисунке 5.6. На шаге 1 выполняется масштабирование изображения в соответствии с размером отпечатка и разрешением печати. На шаге 2 оценивается резкость изображения, в результате чего выставляется одна из четырёх оценок, перечисленных в таблице 5.2. Резкость оценивается по каналу яркости. Для цветового пространства YCbCr JPEG канал яркости это Y. Если изображение резкое (условие 3), то коррекция не требуется. Если изображение размыто сильно (условие 4), то предпочтительнее предупредить пользователя и отказаться от печати такой фотографии (шаг 5), потому что улучшить её до визуально качественной не удаётся.

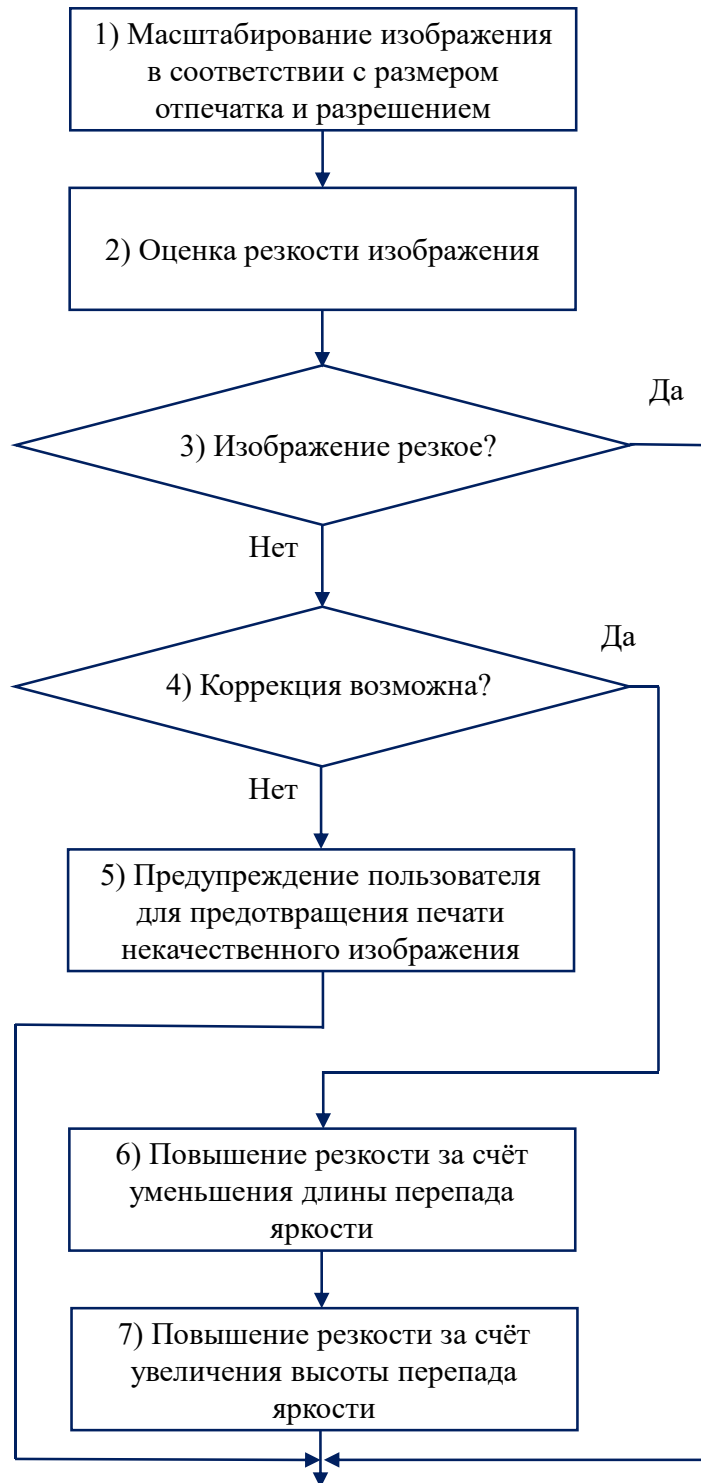


Рисунок 5.6 – Схема способа оценки и повышения резкости печатаемой фотографии.

Для размытых фотографии выполняется повышение резкости сначала за счёт уменьшения длины перепада яркости (шаг 6), затем путём увеличения высоты перепада яркости (шаг 7). Оба этих шага повышения резкости описаны в разделе



3.5 и выполняются для канала яркости. Повышение резкости за счёт уменьшения длины перепада яркости выполняется по формулам (3.38)–(3.44). Формулы (3.54)–(3.56) описывают повышение резкости за счёт увеличения высоты перепада яркости.

Как обсуждалось в разделе 3.5, для цветных изображений после преобразований в канале яркости для предотвращения изменения насыщенности необходимо скорректировать значения в цветовых каналах. Поскольку обработка выполняется в цветовом пространстве YCbCr, то выражения (3.49)–(3.51) преобразуются в следующие:

$$I''_{cb}(r, c) = \begin{cases} \frac{(I_{cb}(r, c) - 2^{D_l-1})I''_I(r, c)}{I_I(r, c)} + 2^{D_l-1}, & I_I(r, c) \neq 0, \\ I_{cb}(r, c), & \text{иначе,} \end{cases} \quad (5.1)$$

$$I''_{cr}(r, c) = \begin{cases} \frac{(I_{cr}(r, c) - 2^{D_l-1})I''_I(r, c)}{I_I(r, c)} + 2^{D_l-1}, & I_I(r, c) \neq 0, \\ I_{cr}(r, c), & \text{иначе,} \end{cases} \quad (5.2)$$

где  $I_I = Y$  – исходный канал яркости;  $I''_I$  – канал яркости после двух шагов повышения резкости;  $I_{cb}$  и  $I_{cr}$  исходные цветоразностные каналы Cb и Cr соответственно;  $D_l$  – глубина цвета на канал.

### Оценка резкости изображения

Рассмотрим способ оценки резкости фотографии. Алгоритмам оценки резкости изображений или обратной ей величины – размытости (blurriness) посвящено множество исследований. В обзоре [237] способы оценки резкости разбиваются на четыре группы:

1. основанные на вычислении градиентов или иных ФВЧ в пространственной области, причем градиенты вычисляются либо только по оцениваемому изображению [238, 239], либо сравниваются градиенты оцениваемой фотографии и её размытой копии [240, 241];
2. анализирующие спектр в частотной области [242, 243];
3. использующие машинное обучение [244, 245];

4. комбинирующие несколько различных оценок, полученных как в пространственной, так и частотной областях [210, 246].

Все существующие способы оценки резкости имеют два недостатка:

- зависимость величины оценки от содержимого фотографии, то есть оценки изображений с одинаковой степенью размытия, но со значительно отличающимся количеством контрастных контурных перепадов, как правило, существенно различаются;
- в присутствии на изображении шума оценки резкости перестают быть адекватными.

Способы оценки отличаются друг от друга выраженностью этих недостатков [121]. Алгоритмы, настроенные или обученные на искусственно размытых фотографиях, часто перестают считать размытыми фотографии при добавлении даже небольшого шума. Дисперсии оценок при одинаковой величине размытия в зависимости от содержимого фотографии у разных методов может отличаться в несколько раз. Следует отметить, что похожие проблемы при оценке резкости имеет и человек: изображение с шумом кажется более резким, чем оно же без шума, при выборе менее размытой фотографии наблюдатель скорее склонен выбрать более интересную и привлекательную по сюжету, даже если она не самая чёткая.

Предлагаемый способ оценки резкости основан на следующем наблюдении: вид гистограммы результата фильтрации размытого изображения с помощью ФВЧ перестаёт сильно изменяться тогда, когда размер ядра свёртки ФВЧ превышает размер ядра свёртки, с помощью которой выполняется размытие. Проиллюстрируем это наблюдение на изображении, которое является фрагментом одной из тестовых таблиц стандарта ИСО 15775 [32] и показано на рисунке 5.7. Изображение размывается с помощью усредняющего фильтра с размером ядра свёртки 3 на 3. Затем были получены четыре результата фильтрации ФВЧ с ядрами свёртки  $[1 \ -1]$ ,  $[1 \ 0 \ -1]$ ,  $[1 \ 0 \ 0 \ -1]$  и  $[1 \ 0 \ 0 \ 0 \ -1]$ . Для простоты используется горизонтальный градиент (конечная разность) с увеличивающимся шагом, но

аналогичные результаты можно получить для ряда других ФВЧ, например лапласианов.

Для изображений – результатов ФВЧ строятся гистограммы. На рисунке 5.8 показаны эти гистограммы в полулогарифмическом масштабе. Значения по оси значений пикселей нормализованы в диапазон от 0 до 1. Можно увидеть, что пока размер ядра свёртки ФНЧ меньше либо равен размеру ядра размытия (рисунок 5.8 а и б), вид гистограммы изменяется. Когда размер ядра свёртки ФНЧ превысил размер ядра размытия (рисунок 5.8с и d), гистограмма изменяется незначительно.



Рисунок 5.7 – Фрагмент тестовой таблицы.

Для характеристики вида гистограммы предлагается использовать энтропию [247]:

$$E_n = - \sum_i h'(i) \log(h'(i)), \quad (5.3)$$

где  $h'$  – нормализованная гистограмма (см. формулу (1.4)) результата ФВЧ; суммирование выполняется по всему диапазону возможных значений  $i$ .

Вид зависимости  $E_n$  от размера ядра свёртки ФНЧ различается для чётких и размытых фотографий даже в присутствии умеренного шума, но сильно зависит от содержимого изображения. Если разделить  $E_n$  на  $\sum_i h'(i)$ , то влияние содержимого изображения на вид зависимости от размера ядра свёртки ФНЧ существенно уменьшается:

$$A = \frac{-\sum_i h'(i) \log(h'(i) + 1)}{\sum_i h'(i)} = -\sum_i \log(h'(i) + 1). \quad (5.4)$$

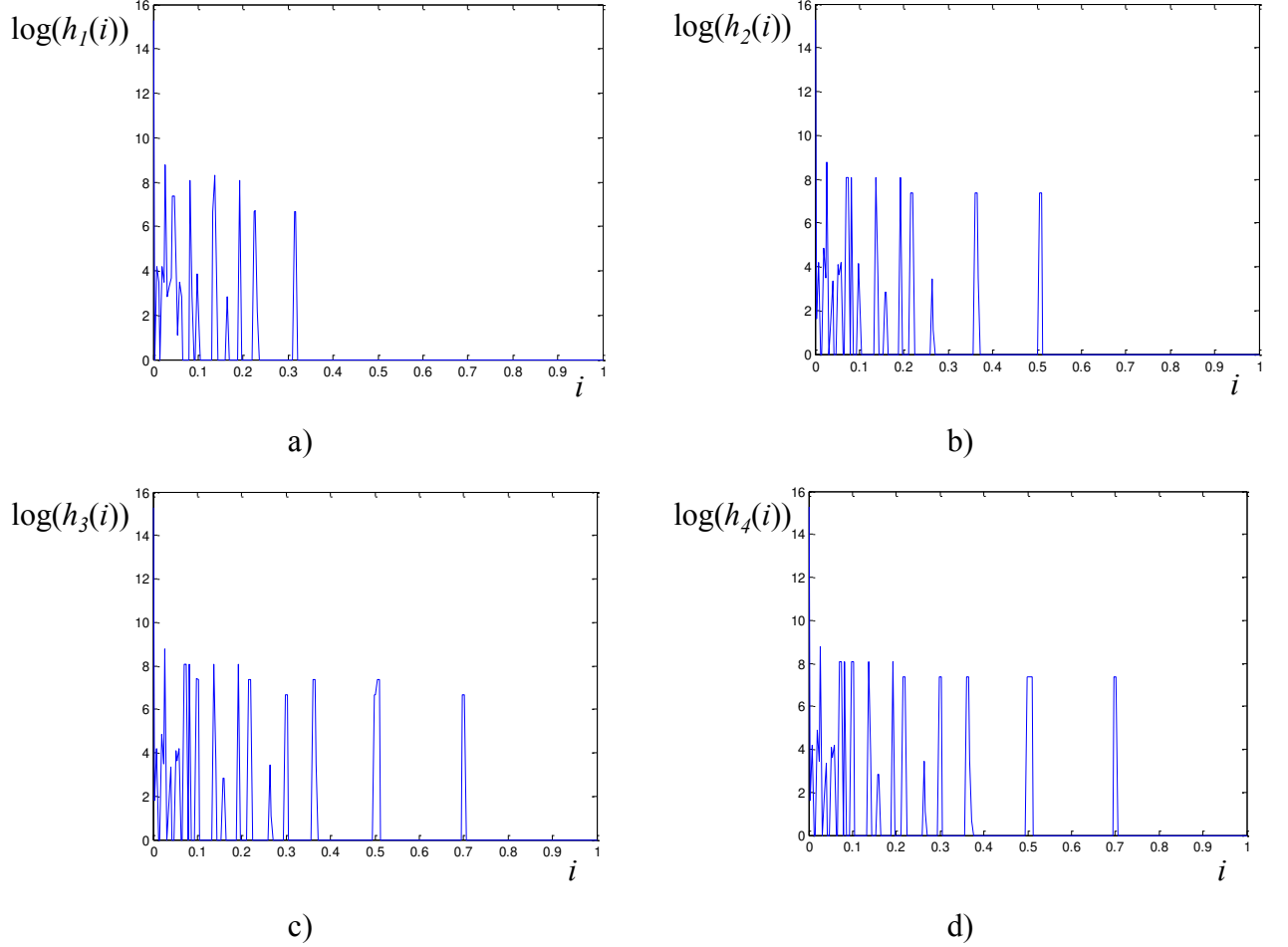


Рисунок 5.8 – Гистограммы в полулогарифмическом масштабе результатов свёртки тестового изображения с ядром ФВЧ: а)  $[1 \ -1]$ ; б)  $[1 \ 0 \ -1]$ ; в)  $[1 \ 0 \ 0 \ -1]$ ; г)  $[1 \ 0 \ 0 \ 0 \ -1]$ .

Под логарифмом единица добавляется к  $h'(i)$ , чтобы избежать некорректной математической операции для нулевых столбцов гистограммы. Значения  $A$  вычисляются для размера ядра свёртки ФНЧ  $S$  от 2 до 10, затем нормализуются:

$$A'(S) = \frac{A(S)}{\max_{S \in [2, 10]} A(S)}. \quad (5.5)$$

На рисунке 5.9 показаны примеры  $A'(S)$  для размытого и резкого изображений. Для удобства визуального сравнения графиков  $A'(1)=0$ , хотя это

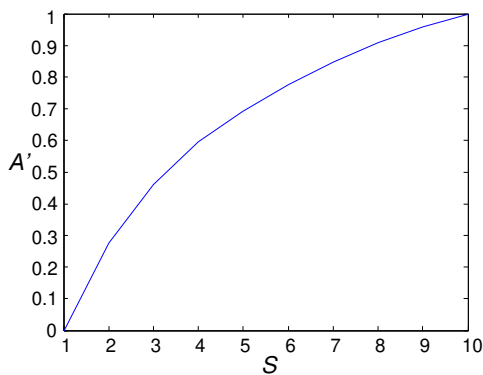
значение не используется в дальнейших вычислениях. Эти примеры являются характерными как для размытых, так и для чётких фотографий:  $A'(S)$  для резких фотографий быстро растёт для  $S$  равных 2 и 3, чего не наблюдается для размытых фотографий; площадь под графиком для резких изображений больше площади для размытых изображений.

Предлагается описать перечисленные отличия графиков следующими признаками и на их основе разработать классификатор:

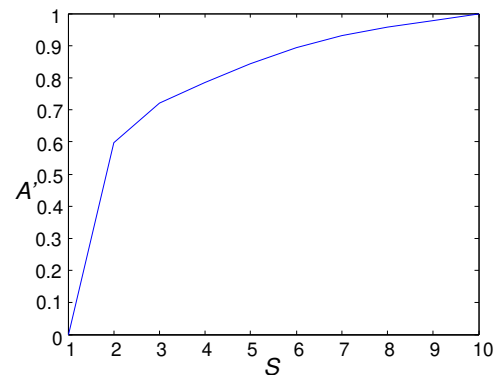
$$F_1 = A'(2), \quad (5.6)$$

$$F_2 = A'(3) - A'(2), \quad (5.7)$$

$$F_3 = \sum_{S=2}^{10} A'(S). \quad (5.8)$$



a)



b)

Рисунок 5.9 – Примеры зависимости  $A'$  от  $S$  – размера ядра ФВЧ: а) для размытого изображения; б) для резкого изображения.

Исходя из тех же соображений, что перечислены в разделе 3.4, для выполнения классификации выбирается адаптивный бустинг в модификации Gentle AdaBoost [105] на деревьях малой глубины. Для обучения собран и размечен набор из 2700 фотографий.

Эксперименты с обучением классификатора на четыре класса показывают, что плохо различаются классы в парах «резкое» – «слегка размытое» и «слегка размытое» – «размытое», а классы «резкое» – «сильно размытое» различаются

хорошо. Поэтому AdaBoost классификатор обучается для двух классов – «резкое» и «сильно размытое», затем для получающегося на выходе классификатора уровня уверенности (confidence level)  $C$  путём максимизации макро-F1-меры [101] определяются три порога  $T_1, T_2, T_3$ : если  $C \geq T_1$ , то изображение относится к классу «сильно размытое»; если  $T_1 > C \geq T_2$ , то относится к классу «размытое»; если  $T_2 > C \geq T_3$ , то относится к классу «слегка размытое»; если  $T_3 > C$ , то относится к классу «резкое».

### Результаты

Для оценки качества классификации размеченный набор изображений разбивался на 5 групп со стратификацией и в процедуре двойного скользящего контроля (аналогично описанному в разделе 3.4.) вычислялось математическое ожидание макро-F1-меры, которое составило 0,76. В работе [248], где решается задача детектирования размытых фотографий для предотвращения их печати, F1-мера равна 0,67.

Для сравнения предлагаемого способа оценки резкости с мнением наблюдателей был проведён опрос. Десять изображений различной степени резкости, которые не входили в обучающую выборку, были напечатаны размером 10 на 15 см на термосублимационном фотопринтере Samsung SPP-2040 [249]. В опросе участвовало 31 человек, которые выставляли оценки напечатанным фотокарточкам в соответствии с таблицей 5.2. Коэффициент корреляции Пирсона [210] между предлагаемой оценкой резкости и средним баллом мнений (MOS) приведён в таблице 5.3. Также в этой таблице содержатся коэффициенты корреляции Пирсона с MOS для существующих способов оценки резкости digital sharpness scale (DSS) [238] и Asub [241]. Предлагаемый способ хорошо согласуется с оценкой человека и превосходит существующие.

Таблица 5.2 Шкала оценок резкости изображений.

Класс	Оценка	Описание
Резкое	0	Определённо чёткое изображение
Слегка размытое	1	Слегка размытое, но это не воспринимается как дефект, требующий коррекции
Размытое	2	Нечёткое, обязательно требуется коррекция
Сильно размытое	3	Размыто настолько сильно, что предпочтительнее отказаться от печати такой фотографии

Для сравнительной оценки способов повышения резкости пять фотографий обрабатывались с помощью известных программных средств для улучшения размытых фотографий Adobe Photoshop SmartSharp, Nik Sharpener [250], FocusMagic [251] и предлагаемым способом. Во всех случаях использовался либо автоматический подбор параметров, если такая возможность предоставлялась ПО, либо параметры по умолчанию. Обработанные и исходные изображения были напечатаны на фотопринтере Samsung SPP-2040. В опросе участвовало 31 человек, которые были проинструктированы для каждой из пяти фотографий ранжировать пять напечатанных фотокарточек (4 способа повышения резкости и необработанное) в порядке убывания качества, где под качеством понимались резкость изображения и отсутствие видимых артефактов. Интервьюируемые не знали способ обработки. Самое качественное изображение получало оценку  $E$  в 5 баллов, следующее 4, и так далее до 1. Исходное и улучшенные изображения для одной из пяти использовавшихся в опросе фотографий показаны на рисунке 5.10.

Таблица 5.3 Коэффициент корреляции Пирсона между оценкой резкости и средним баллом мнений (MOS).

Оценка	Коэффициент корреляции Пирсона
DSS [238]	0,589
Asub [241]	0,907
Предлагаемая	<b>0,924</b>



a)



b)



c)



d)



e)

Рисунок 5.10 – а) исходная нерезкая фотография; б) Adobe Photoshop SmartSharp; в) Nik Sharpener; г) FocusMagic; е) предлагаемый.



В таблице 5.4 приведены средние оценки по всем участникам опроса и пяти фотографиям. Nik Sharpener делает изображения заметно резче чем исходные и не ведёт к образованию каких-либо артефактов. Тем не менее, его результаты выглядят менее чётким по сравнению с другими способами. Результаты обработки Adobe Photoshop SmartSharp резкие, но хорошо заметен эффект ореолов вдоль контрастных границ. FocusMagic получает наиболее чёткие изображения, но усиливает шумы и искажает текстуры, что заметно на напечатанной фотокарточке. Несмотря на это многие участники опроса выбирали результаты FocusMagic как лучшие. Наивысшую среднюю оценку получил предлагаемый способ повышения резкости, который значительно улучшает качество размытых фотографий и не приводит к возникновению видимых артефактов.

Таблица 5.4. Средние оценки способов повышения резкости по результатам опроса.

Способ	$\bar{E}$
Исходная фотография	1,1
Adobe Photoshop SmartSharp	2,9
Nik Sharpener	2,4
FocusMagic	4,2
Предлагаемый	<b>4,4</b>

Время оценки и улучшения резкости фотографий на целевой платформе измерялось с помощью эмулятора и составило около 1,6 с.

### 5.3.2. Детектирование и коррекция эффекта «красных глаз»

#### Существующие подходы

В литературе описано несколько десятков способов автоматической, то есть без участия человека, коррекции эффекта «красных глаз». Как правило, их работа заключается в детектировании, затем ретушировании искажённых областей.

Детектирование включает сегментацию областей-кандидатов и их последующую классификацию. Способы сегментации областей-кандидатов можно разбить на следующие группы:

- ограничения по порогам в цветовом пространстве YCbCr [252], RGB [253],  $La^*b^*$  [254];
- пороговое ограничение в специальном цветовом канале «красноты» (redness) [255, 256];
- корреляция со структурным элементом круглой формы в цветовом канале «красноты» [257, 258].

После сегментации образуется большое количество областей-кандидатов, но только несколько из них относятся к «красным глазам». Для отбрасывания ложно положительных кандидатов используются такие способы:

- детектирование лиц и рассмотрение кандидатов только в местах предположительного нахождения глаз [259, 260], для детектирования лиц, как правило, применяется алгоритм Виолы-Джонса [261];
- рассматриваются только области-кандидаты, около которых есть яркое пятно, соответствующее отблеску (glint) в зрачке [253];
- поиск среди областей-кандидатов пары со сходными характеристиками, что с учётом геометрических ограничений позволяет детектировать пары «красных глаз» [262].
- построение дерева решений на основе признаков, учитывающих размер, форму области кандидата и цвета в её ближайшей окрестности, в частности присутствие оттенков кожи человека (skin tones) [254];
- обучение классификатора, где используются признаки, описывающие форму и цвета области-кандидата, а также её локальной окрестности [258, 259].

При коррекции областей «красных глаз» обычно применяется комбинация следующих способов:

- итеративное расширение областей [263] на соседние пиксели, имеющие оттенки, характерные для «красных глаз» [264];
- понижение насыщенности и яркости для пикселей области [253];
- уменьшение значения в канале красного цвета [265];
- размытие области [266];
- альфа-смешивание в области глаз исходного изображения и затемнённого канала яркости [256].

### **Анализ набора фотографий с эффектом «красных глаз»**

Целесообразно начать разработку метода детектирования и коррекции эффекта «красных глаз» со сбора и анализа представительного набора фотографий, имеющих такой дефект и снятых различными камерами. Собранный набор включает 900 фотографий, на которых суммарно есть 2738 «красных глаз». Из них 1304 пар глаз, а для 130 глаз второй не виден или не «красный». Все дефекты разделены на две группы: явно заметные (рисунок 5.11a) и слабо заметные (рисунок 5.12b). Доля первых в наборе составила 72%, а вторых 28%. Распределение изображений по количеству «красных глаз» на них приведено в таблице 5.5. На большинстве фотографий 2 или 4 «красных глаза», но и другие варианты встречаются нередко.

Поскольку многие из описанных в литературе методов на первом этапе используют детектирование лиц человека в анфас с помощью алгоритма Виолы-Джонса [261], то интересно оценить, какая в этом случае доля «красных глаз» не обнаруживается. Распределение положений лиц на изображениях приведено в таблице 5.6. Положение лица в анфас есть на 72 % фотографий, для обнаружения остальных требуется применять дополнительные классификаторы или использовать более сложный и ресурсоёмкий подход для детектирования лиц.

По изображениям набора были собраны значения цветов пикселей в области «красного глаза» и оттенки кожи человека из соседних с глазами зон. Всего собрано примерно по 5000 примеров цвета пикселей как для глаз, так и для кожи. На

рисунке 5.12 показано как расположены эти цвета в координатах  $a^*b^*$  цветового пространства  $La^*b^*[8]$ . Диапазон изменения оттенков довольно широк. Цвет «красных глаз» изменяется от жёлтого до пурпурного. Кроме того, эти два диапазона цветов частично перекрываются друг с другом, что в общем случае делает невозможным успешное детектирование областей дефектов за счёт порогового отсечения.



a)



b)

Рисунок 5.11 – Примеры «красных глаз»: а) явно заметные; б) слабо заметные.

Таблица 5.5 Распределение изображений по количеству «красных глаз» на них.

Количество «красных глаз»	% фотографий
1	1
2	59
3	5
4	22
5	3
6	7
больше 6	3

Таблица 5.6 Распределение положений лиц на изображениях.

Положение лица	% фотографий
Анфас	72
Повёрнуто в плоскости изображения	17
Положение между профиль и анфас	6
Частично заслонено	5

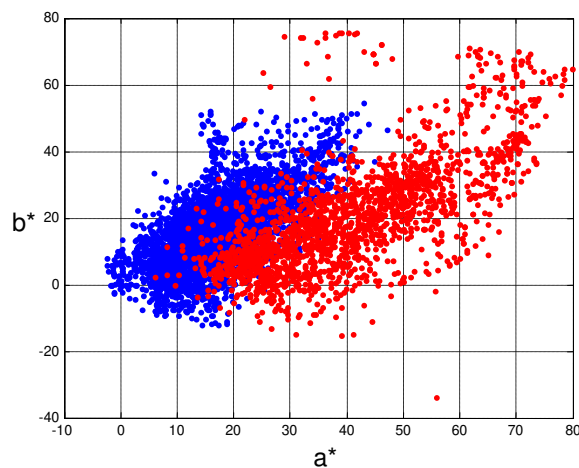


Рисунок 5.12 – Оттенки кожи человека (синие точки) и «красных глаз» (красные точки) в координатах  $a^*b^*$  цветового пространства  $La^*b^*$ .

### Схема метода

В соответствии с предлагаемым подходом, на основе обзора литературы и анализа набора изображений автором предложен метод детектирования и коррекции эффекта «красных глаз» [223–227], схема которого показана на рисунке 5.13. Обработка выполняется в цветовом пространстве  $YCbCr$ . На первом шаге выполняется масштабирование изображения в соответствии с его размером на бумаге и разрешением печати. Как отмечалось выше, в абсолютном большинстве случаев фотографии уменьшаются при масштабировании. При этом маленькие по площади «красные глаза» перестают быть заметными, так что нет необходимости их детектировать и корректировать. Далее проверяется условие 2. Если файл изображения содержит метайнформацию, в которой указано, что вспышка не

применялась, то «красных глаз» на изображении нет, и работа способа завершается. Если же вспышка при съёмке использовалась или это неизвестно, то выполняется переход на детектирование областей-кандидатов.

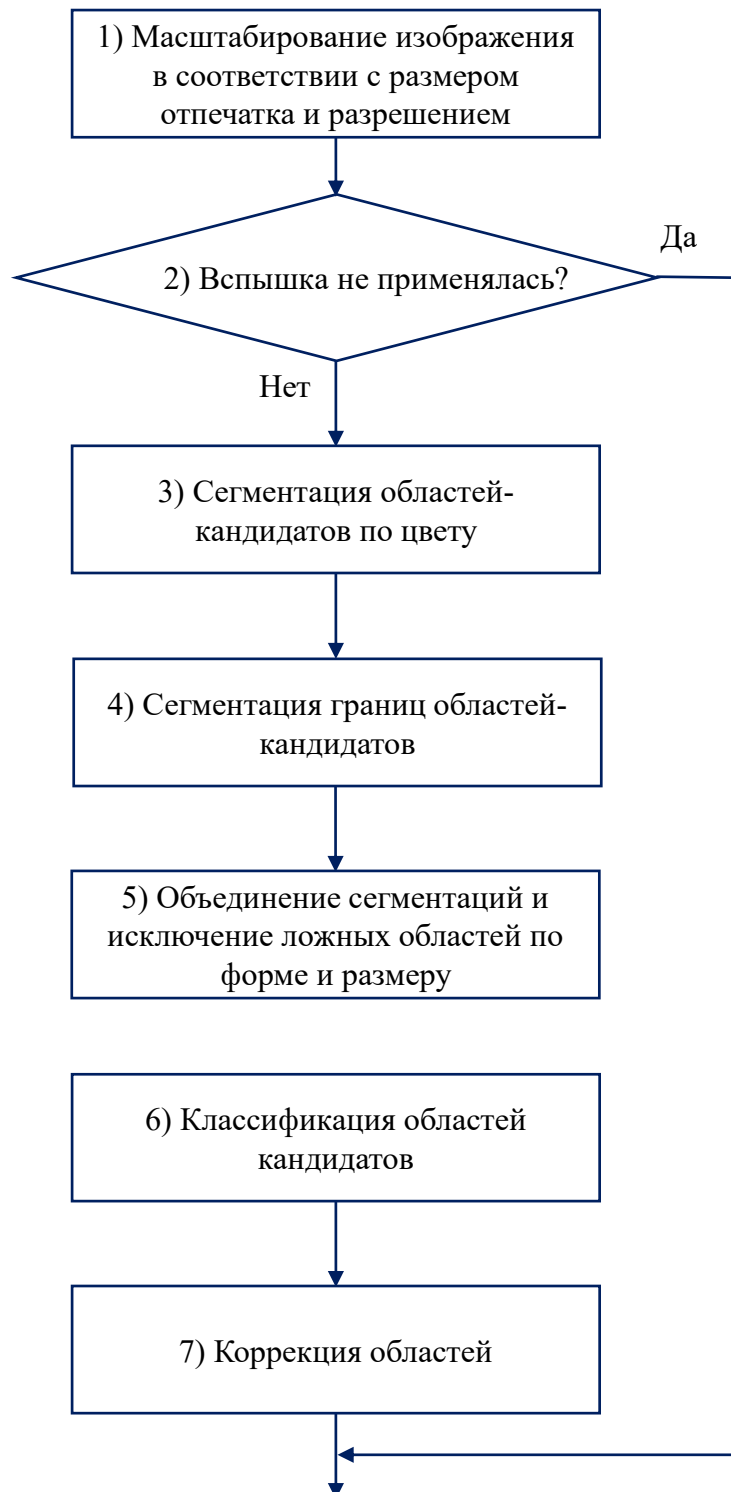


Рисунок 5.13 – Схема метода детектирования и коррекции «красных глаз».

На шаге 3 выполняется сегментация пикселей областей-кандидатов по цвету. В отличие от описанных в литературе подходов, на основе собранных 5000 значений оттенков в области «красных глаз» в пространстве YCbCr построена 3D таблица, которая задаёт значения сегментируемых пикселей. Сегментация границ областей-кандидатов выполняется на шаге 4. В отличие от существующих способов, которые для поиска округлых областей красных оттенков применяют набор корреляционных фильтров с круглым структурным элементом, предлагается использовать четыре направленных фильтра для нахождения границ областей слева, справа, сверху и снизу, что позволяет искать области глаз любого размера и, в том числе, в форме эллипса.

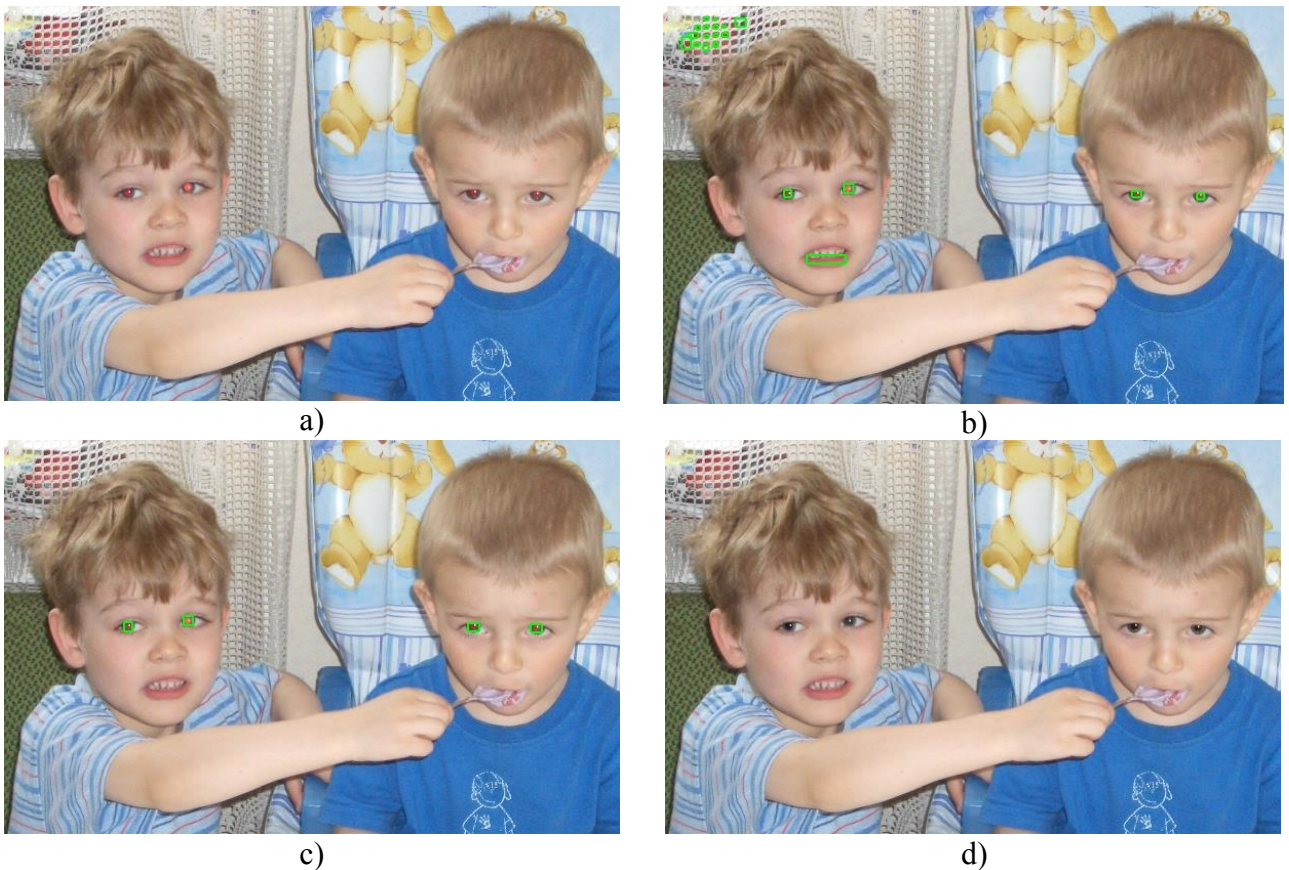


Рисунок 5.14 – Пример работы предлагаемого способа: а) исходное изображение; б) сегментированные области-кандидаты (указаны зелёным); в) детектированные области «красных глаз»; д) скорректированное изображение.

Далее (шаг 5) результаты сегментаций границ и по цвету объединяются, выполняется поиск связанных областей и ложные области исключаются, исходя из их формы и размера. На шаге 6 для классификации областей используется адаптивный бустинг в модификации Gentle AdaBoost [105] на деревьях малой глубины. Преимущества AdaBoost для реализации во встроенном ПО обсуждались в разделе 3.4. Коррекция цвета областей, классифицированных как «красный глаз», выполняется на шаге 7.

Пример работы предлагаемого способа показан на рисунке 5.14. После шагов сегментации помимо «красных глаз» выделены губы и красные области в левом верхнем углу фотографии (рисунке 5.14b). Области «красных глаз» обнаруживаются с помощью классификации (рисунке 5.14c). В этих областях выполняется исправление эффекта «красных глаз» (рисунке 5.14d).

### Сегментация областей-кандидатов

Направленные фильтры основаны на отношении сумм значений в квадратных блоках 3 на 3 пикселя канала «красноты», который равен сумме канала яркости  $I_I$  ( $I_I=Y$  в YCbCr) и канала  $I_{Cr}$  ( $I_{Cr}=Cr$ ), умноженного на 5. Суммы в блоках вычисляются по формулам:

$$B_0(r, c) = \sum_{i=r-1}^{r+1} \sum_{j=c-1}^{c+1} (I_I(r, c) + 5I_{Cr}(r, c)), \quad (5.9)$$

$$B_1(r, c) = \sum_{i=r-4}^{r-2} \sum_{j=c-1}^{c+1} (I_I(r, c) + 5I_{Cr}(r, c)), \quad (5.10)$$

$$B_2(r, c) = \sum_{i=r+2}^{r+4} \sum_{j=c-1}^{c+1} (I_I(r, c) + 5I_{Cr}(r, c)), \quad (5.11)$$

$$B_3(r, c) = \sum_{i=r-1}^{r+1} \sum_{j=c-4}^{c-2} (I_I(r, c) + 5I_{Cr}(r, c)), \quad (5.12)$$



$$B_4(r, c) = \sum_{i=r-1}^{r+1} \sum_{j=c+2}^{c+4} (I_l(r, c) + 5I_{cr}(r, c)). \quad (5.13)$$

Направленные фильтры:

$$E_d(r, c) = \frac{B_0(r, c)}{1 + B_d(r, c)}, \quad (5.14)$$

где  $d \in [1, 4]$ ; высокие значения  $E_1$  соответствуют границе сверху от светлой области в канале красноты; для  $E_2$  границе снизу; для  $E_3$  границе слева; для  $E_4$  границе справа.

Бинарный результат сегментации границ областей равен:

$$I_e(r, c) = \begin{cases} 1, & \max_{d \in [1, 4]} E_d(r, c) > T_e, \\ 0, & \text{иначе,} \end{cases} \quad (5.15)$$

где  $T_e$  – порог.

Помимо результата сегментации границ  $I_e$ , вычисляются четыре бинарных результата сегментации границ для каждого из фильтров  $E_d$ , которые показывают, какой из направленных фильтров имел наибольшее значение в данном пикселе:

$$I_{e_d}(r, c) = \begin{cases} 1, & E_d(r, c) = \max_{d \in [1, 4]} E_d(r, c) \text{ AND } I_e(r, c) = 1, \\ 0, & \text{иначе.} \end{cases} \quad (5.16)$$

Пример сегментации границ областей в канале «красноты» показан на рисунке 5.15. Пиксели верхних, нижних, правых и левых границ обозначены на рисунке 5.15b разными цветами. Для округлых и эллипсообразных областей границы обнаруживаются по меньшей мере с двух сторон, чаще с трёх или четырёх.

Результаты сегментации по цвету  $I_c$  и границ  $I_e$  объединяются логическим ИЛИ:

$$I_b(r, c) = I_c(r, c) \text{ OR } I_e(r, c). \quad (5.17)$$

На изображении  $I_b$  выполняется поиск связных областей и для каждой области  $\Omega$  вычисляются следующие признаки. Сумма пикселей области (площадь):

$$N_\Omega = \sum_{(r, c) \in \Omega} I_b(r, c). \quad (5.18)$$

Сумма пикселей, сегментированные по цвету:

$$N_{\Omega c} = \sum_{(r,c) \in \Omega} I_c(r, c). \quad (5.19)$$

Суммы пикселей, сегментированные каждым из направленных фильтров:

$$N_{\Omega e_d} = \sum_{(r,c) \in \Omega} I_{e_d}(r, c). \quad (5.20)$$

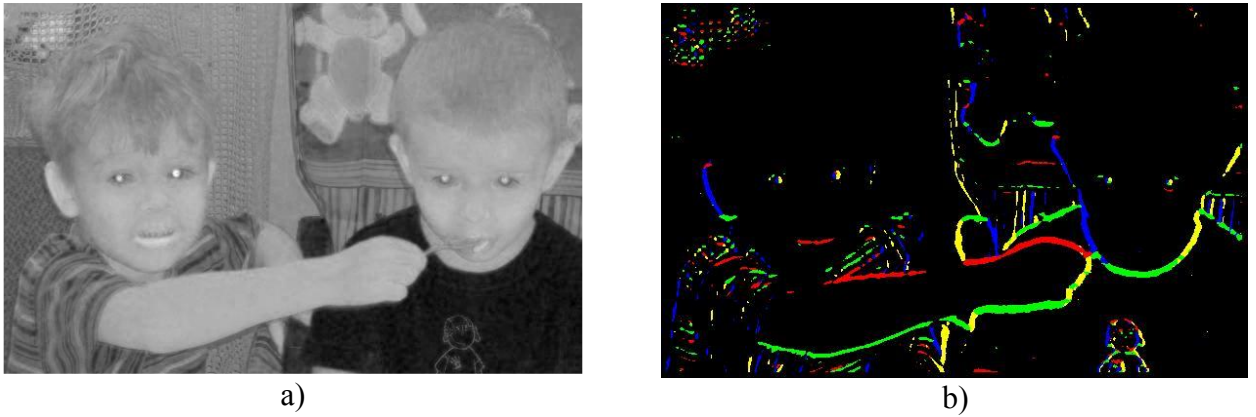


Рисунок 5.15 – Пример работы направленных фильтров: а) цветовой канал «красноты»; б) цветом отмечены сегментированные пиксели границ: красным показаны пиксели  $I_{e_1}$ ; зелёным –  $I_{e_2}$ ; синим –  $I_{e_3}$ ; жёлтым –  $I_{e_4}$ .

Размеры описывающего прямоугольника:

$$S_{\Omega 1} = \max_{(r,c) \in \Omega} r - \min_{(r,c) \in \Omega} r, \quad (5.21)$$

$$S_{\Omega 2} = \max_{(r,c) \in \Omega} c - \min_{(r,c) \in \Omega} c. \quad (5.22)$$

Коэффициент компактности:

$$k_{c\Omega 1} = \frac{N_{\Omega}}{S_{\Omega 1} S_{\Omega 2}}. \quad (5.23)$$

Коэффициент удлинения:

$$k_{e\Omega 1} = \frac{\min(S_{\Omega 1}, S_{\Omega 2})}{\max(S_{\Omega 1}, S_{\Omega 2})}. \quad (5.24)$$

Количество ненулевых  $N_{\Omega e_d}$  :

$$M_{\Omega} = \sum_{d \in [1,4]} \text{sign}(N_{\Omega e_d}). \quad (5.25)$$

Доля максимального из  $N_{\Omega e_d}$  в площади области:

$$k_{a\Omega} = \frac{\max_{d \in [1,4]} N_{\Omega e_d}}{N_{\Omega}}. \quad (5.26)$$

Для исключения ложно положительных сегментированных областей последовательно применяются три элементарных классификатора:

1. Правило  $N_{\Omega c} = 0$  OR  $M < 2$ , то есть в области нет цветов, характерных для «красных глаз», или меньше двух направленных фильтров обнаружили перепад.
2. На основе признаков  $k_{c\Omega 1}$ ,  $k_{e\Omega 1}$ ,  $M_{\Omega}$ ,  $k_{a\Omega}$  строится дерево решений, которое отбрасывает области по форме. При построении дерева решений на собранном наборе изображений минимизируется число ложно отрицательных областей для явно заметных «красных глаз».
3. В предположении того, что лицо может занимать всё изображение, и на основе информации о типичных соотношениях размеров лица и глаза человека [267] вводятся ограничения на размеры описывающего прямоугольника.

### Признаки для классификации областей

При разработке классификатора оставшихся областей-кандидатов исходно рассматривалось 80 признаков. Помимо указанных в формулах (5.17)-(5.25), это были: различные отношения между  $N_{\Omega}$ ,  $N_{\Omega c}$ ,  $N_{\Omega e_d}$ ; процент пикселей с оттенками кожи человека в ближайшей окрестности разных размеров; процент пикселей с цветами, нехарактерными для кожи человека в ближайшей окрестности; контраст; средние значения и дисперсии цветовых каналов в сегментированном регионе и ограничивающем его прямоугольнике; процент ярких пикселей в локальной окрестности; признаки, описывающие форму области. Применённая реализация Gentle AdaBoost [106] для ансамбля деревьев отобрала 48 признаков. Наиболее информативными для классификации являются следующие признаки:

- Доля пикселей с оттенками кожи человека в прямоугольнике с размерами  $5S_{\Omega_1}$  на  $5S_{\Omega_2}$  и центром в середине ограничивающего область прямоугольника; оттенки кожи человека определяются с помощью 3D таблицы, построенной в пространстве YCbCr на основе собранных 5000 значений.
- Доля ярких пикселей в ограничивающей область прямоугольнике  $\Psi$ :

$$k_{h\Omega} = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Psi} 1 \mid (I_I(r,c) > T_{h1} \text{ AND } I_{Cr}(r,c) > T_{h2} \text{ AND} \\ \text{AND } I_{Cr}(r,c) < T_{h3} \text{ AND } I_{Cb}(r,c) > T_{h4} \text{ AND } I_{Cb}(r,c) < T_{h5}), \quad (5.27)$$

где  $T_{hi}$  – пороги.

- Доля пикселей с цветом, характерным для «красных глаз» в сегментированной области:

$$k_{r\Omega} = \frac{N_{\Omega c}}{N_{\Omega}}. \quad (5.28)$$

- Контраст – отношение между разностью и суммой максимальной и минимальной яркостей в ограничивающей прямоугольнике  $\Psi$ :

$$k_{m\Omega} = \frac{\max_{(r,c) \in \Psi} I_I(r,c) - \min_{(r,c) \in \Psi} I_I(r,c)}{\max_{(r,c) \in \Psi} I_I(r,c) + \min_{(r,c) \in \Psi} I_I(r,c)}. \quad (5.29)$$

- Коэффициент удлинения:

$$k_{e\Omega} = \sqrt{\frac{\mu_{20} + \mu_{02} + \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}{\mu_{20} + \mu_{02} - \sqrt{(\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2}}}. \quad (5.30)$$

где  $\mu_{pq}$  – центральные геометрические моменты второго порядка:

$$\mu_{pq} = \sum_{(r,c) \in \Omega} (r - r')^p (c - c')^q, \quad (5.31)$$

$$r' = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Omega} r, \quad (5.32)$$

$$c' = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Omega} c. \quad (5.33)$$

- Средние значения в цветовых каналах:

$$\overline{I_{I\Omega}} = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Omega} I_I(r,c), \quad (5.34)$$

$$\overline{I_{Cb\Omega}} = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Omega} I_{Cb}(r,c), \quad (5.35)$$

$$\overline{I_{Cr\Omega}} = \frac{1}{N_{\Omega}} \sum_{(r,c) \in \Omega} I_{Cr}(r,c). \quad (5.36)$$

### Коррекция детектированных областей

Коррекция классифицированных как «красные глаза» областей включает три этапа:

1. итеративное расширение областей [263] на соседние пиксели розовых и пурпурных оттенков, которые получаются при смешении красного с цветами склеры, ресниц, радужки глаза;
2. заливка «дыр» (fill holes) внутри связных областей;
3. понижение насыщенности, яркости и альфа-смешивание на краях области, чтобы обеспечить плавный переход от неизменённых пикселей изображения к скорректированным.

Значения в альфа-канале вычисляются в зависимости от количества соседних пикселей, относящихся к области  $\Omega$ ; если пиксель  $(r,c)$  находится в центре корректируемой области, то  $\alpha(r,c) = 1$ ; если с краю, то  $\alpha(r,c)$  близко к 0:

$$\alpha(r,c) = -1 + \frac{1}{8} \sum_{i=-1}^1 \sum_{j=-1}^1 1 | (r+i, c+j) \in \Omega. \quad (5.37)$$

В центре области обнуляется насыщенность и понижается яркость:

$$I_I(r,c) = (2^{D_l} - 1)(I_I(r,c)\alpha(r,c)/(2^{D_l} - 1))^2 + I_I(r,c)(1 - \alpha(r,c)), \quad (5.38)$$

$$I_{Cr}(r,c) = 2^{D_l-1}\alpha(r,c) + I_{Cr}(r,c)(1 - \alpha(r,c)), \quad (5.39)$$

$$I_{Cb}(r,c) = 2^{D_l-1}\alpha(r,c) + I_{Cb}(r,c)(1 - \alpha(r,c)), \quad (5.40)$$

где  $D_l$  – глубина цвета на канал.

### Критерий качества детектирования и коррекции «красных глаз»

В литературе вопросу количественной оценки качества детектирования и коррекции «красных глаз» уделяется незаслуженно мало внимания. Обычно качество коррекции не обсуждается, а при оценке обнаружения ограничиваются типичными для машинного обучения критериями качества, вычисляемыми из количества ложно положительных и ложно отрицательных [268], хотя одинаковые виды ошибок по-разному воспринимаются пользователем [269]. Автором выделены следующие семь факторов качества детектирования и исправления «красных глаз»:

1.  $FN_m$  – необнаружение явно заметного дефекта;
2.  $FN_d$  – необнаружение слабо заметного дефекта;
3.  $FP_c$  – ложное обнаружение, которое при коррекции заметно и портит фотографию;
4.  $FP_n$  – ложное обнаружение, которое при коррекции незаметно;
5.  $N_p$  – исправлен только один глаз из двух, что заметно и воспринимается пользователем негативно;
6.  $C_i$  – результат коррекции выглядит хуже, чем исходное изображение;
7.  $C_n$  – коррекция заметна, но результат приемлемый.

Для демонстрации факторов качества на рисунке 5.16 приведены два результата коррекции «красных глаз» с помощью web-сервиса Redbot компании HP [270]. На рисунке 5.16а один из двух явно заметных глаз не исправлен, заметные ложные срабатывания на губах. Такой результат коррекции негативно оценивается пользователем. На рисунке 5.16b один из двух слабо заметных глаз не исправлен, незаметная ложно положительная область на шарфе. Ошибки являются нежелательными, но не портят фотографию.

Предлагается сформировать общий критерий качества с использованием взвешенной суммы перечисленных семи факторов. Веса определяются с помощью метода анализа иерархий (МАИ) [45], в котором факторы попарно сравниваются

друг с другом по важности с точки зрения пользователя: одинаковая важность, или один важнее другого, или один значительно важнее другого. Сравнение факторов в процедуре МАИ и полученные веса приведены в таблице 5.7. Общий критерий качества для способа исправления «красных глаз»:

$$Q = \max\left(\frac{O_{re} - 1,3FN_m - 0,7FN_d - 3,6FP_c - 0,4FP_n - 1,6N_p - 2,1C_i - 0,3C_n}{O_{re}}, 0\right), \quad (5.41)$$

где  $O_{re}$  – количество «красных глаз»; веса из таблицы 5 умножены на 10.



a)



b)

Рисунок 5.16 – Примеры коррекции «красных глаз» сервисом Redbot [270].

Таблица 5.7. МАИ для оценки веса факторов качества детектирования и коррекции «красных глаз».

	$FN_m$	$FN_d$	$FP_c$	$FP_n$	$N_p$	$C_i$	$C_n$	Геом. ср.	Вес
$FN_m$	1,00	5,00	0,20	5,00	1,00	0,20	5,00	1,26	0,13
$FN_d$	0,20	1,00	0,33	5,00	0,20	0,20	5,00	0,68	0,07
$FP_c$	5,00	3,00	1,00	3,00	5,00	5,00	5,00	3,43	0,36
$FP_n$	0,20	0,20	0,33	1,00	0,20	0,20	1,00	0,34	0,04
$N_p$	1,00	5,00	0,20	5,00	1,00	1,00	5,00	1,58	0,16
$C_i$	5,00	5,00	0,20	5,00	1,00	1,00	5,00	1,99	0,21
$C_n$	0,20	0,20	0,20	1,00	0,20	0,20	1,00	0,32	0,03

Для определения факторов  $C_i$  и  $C_n$ , связанных с коррекцией, требуется мнение наблюдателя. Определение остальных пяти факторов можно сделать автоматически при условии специальной разметки обучающего набора. Параметры

AdaBoost классификатора (глубина деревьев, итоговый набор признаков) подбирались путём максимизации  $Q$  (без учёта коррекции) в ходе скользящего контроля по группам фотографий, снятых одинаковой моделью камеры.

### Результаты

На основе критерия  $Q$  проведено сравнение способов автоматической коррекции «красных глаз» на тестовой выборке из 50 фотографий, содержащих 152 дефектные области. Сравнивались: ПО Nikon View 6.2.7, в котором реализован алгоритм компании Fotonation [254, 271]; фотопринтер HP Photosmart 475 [272], web-сервис Redbot [270]; ПО Kodak Easy Share 6.01; Canon Easy-PhotoPrint 3.4; Microsoft Digital Image Starter Edition; ArcSoft Photoprinter 5.0; Cyberlink PhotoNow 1.0, использующий детектор лиц Виолы-Джонса [261]. Результаты сравнения приведены в таблице 5.8. Примеры коррекции предлагаемым способом, методом Fotonation и ПО ArcSoft Photoprinter показаны на рисунке 5.17.

Таблица 5.8. Сравнение способов автоматической коррекции «красных глаз».

	$FN_m$	$FN_d$	$FP_c$	$FP_n$	$N_p$	$C_i$	$C_n$	$Q$
Предлагаемый	2	21	0	2	13	5	13	<b>0.60</b>
Fotonation	7	21	0	3	12	4	13	0.56
Arcsoft	10	22	0	2	13	4	7	0.55
Canon	12	31	2	3	9	3	7	0.48
Kodak	24	39	1	1	2	5	4	0.47
HP Redbot	18	17	3	4	5	4	10	0.46
HP PS 475	11	27	4	9	14	3	6	0.25
Microsoft	23	31	6	6	15	15	25	0
Cyberlink	30	27	21	18	6	9	6	0

Детектирование «красных глаз» (этапы сегментации и классификации для фотографий, снятых со вспышкой) обеспечивают высокие показатели качества, традиционные для машинного обучения. На рисунке 5.18 приведена кривая ошибок (receiver operating characteristic, ROC) [273]. Площадь под кривой ошибок (area under curve, AUC) равна 0,986.



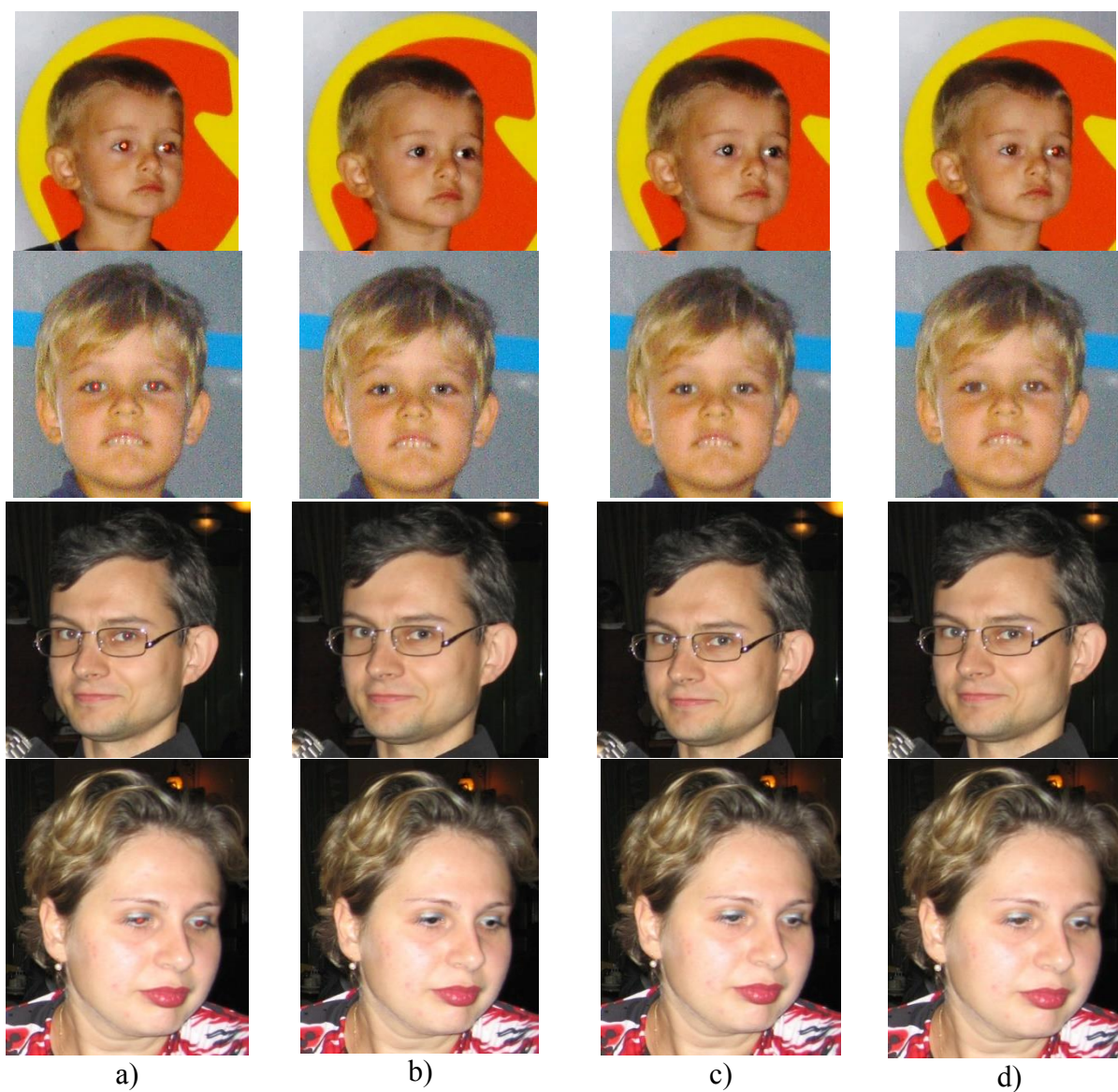


Рисунок 5.17 – Примеры коррекции «красных глаз»: а) исходные изображения; б) результаты предлагаемого способа; в) ArcSoft; г) Fotonation.

Процедура детектирования и коррекции «красных глаз» не оказывает заметное влияние на время печати. Этап масштабирования изображений всегда существует в конвейере обработки фотопринтера, поэтому на него не тратится дополнительного времени. Для остальных этапов обработки предлагаемого способа была выполнена эффективная реализация для встроенного ПО. Сегментация, разметка связных областей и отбрасывание ложно положительных

кандидатов выполняются в один проход по изображению на основе алгоритма вычисления геометрических признаков областей из [274]. Для сокращения времени выполнения классификации, веса и параметры пороговых классификаторов в узлах деревьев конвертируются из данных в исполняемый код. Для тестового набора среднее время работы на целевой платформе составило 1,2 с, что удовлетворяет требованиям к производительности.

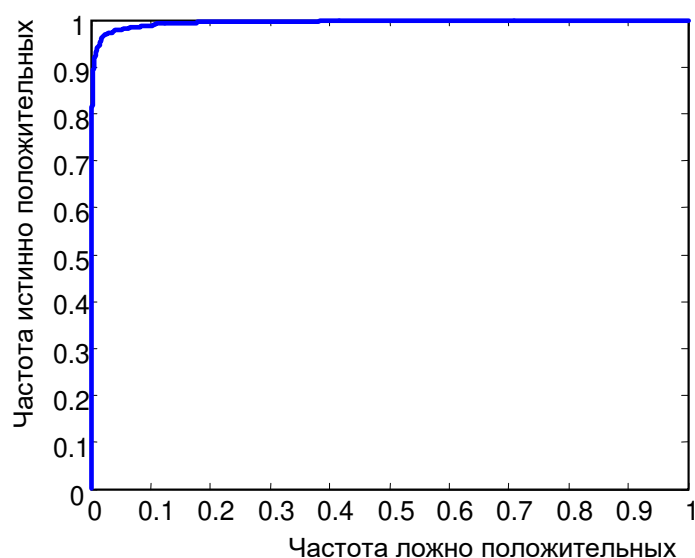


Рисунок 5.18 – Кривая ошибок (ROC) детектирования «красных глаз».

### 5.3.3. Оценка и коррекция затемнённых областей

#### Существующие подходы

Существует множество методов для улучшения яркости и контраста, но большинство из них предназначены для обработки целиком тёмных (low-light) и/или слабоконтрастных изображений. В случае съёмки с контровым освещением общий контраст изображения, как правило, высокий, но часть фотографии имеет тёмные оттенки и низкий контраст. В англоязычной литературе коррекция такого рода дефекта получила название *shadow enhancement* – улучшение затемнённых/затемнённых областей или *backlit image enhancement* – улучшение изображений с задней засветкой.

Многие способы улучшения локального контраста основаны на простейшей физической модели формирования изображения  $I$  как произведения диффузной отражательной способности (альбедо)  $R$  и освещённости сцены  $L$  [275]:

$$I(r, c) = R(r, c)L(r, c). \quad (5.42)$$

Схожую концепцию имеет теория цветового зрения ретинекс (retinex), которая утверждает, что “система сетчатки (retina) и коры (cortex) головного мозга может трактовать цвет как трёхэлементное сообщение от сетчатки, независимо от потока лучистой энергии, но коррелирующего с отражательной способностью объектов” [276]. Помимо того, что теория ретинекс используется для объяснения цветопостоянства (color constancy) человеческого зрения, на её основе был предложен метод обработки изображений также называемый ретинекс:

$$I'(r, c) = \sum_{m=1}^M \omega_m \log\left(\frac{I(r, c)}{I_{LPF_m}(r, c)}\right), \quad (5.43)$$

где  $M$  – число масштабов;  $\omega_m$  – веса, такие что  $\sum_{m=1}^M \omega_m = 1$ ;  $I_{LPF_m}$  – результат размытия  $I$  фильтром Гаусса с дисперсией  $\sigma_m^2$ .

Если  $M$  равно 1, то говорят об одномасштабном ретинексе (single scale retinex, SSR), иначе о многомасштабном ретинексе (multiscale retinex, MSR). Существует множество модификаций MSR, например [128, 277, 278], которые различаются подходами к обработке цветовых каналов и разрешению неопределённости в формуле (5.43) в случае равенства нулю аргумента логарифма. Действительно большинство реализаций MSR способно значительно улучшить контраст в затемнённых областях фотографий, но при этом искажаются светлые части изображения, возникают артефакты в виде ореолов вдоль контурных перепадов, искажения цветов [279]. Ретинекс, выполненный по формуле 5.43, помогает лучше увидеть детали слабоконтрастных изображений, но обработанные фотографии выглядят неестественно. Существенно более естественным получается результат при смешивании исходного изображения и результата MSR [280].

В работе [281] предлагается следующий способ улучшения затемнённых областей на изображении с каналом яркости  $I$ :

$$I'(r, c) = k_1 \frac{I(r, c)}{I_f(r, c) + 1} + k_2 I_f(r, c), \quad (5.44)$$

где  $I_f$  – результат фильтрации  $I$  сохраняющим границы фильтром, параметры которого зависят от оценки уровня шума;  $k_1$  – коэффициент усиления;  $k_2$  – коэффициент ослабления.

Результат фильтрации  $I_f$  рассматривается как оценка освещённости, отношение  $I/I_f$  – в качестве оценки отражательной способности. Этот способ хорошо повышает яркость и контраст в затемнённых областях, мало влияет на светлые области, не образует ореол вдоль перепадов яркости, но усиливает шумы в тёмных оттенках и ведёт к потере насыщенности, изображение становится блёклым. Выбор подходящих коэффициентов  $k_1$  и  $k_2$  выполняется пользователем вручную, что часто является нетривиальной задачей.

Адаптивная гамма-коррекция в зависимости от оценки освещённости предлагается в работе [129]. Цветовые RGB каналы  $I_l(r, c)$ , нормализованные в диапазон от 0 до 1, преобразуются по формуле:

$$I_l'(r, c) = I_l(r, c)^{2^{1-2I_{LPF}(r, c)}}, \quad (5.45)$$

где  $I_{LPF}$  – оценка освещённости, которая выполняется путём Гауссова размытия канала яркости.

В тёмных областях кривая преобразования уровней выпуклая, и они становятся светлее; в светлых областях кривая преобразования вогнутая и они становятся темнее. Локальный контраст как в тёмных, так и в светлых областях повышается, но часто недостаточно. При этом данный способ не позволяет регулировать величину усиления контраста в тёмных и светлых областях. Кроме того, его применение ведёт к заметному падению насыщенности.

Ряд преобразований для отображения на мониторах изображений расширенного динамического диапазона (high dynamic range, HDR) применим для

преобразования тонов изображений стандартного динамического диапазона (standard dynamic range, SDR) [282, 283]. Для коррекции затемнённых областей на фотографиях эти методы подбирают такую функцию преобразования уровней, что тёмные области становятся светлее, при этом часто возникают нежелательные искажения яркости и цвета в средних и светлых оттенках.

Общим недостатком существующих методов коррекции затемнённых областей является то, что они ухудшают качественные изображения, которые не нуждаются в коррекции. Необходимо отметить, что автору неизвестны способы автоматического детектирования фотографий, повреждённых контровым светом и оценки степени выраженности данного дефекта.

В последние годы появились методы на основе глубоких нейронных сетей, которые предназначены для коррекции изображений, повреждённых контровым светом. Применение методов машинного обучения для этой задачи ограничено отсутствием большого количества разнообразных размеченных данных, сложностью и неоднозначностью разметки. В работах [284, 285] используется набор специально снятых против света 3000 фотографий, которые получены с помощью только 4-х фотокамер двух производителей и содержат небольшой набор сцен. Фотографии скорректированы в ПО Adobe Lightroom. Таким образом, эти нейронные сети стремятся повторить результат коррекции, реализованный в ПО Adobe, но сделать это автоматически без участия пользователя. Заметим, что результат обработки этими нейронными сетями качественных изображений в литературе не обсуждается.

### **Алгоритм улучшения затемнённых областей**

В соответствии с подходом к оценке и улучшению качества печатаемых фотографий на первом этапе требуется проверить по метаинформации фотографии, что при съёмке не применялась вспышка (так как при вспышке нет контрового света) затем выполнить масштабирование изображения, исходя из разрешения печати и размера отпечатка.

Из анализа существующих способов коррекции затемнённых областей сделан вывод, что целесообразно использовать модель формирования изображения (5.42). Пусть есть альфа-канал  $\alpha_s$ , значения которого близки к 1 в светлых областях и к 0 в тёмных. Тогда повысить яркость и контраст в затемнённых областях канала яркости  $I$  можно путём смешивания с альфа-каналом  $\alpha_s$  исходного  $I$  и оценки отражательной способности  $I_a$ . Однако в  $I_a$  отсутствует освещённость сцены, и такое изображение выглядит неестественно. Поэтому предпочтительно смешать  $I_a$  с исходным каналом яркости  $I$ . Для нормализованного в диапазон от 0 до 1 канала яркости  $I$  формула улучшения затемнённых областей выглядит следующим образом:

$$\begin{aligned} I'(r, c) &= \alpha_s(r, c)I(r, c) + (1 - \alpha_s(r, c))(k_s I_a(r, c) + (1 - k_s)I(r, c)) = \\ &= I(r, c) + k_s(1 - \alpha_s(r, c))(I_a(r, c) - I(r, c)), \end{aligned} \quad (5.46)$$

где  $k_s$  – коэффициент коррекции затемнённых областей;  $k_s \in [0, 1]$ .

Оценку освещённости  $I_f$  в модели формирования изображения (5.42) предлагается выполнять с помощью модифицированного билатерального фильтра, вычисляемого по формуле (3.55), где весовая функция по диапазону яркостей  $\Phi(x)$  приведена в формуле (3.54). Такой подход, в отличие от использования гауссиана, не приводит к образованию заметного ореола вдоль контрастных перепадов. Оценка отражательной способности вычисляется как умноженное на коэффициент  $a$  отношение значений в канале яркости изображения и суммы оценки освещённости и  $b\sigma_r$ :

$$I_a(r, c) = \frac{aI(r, c)}{I_f(r, c) + b\sigma_r}, \quad (5.47)$$

где  $\sigma_r$  – величина перепада, которая считается «полезной» (т.е. не образованной шумом) границей между областями, и совпадает с  $\sigma_r$  в формуле (3.54);  $a$  и  $b$  – коэффициенты.

Предлагается использовать оценку освещённости  $I_f$  в качестве альфа-канала  $\alpha_s$ . Тогда формулы (5.46) и (5.47) преобразуются к виду:



$$I'(r, c) = I(r, c) + k_s I(r, c) \left(1 - I_f(r, c)\right) \max\left(0, \frac{a}{I_f(r, c) + b\sigma_R} - 1\right). \quad (5.48)$$



a)



b)



c)



d)

Рисунок 5.19 – Пример коррекции затемнённых областей: а) исходное изображение; б) результат обработки; в) оценка освещённости; д) оценка отражательной способности.

После коррекции канала яркости значения цветоразностных каналов вычисляются по формулам (5.1) – (5.2). На рисунке 5.19 показаны примеры оценки

освещённости и отражательной способности, результат коррекции затемнённых областей предлагаемым способом.

Величины  $a$ ,  $b$ ,  $\sigma_r$ , размер окна билатерального фильтра  $N_o$ , параметр  $\mu$  функции  $\Phi(x)$  (см. формулу (3.54)) зависят от технологии печати и модели принтера. Они выбираются экспериментально по отпечаткам исходя из следующих соображений: яркость и контраст в затемнённых областях увеличивается, но при этом чёрные области не должны светлеть, небольшие шумы не должны усиливаться, заметные перепады яркости между областями не должны размываться. Например, для фотопринтера Samsung SPP-2040 выбраны следующие параметры:  $a=1,3$ ;  $b=3$ ;  $\sigma_r=10$ ;  $\mu=3$ .

Сходный способ коррекции чрезмерно осветлённых областей фотографий предложен в [209]. Однако такой дефект встречается на фотографиях редко, поэтому такого рода коррекция не реализовывалась во встроенном ПО принтеров.

### **Определение коэффициента коррекции затемнённых областей**

Степень коррекции затемнённых областей регулируется единственным интуитивно понятным параметром  $k_s$ . При  $k_s=0$  коррекции не происходит; при  $k_s$  немного больше 0 слегка осветляются тёмные области; при  $k_s$  больше 0,5 тёмные области осветляются сильнее; при  $k_s$  близком к 1 выполняется максимальная коррекция, но осветлённые области часто выглядят неестественно, такие фотографии предпочтительнее считать испорченными и не печатать. Таким образом, в алгоритме автоматической коррекции затемнённых областей требуется определять подходящее значение  $k_s$ , причём для фотографий неповреждённых контровым светом  $k_s$  должно быть близко к 0, а для фатально испорченных фотографий близко к 1.

В работе [124] автором предложено дерево решений, способное на основе признаков, вычисляемых по гистограмме яркостей (см. формулу (1.4)), выдавать четыре дискретные значения  $k_s$ : 0; 0,3; 0,7 и 1. Пример гистограммы яркостей изображения, повреждённого контровым светом показан на рисунке 5.20. У



гистограммы таких изображений обычно наблюдаются пики в тёмных и светлых значениях яркости, тогда как в средних тонах есть провал.

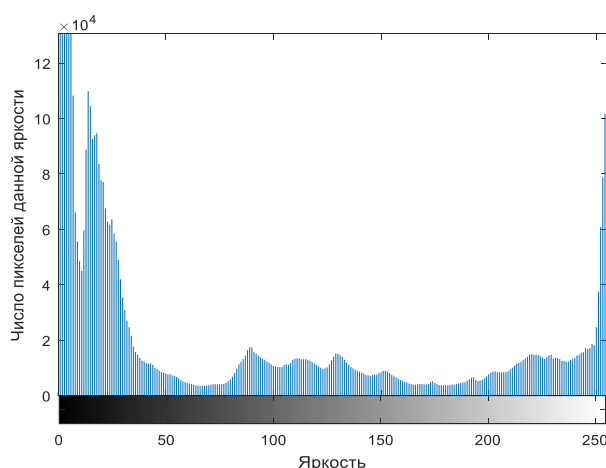


Рисунок 5.20 – Пример гистограммы яркостей изображения, повреждённого контровым светом.

Дерево решений из [124] хорошо определяет фотографии, повреждённые контровым светом, но нередко относит к ним и качественные изображения, имеющие похожую гистограмму яркостей. Дискретные значения  $k_s$  при коррекции часто не позволяют обеспечить наилучшее качество изображения. По этим причинам потребовалась разработка регрессионной модели для предсказания  $k_s$ . Для этого необходимо собрать репрезентативный набор изображений, предложить признаки и выбрать метод машинного обучения, имеющий компактную реализацию.

Для создания набора изображений была разработана программа, которая скачивала фотографии из социальных сетей vk.com и flickr.com, затем с помощью дерева решений [124] выбирала изображения, повреждённые контровым светом. Далее пользователи по формуле (5.48) корректировали затемнённые области, подбирая наилучшее значение  $k_s$ . Монитор был откалиброван для отображения цветов наиболее близко к цветам фотопринтера. Изображения масштабировались так, чтобы иметь размер на экране равный размеру отпечатка. Примерно каждая 50-я фотография печаталась для контроля соответствия визуального восприятия

изображения на мониторе и отпечатка. В набор изображений было добавлено 300 качественных изображений, для которых  $k_s=0$ . В результате был собран и размечен набор из 1500 изображений.

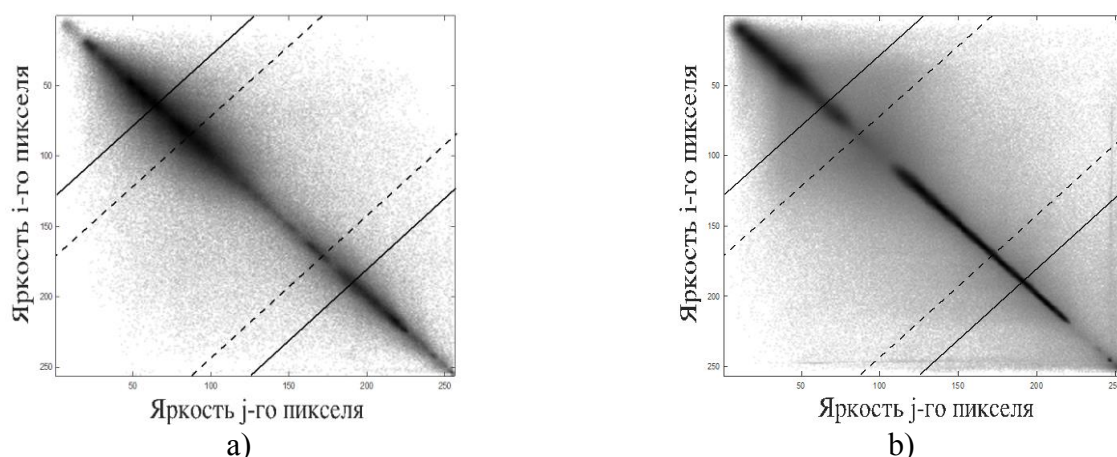


Рисунок 5.21 – Примеры матриц совместной встречаемости: а) качественное изображение; б) фотография, повреждённая контровым светом.

Матрицы совместной встречаемости изображений, повреждённых контровым светом, имеют характерный вид: в частности, наблюдается широкий пик в тёмных оттенках. Пример матриц совместной встречаемости для качественной и искажённой задней засветкой фотографий показан на рисунке 5.21. Предлагается вычислять признаки из нормализованной матрицы совместной встречаемости  $h_2'$ , вычисленной по формулам (1.5) – (1.6) для смещений  $s_r = 0, s_c = 3$  [233]. Поскольку выполняется обработка изображений с глубиной цвета на канал  $D_l = 8$ , то размер  $h_2'$  равен 256 на 256 элементов.

Область матрицы разбивается на три части линиями, проведёнными перпендикулярно главной диагонали. На рисунке 5.21 эти линии показаны сплошными. По матрице  $h_2'$  вычисляется 15 признаков  $F_k$ . Вычисляются суммы значений  $h_2'$  для каждой из частей матрицы:

$$F_1 = \sum_{i=0}^{127} \sum_{j=0}^{127-i} h_2'(i, j), \quad (5.49)$$

$$F_2 = \sum_{i=128}^{255} \sum_{j=0}^{i-127} h_2'(i, j), \quad (5.50)$$

$$F_3 = 1 - F_1 - F_2. \quad (5.51)$$

Та часть матрицы, которая расположена в левом верхнем углу и соответствует тёмным оттенкам также разбивается на две части перпендикулярно главной диагонали. Для каждой из этих частей вычисляется сумма значений  $h_2'$ :

$$F_4 = \sum_{i=0}^{63} \sum_{j=0}^{63-i} h_2'(i, j), \quad (5.52)$$

$$F_5 = F_1 - F_4. \quad (5.53)$$

Для левой верхней и правой нижней частей матрицы  $h_2'$  вычисляется отношение максимального значения в данной части к максимальному значению всей матрицы:

$$F_6 = \max_{\substack{i \in [0, 127] \\ j \in [0, 127-i]}} h_2'(i, j) / \max_{\substack{i \in [0, 255] \\ j \in [0, 255]}} h_2'(i, j), \quad (5.54)$$

$$F_7 = \max_{\substack{i \in [128, 255] \\ j \in [0, i-127]}} h_2'(i, j) / \max_{\substack{i \in [0, 255] \\ j \in [0, 255]}} h_2'(i, j). \quad (5.55)$$

Для левой верхней и правой нижней частей матрицы  $h_2'$  определяется номер столбца, в котором расположен максимальный элемент:

$$F_8 = \max_{i \in [0, 127]} ( \operatorname{argmax}_{j \in [0, 127-i]} h_2'(i, j) ), \quad (5.56)$$

$$F_9 = \max_{i \in [128, 255]} ( \operatorname{argmax}_{j \in [0, i-127]} h_2'(i, j) ). \quad (5.57)$$

Введём функцию  $\Lambda(x)$ , которая равна 1, когда её аргумент больше равен среднего значения в  $h_2'$ , и равна 0 в противном случае:

$$\Lambda(x) = \begin{cases} 1, & x \geq \frac{1}{256^2} \sum_{i=0}^{255} \sum_{j=0}^{255} h_2'(i, j), \\ 0, & \text{иначе.} \end{cases} \quad (5.58)$$

Для трёх частей матрицы вычисляются суммы значений  $\Lambda(x)$  и отношения сумм между собой:

$$F_{10} = \sum_{i=0}^{127} \sum_{j=0}^{127-i} \Lambda(h_2'(i, j)), \quad (5.59)$$

$$F_{11} = \sum_{i=0}^{127} \sum_{j=127-i}^{255} \Lambda(h_2'(i, j)) + \sum_{i=128}^{255} \sum_{j=0}^{255-i} \Lambda(h_2'(i, j)), \quad (5.60)$$

$$F_{12} = \sum_{i=128}^{255} \sum_{j=0}^{i-127} \Lambda(h_2'(i, j)), \quad (5.61)$$

$$F_{13} = F_{11}/F_{10}, \quad (5.62)$$

$$F_{14} = F_{11}/F_{12}, \quad (5.63)$$

$$F_{15} = F_{10}/F_{12}. \quad (5.64)$$

Для выбора метода построения регрессионной модели сравнивались: метод наименьших квадратов (МНК) [286], машина опорных векторов для регрессии (support vector regression, SVR) [287], дерево для классификации и регрессии (classification and regression tree, CART) [288], нейронная сеть обобщенной регрессии (general regression neural network, GRNN) [289], алгоритм «случайный лес» (random forest) [290]. В большинстве современных работ сравнение регрессионных моделей осуществляется по нескольким критериям [291]. Таблица 5.9 содержит сравнение эталонного и предсказанного  $k_s$  по средней абсолютной ошибке (mean absolute error, MAE), среднеквадратичной ошибке (mean squared error, MSE), медианной абсолютной ошибке (median absolute error, MedAE) и коэффициенту корреляции Пирсона  $c_r$ . Чем меньше ошибки, тем лучше модель. Чем ближе абсолютное значение коэффициента корреляции  $c_r$  к единице, тем лучше модель. Обучение выполнялось по признакам, вычисленным по формулам

(5.49) – (5.64). Значения критериев в таблице являются средними значениями, полученными в ходе процедуры двойного скользящего контроля по группам однотипных изображений (см. описание в разделе 3.4). Набор фотографий разделён на группы по производителю камеры. По всем рассмотренным критериям лучшие результаты работы демонстрирует алгоритм «случайный лес».

Таблица 5.9 Оценки качества работы моделей, построенных различными методами.

Метод	$MAE$	$MSE$	$MedAE$	$c_r$
МНК	15,2	350	13,3	0,59
SVR	15,1	377	12,1	0,54
CART	18,5	559	15,3	0,41
GRNN	15,0	344	12,9	0,59
Random forest	<b>13,8</b>	<b>290</b>	<b>11,8</b>	<b>0,67</b>

Дана оценка того, как предлагаемый набор признаков влияет на качество предсказания  $k_s$ . Для этого «случайный лес» обучается с помощью признаков из [124], вычисляемых из гистограммы яркостей, и вычисляемых из матрицы совместной встречаемости  $h_2'$  аналогично формулам (5.49) – (5.64), но матрица разделяется на три части как показано на рисунке 5.21 штриховыми линиями. Модель, обученная с использованием предложенного набора признаков, имеет наилучшее качество по рассмотренным критериям (см. таблицу 5.10).

Таблица 5.10 Оценки качества работы модели Random forest, обученной на различных наборах признаков.

Набор признаков	$MAE$	$MSE$	$MedAE$	$c_r$
Из гистограммы яркости [124]	14,8	334	13,1	0,60
Из $h_2'$ , на рис. 5.21 части разделены сплошными линиями	<b>13,8</b>	<b>290</b>	<b>11,8</b>	<b>0,67</b>
Из $h_2'$ , на рис. 5.21 части разделены штриховыми линиями	14,0	297	12,1	0,66

## Результаты

Коэффициент коррекции затемнённых областей  $k_s$  можно рассматривать как оценку качества изображений, повреждённых контрольным светом: чем больше  $k_s$ , тем хуже качество фотографии. Предлагаемая регрессионная модель предсказания  $k_s$  применяется для оценки качества изображений. Сравним эту модель с известными универсальными критериями качества изображений BIQI [292], BRISQUE [293], NIQE [294], ILNIQE [295], OG-IQA [296] для оценки 300 фотографий тестовой выборки. Таблица 5.11 содержит абсолютное значение коэффициента корреляции Пирсона  $c_r$  между размеченными значениями  $k_s$  и оценками универсальных критериев качества, а также размеченными и предсказанными  $k_s$  предлагаемой моделью и деревом решений из [124]. Из таблицы можно видеть, что универсальные критерии качества изображений не способны адекватно оценивать качество фотографий, повреждённых контрольным светом, тогда как предлагаемая модель демонстрирует высокую корреляцию с результатами разметки человеком. Предсказанное значение  $k_s$  используется для автоматической коррекции и предотвращения печати испорченных фотографий.

Таблица 5.11 Абсолютное значение коэффициента корреляции Пирсона  $c_r$  для универсальных критериев качества фотографий и предлагаемой регрессионной модели.

	BIQI	BRISQUE	NIQE	ILNIQE	OG-IQA	Дерево из [124]	Предлагаемая модель
$ c_r $	-0,22	0,05	0,13	0,23	0,13	0,47	<b>0,69</b>

Существующие способы коррекции затемнённых областей помимо полезного увеличения яркости и контраста в теневых областях часто добавляют на изображения нежелательные артефакты, искажают цвет. Для оценки присутствия и величины искажений предлагается тестовая таблица, показанная на рисунке 5.22. Тестовая таблица содержит следующие элементы:

1. Затемнённая область содержит аддитивный белый гауссов шум. Используется для оценки усиления уровня шума.
2. Цветной темный квадрат. Используется для оценки искажения цвета.
3. Область плавного изменения оттенков серого. Используется для обнаружения эффекта постеризации («cartoon») на градиентных заливках.
4. Равномерно чёрный квадрат на сером фоне. Используется для оценки искажения черного цвета в незатемнённой области и образования ореола вдоль контрастных перепадов.
5. Чёрный квадрат с добавлением аддитивного белого гауссова шума. Используется для оценки искажения черного цвета в затемнённой области.
6. Равномерные цветные квадраты с яркостью из середины диапазона тонов. Используются для оценки изменения цвета в середине диапазона тонов.

Сравнивались результаты обработки тестовой таблицы с помощью следующего ПО, предназначенного для улучшения затемнённых областей: функция Adobe Photoshop CS Shadow/Highlight; подключаемые модули для Adobe Photoshop: Kodak Digital SHO professional 2.0, Chroma Shadow Control 1.3, FixerLabs Shadow Fixer 1.1.7, IntrigueTek ShadowIlluminator [281]; реализация Retinex в TruView PhotoFlair 2.2 [128]. Параметры методов коррекции устанавливались таким образом, чтобы увеличить среднюю яркость в области 1 примерно в два раза. Количественные результаты сравнения приведены в таблице 5.12.

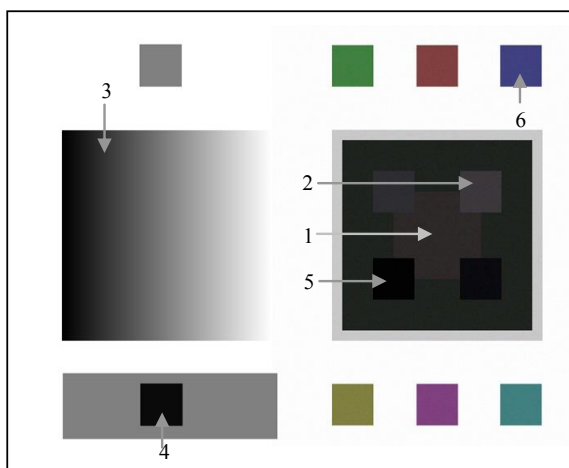


Рисунок 5.22 – Тестовая таблица для оценки методов улучшения затемнённых областей.

Таблица 5.12 Сравнение работы способов коррекции затемнённых областей на тестовой таблице.

ПО	С.к.о. в области 1	Насыщенность в области 2	Средняя яркость в области 5	Средняя яркость в области 6
IntrigueTek	7,9	0,07	3,6	90
Fixer Labs	6,6	0,06	11,5	110
Photo Flair	17,7	0,12	4,2	<b>85</b>
Chroma	6,4	0,06	11,8	<b>85</b>
Kodak SHO	9,3	0,08	4,0	116
Adobe PS CS	8,1	0,07	4,4	92
Предлагаемый способ	<b>5,9</b>	<b>0,08</b>	<b>2,2</b>	90
Исходное изображение	5,8	0,08	2,2	85

У всех рассматриваемых способов помимо увеличения средней яркости в области 1 вырос и уровень шума. Только в предлагаемом способе он вырос незначительно. Коррекция рядом способов привела к изменению насыщенности в области 2. Предлагаемый способ не изменяет насыщенность. Все рассматриваемые способы кроме предлагаемого осветлили практически чёрную область 5. Область 6, которая не нуждалась в коррекции, стала светлее в результате обработки большинством способов. Предлагаемый способ незначительно повысил яркость в средних тонах. Кроме того, все рассмотренные способы, кроме предлагаемого и IntrigueTek ShadowIlluminator привели к образованию заметных полос на плавной градиентной заливке 3. Заметный ореол возникает вдоль контрастных границ в результате применения Adobe Photoshop CS Shadow/Highlight, Nikon D-lighting и Photo Flair. Таким образом, применение предлагаемого способа улучшения затемнённых областей обеспечивает наименьшее количество нежелательных эффектов. Искажения Chroma Shadow Control, FixerLabs Shadow Fixer и Photo Flair самые заметные, поэтому далее эти программы не рассматриваются.



Для оценки пользователями качества коррекции затемнённых областей пять фотографий, повреждённых контровым светом, обрабатывались с помощью Adobe Photoshop CS Shadow/Highlight, Kodak Digital SHO professional, IntrigueTek ShadowIlluminator и предлагаемым способом. Использовались пробные версии Digital SHO и IntrigueTek ShadowIlluminator, которые добавляли на изображение надписи. Во всех случаях использовался либо автоматический подбор параметров, если такая возможность предоставлялась, либо параметры по умолчанию. Обработанные и исходные изображения печатались на фотопринтере Samsung SPP-2040. В опросе участвовало 24 человека, которые были проинструктированы для каждой из пяти фотографий ранжировать пять напечатанных фотокарточек (4 способа коррекции и необработанное) в порядке убывания качества. Интервьюируемые не знали способ обработки. Самое качественное изображение получало оценку  $E$  в 5 баллов, следующее 4, и так далее до 1. Исходное и улучшенные изображения для одной из пяти использовавшихся в опросе фотографий показаны на рисунке 5.23.

В таблице 5.13 приведены средние оценки  $\bar{E}$  по всем участникам опроса и пяти фотографиям. Все участники опроса отметили исходное изображение как наименее качественное. Следующими по качеству выбирались либо результат Adobe Photoshop CS, либо результат Kodak Digital SHO. Наилучшим по качеству выбирались результаты либо предлагаемого способа, либо IntrigueTek ShadowIlluminator. Напечатанный результат обработки фотографии предлагаемым способом чаще других отмечается как наиболее качественный.

Время оценки и коррекции затемнённых областей изображений на целевой платформе измерялось с помощью эмулятора и составило около 1,1 с.

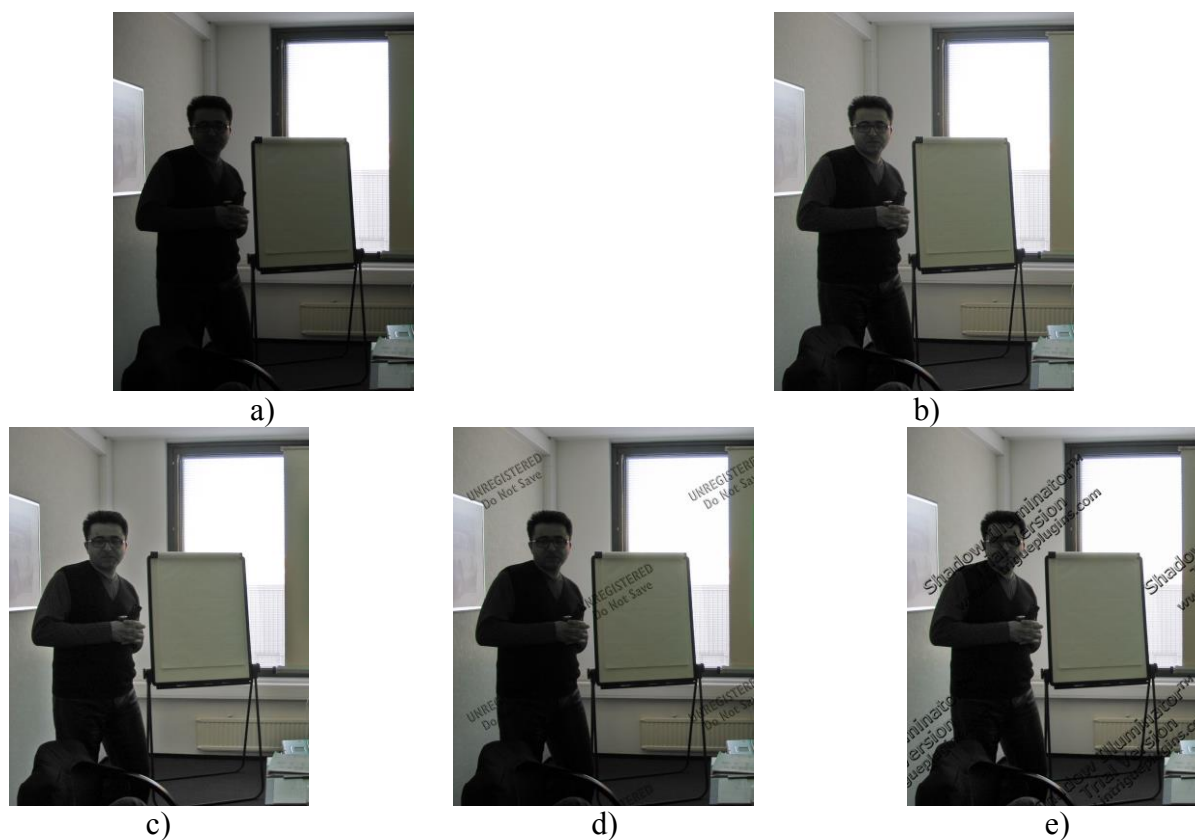


Рисунок 5.23 – а) исходная фотография; б) обработка предлагаемым способом; с) Adobe PS CS; д) Kodak SHO; е) IntrigueTek.

Таблица 5.13. Средние оценки способов улучшения затемнённых областей.

Способ	$\bar{E}$
Исходная фотография	1,0
Adobe PS CS	3,1
Kodak SHO	2,9
IntrigueTek	3,9
Предлагаемый способ	<b>4,1</b>

#### 5.4. Эффективность методов автоматической оценки и улучшения качества печатаемых фотографий

Для оценки эффективности разработанных методов автоматической оценки и улучшения качества печатаемых фотографий был проведён эксперимент, целью

которого был анализ типичной ситуации, когда пользователь печатает набор фотографий, включающий как качественные изображения, так и изображения, нуждающиеся в коррекции, а также испорченные фотографии. Тридцать изображений были случайным образом выбраны из 1000 фотографий, где примерно половина имела хотя бы один из дефектов, перечисленных в разделе 5.1. Все фотографии были сняты компактными цифровыми камерами, ранее не обрабатывались, хранились в файлах JPEG, содержащих EXIF информацию. Пять пользователей получили следующие инструкции:

1. на собственном ПК и мониторе просмотреть эти фотографии и исключить из рассмотрения испорченные, коррекция которых до приемлемого качества затруднительна;
2. улучшить с помощью Adobe Photoshop CS2 нуждающиеся в коррекции изображения, причём тратить на каждое изображение не более нескольких минут.

Этот же набор фотографий обрабатывался с помощью разработанного режима автоматической оценки и улучшения качества изображений. Помимо рассмотренных в этой главе методов повышения резкости, исправления эффекта «красных глаз» и коррекции затемнённых областей, режим включает способы коррекции артефактов JPEG сжатия и подавления шумов [209]. Также в главе 7 рассматривается алгоритм определения испорченных JPEG-сжатием изображений.

Фиксировалось количество фотографий, оцененных как испорченные, и время обработки изображений пользователями, а также предлагаемым режимом. Далее на фотопринтере Samsung SPP-2040 печатались только те фотографии, которые никто не отнёс к испорченным. Размер напечатанной фотокарточки 15 на 10 см. Полученные 6 отпечатков (пять, полученных от пользователей, и один от предлагаемого автоматического режима) каждой из фотографий ранжировались пользователями в порядке убывания качества. Самое качественное изображение получало оценку  $E$  в 5 баллов, следующее 4, и так далее до 0. Интервьюируемые не

знали кем обрабатывалось изображение для фотокарточек. Для всех напечатанных фотографий и всех пользователей вычислялся средний балл мнений (MOS)  $\bar{E}$ .

Таблица 5.14 содержит результаты обработки набора фотографий предлагаемым режимом и пользователями. Пользователи и режим отметили от 3 до 6 фотографий из набора как испорченные, которые не следует печатать. В множество из шести фотографий, выбранном пользователем № 4, входят все изображения, выбранные другими пользователями. Очевидно, что часто нет чёткой границы между безнадежно испорченной фотографией и изображением, которое можно улучшить до приемлемого качества. Тем не менее, предположим, что наиболее требовательный пользователь прав, и шесть изображений не стоит печатать. Таким образом, предлагаемые методы оценки качества позволяют предотвратить печать не менее 67% испорченных фотографий.

Все методы предлагаемого режима оценивают качество и улучшают изображения из набора, если это требуется, за 178 с. Для замеров времени использовался программный эмулятор встроенного ПО фотопринтера. Анализ качества и обработка изображений пользователями требует в 10–14 раз больше времени.

Таблица 5.14 Сравнение подготовки к печати набора фотографий.

Кто обрабатывал	Количество оцененных как испорченные	Время подготовки набора к печати, с	$\bar{E}$
Пользователь 1	5	1850	2,0
Пользователь 2	3	2420	2,5
Пользователь 3	5	2045	2,2
Пользователь 4	6	1900	1,8
Пользователь 5	3	2615	3,1
Предлагаемый режим	4	<b>178</b>	<b>3,4</b>

Для сравнения эффекта от улучшения изображений 6 отпечатков каждой из 24 фотографий ранжировались по качеству, шесть фотографий были исключены

как испорченные. Надо сказать, что примерно половина фотографий набора была хорошего качества и не требовала дополнительной коррекции, четверть достаточно было слегка исправить, и только оставшаяся четверть требовала существенной коррекции. Таким образом, во многих случаях несколько фотокарточек были практически одинаковыми (иногда все шесть) и ранжировались случайным образом. Поэтому разница между средним баллом мнений  $\bar{E}$  невелика, а с.к.о. оценок равно примерно 1,5 баллам для каждого из пользователей. Тем не менее средний балл мнений  $\bar{E}$  выше у предлагаемого режима автоматического улучшения качества фотографий. Тщательный анализ результатов ранжирования позволил сделать следующие выводы:

- для изображений, требующих коррекции, предлагаемый режим, как правило, получал оценку выше средней, так как у пользователей, которые ориентировались на изображение на мониторе и исходный размер фотографии, часто не получалось адекватно улучшить изображение, возникали артефакты, изменялись цвета;
- некоторые пользователи пытались улучшить фотографии хорошего качества, редко это положительно оценивалось наблюдателями, чаще негативно; предлагаемый режим оставлял качественные фотографии без изменений, что при случайном выборе приводило к тому, что фотокарточки, полученные в результате обработки предлагаемым режимом, редко получали самые низкие оценки, что ведёт к росту среднего значения.

Таким образом, применение предлагаемых способов оценки и улучшения качества печатаемых изображений ведёт к повышению качества печатаемых изображений и значительному сокращению времени на подготовку фотографий к печати.

### Выводы по пятой главе

Проанализированы характерные дефекты цифровых фотографий, требующие автоматической коррекции при печати. Чаще других требуют улучшения: размытые изображения; фотографии с эффектом «красных глаз»; фотографии с затемнением всей сцены или её отдельных частей; изображения, повреждённые артефактами JPEG компрессии и высоким уровнем шума. Сформулированы требования к алгоритмам оценки качества изображений и исправления обнаруженных дефектов.

Предложен подход к автоматическому улучшению качества печатаемых фотографий, который рекомендует:

1. учитывать детали изображения, видимые для данного размера фотокарточки, разрешения и технологии печати;
2. выполнять оценку характерных дефектов фотографий на основе машинного обучения по визуальным данным изображений и метаданной;
3. выполнять обработку в цветовом пространстве входных данных без конвертации в промежуточные представления.

На основе предложенного подхода разработаны новые методы автоматической оценки и улучшения качества фотографий перед их печатью: оценка и повышение резкости изображений, детектирование и исправление эффекта «красных глаз», коррекция затемнённых областей на фотографиях, повреждённых контрольным светом. Предлагаемые методы оценки и улучшения качества фотографий превосходят существующие способы.

Описанные в главе методы реализованы в режиме автоматической оценки и улучшения качества фотографий. Режим предназначен для встроенного ПО устройств печати и позволяет по меньшей мере в 10 раз сократить время подготовки фотографий к печати, повысить качество напечатанных изображений и предотвратить печать не менее 67% испорченных фотографий.

## **Глава 6. Преобразования для изменения размеров и соотношения сторон документов и фотографий**

При печати и копировании документов на бумагу, размер которой отличается от размера исходного документа, либо выполняется масштабирование содержимого документа, либо изображение обрезается (выполняется кадрирование), либо увеличиваются поля. Причем это происходит даже при преобразованиях, когда размеры бумаги отличаются всего на несколько процентов. Например, при копировании документа размера A4 на бумагу Letter в режиме «Fit to Page», чтобы согласовать разницу в высоте страницы, изображение уменьшается примерно на 6%, то есть по 3% строк пикселей отбрасывается сверху и снизу изображения. При обрезке часть важной информации, например, номер страницы, может пропасть. При масштабировании копируемого документа качество ухудшается. Необходимо разработать такие способы преобразования размеров изображений документов, при котором удаляются малозначимые фоновые области, а основные смысловые элементы документа не модифицируются и не изменяют своих размеров.

Схожая задача возникает при печати фотографий без полей (borderless) на фотопринтерах, имеющих фиксированный размер отпечатка. Например, термосублимационный фотопринтер Samsung SPP-2040 [249] печатает на бумаге 15 на 10 см. Для печати без полей соотношение сторон фотографии должно быть 3:2, тогда как многие фотокамеры снимают изображения с иным соотношением сторон: 4:3, 16:9 и т. п. Для получения при печати требуемого соотношения сторон такие фотографии автоматически обрезаются, что иногда приводит к существенному ухудшению композиции, потере важных деталей. На рисунке 6.1 показан пример стандартной обрезки фотографии при печати без полей. Лицо на фотографии обрезается снизу и сверху, что отрицательно оценивается пользователями.



Рисунок 6.1 – Пример автоматической обрезки фотографии при печати без полей.

Требуется разработка преобразований, позволяющих изменять соотношение сторон фотографии без ухудшения визуального восприятия. Предпочтительна реализация подобных преобразований изображений во встроенном программном обеспечении систем сканирования и печати, что с учётом доступных вычислительных ресурсов требует разработки вычислительно простых и быстрых алгоритмов.

### 6.1. Существующие подходы

Ряд публикаций описывает методы автоматического кадрирования (cropping), то есть вырезания прямоугольного фрагмента изображения. Большинство из них предназначено для показа наиболее важной области изображения на небольшом дисплее или улучшения композиции фотографий за счёт исключения малозначимых полос по краям изображения. Как правило, в существующих методах неявно [297, 298] или явно [299, 300] строится карта значимости (saliency map), и находится прямоугольник, включающий в себя области с высокими значениями карты значимости. Несмотря на различия в алгоритмах, все они стремятся отрезать фоновые области с плавным изменением тонов, при этом оставить на фотографии области с резкими изменениями цветов и изображения людей, особенно лиц. В настоящее время методы автоматического



кадрирования в основном разрабатываются на основе глубоких нейронных сетей. Несмотря на усилия исследователей остаётся много нерешённых проблем, связанных с неоднозначностью оценки результатов кадрирования и созданием больших размеченных наборов данных [301].

Преобразования, отличающиеся от масштабирования и кадрирования, предназначенные для показа изображения на экране или печати на бумаге с иным соотношением сторон чем у исходного изображения, в англоязычной литературе объединяются под названием *retargeting*. Будем называть подобные операции *переформатированием*. Наиболее распространёнными подходами к переформатированию растровых изображений являются деформация по неравномерной сетке (*warping*) [302] и «резьба по шву» (*seam carving*) [303], в которой с помощью динамического программирования [304] путём минимизации суммы значений пикселей по карте стоимости определяется оптимальная связанная цепочка пикселей между противоположными сторонами изображения, а затем пиксели цепочки удаляются или дублируются. В [303] карта стоимости вычисляется как абсолютное значение результата применения ФВЧ к каналу яркости изображения.

Рисунок 6.2 поясняет работу переформатирования. На верхнем изображении чёрными кружочками показана цепочка пикселей от верхней до нижней строки, сумма пикселей карты стоимости, для которой является наименьшей по сравнению со всеми другими возможными цепочками. Пиксели слева и справа от цепочки отмечены серым цветом. Цепочка пикселей может быть удалена (рисунок 6.2a), что уменьшает количество столбцов на единицу. Рядом (для определённости справа) с каждым пикселем цепочки с помощью интерполяции может быть добавлен новый пиксель (рисунок 6.2b), что на единицу увеличивает количество столбцов. Итеративное применение таких операций к строкам или столбцам изображения позволяет добиться необходимого соотношения сторон.

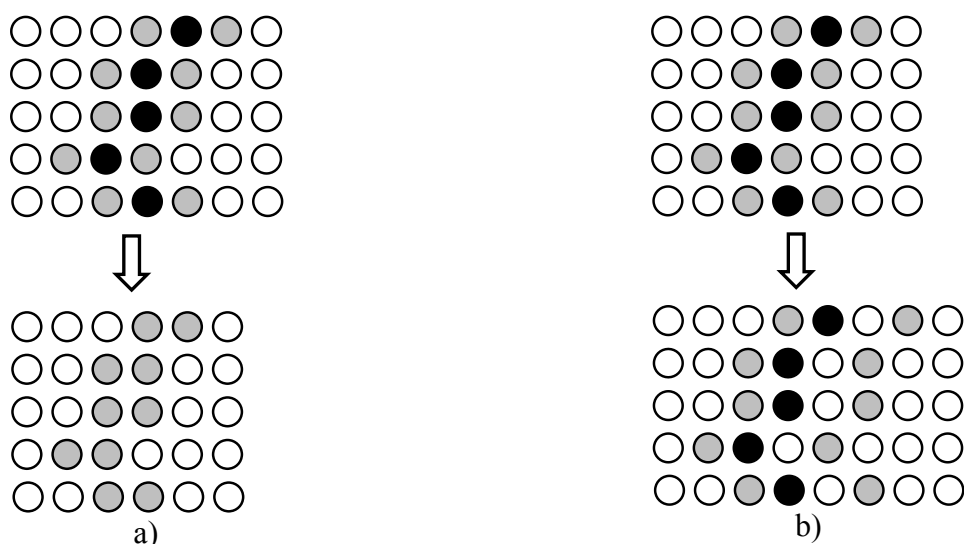


Рисунок 6.2— Цепочка пикселей в алгоритме seam carving: а) удаление пикселей цепочки; б) добавление пикселей справа от пикселей цепочки.

Переформатирование, описанное в [303], хорошо работает для фотографий природы, но часто искажает изображения людей и искусственных объектов, например строений. Для преодоления недостатков алгоритма [303] предложено множество его модификаций, например, увеличивается стоимость прямых линий [305], в [306, 307] карта значимости используется в качестве карты стоимости.

## 6.2. Преобразования для изменения соотношения сторон

Можно выделить два типа «заметных» участков изображений, модификация которых нежелательна:

- на фрагменте изображения присутствует лицо, тело человека или его часть;
- на участке изображения есть протяжённые резкие изменения яркости.

В настоящее время достигнут значительный прогресс в алгоритмах детектирования лиц человека [308]. Однако их преимущество в качестве детектирования нивелируется сложностью реализации во встроенном программном обеспечении систем сканирования и печати. Алгоритм Виолы-Джонса [261] позволяет достигать высокой производительности, был многократно

реализован для различных аппаратно-программных платформ, обеспечивает приемлемое качество детектирования лиц. Недостатком этого алгоритма является большее количество ложных срабатываний по сравнению с современными подходами. В работе [309] предлагается способ уменьшения ложных срабатываний для цветных изображений за счёт определения в детектированных зонах наличия пикселей с оттенками кожи человека. Наконец, вычислительно самым простым способом обнаружения на фотографии людей является детектирование областей с оттенками кожи человека [310]. Очевидно, что такой подход ведёт к большому количеству ложных срабатываний, однако для данной задачи вполне применим в случае, когда более вычислительно сложный способ не может быть использован из-за ограничений на допустимое время обработки.

Для выделения контурных перепадов предлагается использовать свёртку канала яркости  $I_I$  с ядром лапласиана гауссиана  $\omega_{lg}$  или разницы гауссиан  $\omega_{dg}$  [78], которые менее чувствительны к шуму на изображении, чем такие ФВЧ, как градиенты и лапласианы.

Пусть для нахождения областей с изображением людей используется поиск пикселей, имеющих оттенок кожи человека. Тогда карта заметности вычисляется как:

$$I_v = |I_I * \omega_{lg}| + kI_{st}, \quad (6.1)$$

где  $k$  – положительный коэффициент;  $I_{st}$  – бинарное изображение, в котором  $I_{st}(r, c)$  равно 1 в случае истинности логического выражения [310]:

$$\frac{I_b}{I_g} < 1,249 \wedge \frac{1}{3I_r} > 0,696 \wedge \frac{1-3I_b}{3} > 0,014 \wedge \frac{I_g}{3} < 0,108, \quad (6.2)$$

где

$$I_r(r, c) = \frac{I_R(r, c)}{I_R(r, c) + I_G(r, c) + I_B(r, c)}, \quad (6.3)$$

$$I_g(r, c) = \frac{I_G(r, c)}{I_R(r, c) + I_G(r, c) + I_B(r, c)}, \quad (6.4)$$

$$I_b(r, c) = \frac{I_B(r, c)}{I_R(r, c) + I_G(r, c) + I_B(r, c)} . \quad (6.5)$$

Для изменения размеров и соотношения сторон изображений документов и фотографий предлагаются следующие преобразования, основанные на анализе карты заметности:

1. обрезка полос изображения с двух противоположных сторон, где размеры полос выбираются путём минимизации в обрезаемых областях суммы пикселей, которые оцениваются как заметные [209];
2. переформатирование изображений за счёт удаления или добавления малозаметных участков по всему полю изображения [311, 312, 313];
3. дополнение изображений симметрично отраженными полосами с двух противоположных сторон, где размеры полос выбираются путём оценки заметности отражения [314].

Заметим, что перечисленные преобразования можно комбинировать как друг с другом, так и с масштабированием. Помимо использования этих преобразований при печати фотографий и копировании документов, они применимы для модификации изображений при их показе на дисплее МФП [315].

Существуют такие изображения, размеры и соотношение сторон которых невозможно изменить предлагаемыми преобразованиями так, что модификация будет незаметна, и визуальное качество не пострадает. Поэтому применимость того или иного метода преобразования оценивается по карте заметности. Альтернативным вариантом является предоставление пользователю возможности выбрать понравившийся результат преобразования размеров и соотношения сторон из нескольких возможных.

### **Обрезка полос изображения**

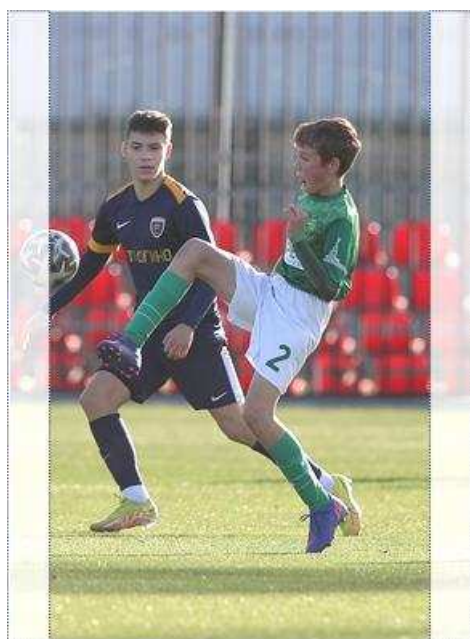
Без ограничения общности предположим, что для получения требуемого соотношения сторон необходимо уменьшить ширину изображения на  $N_c$  столбцов, то есть отрезать с левого края изображения  $x$  столбцов, а с правого края  $N_c - x$

столбцов. Значение  $x$  определяется с помощью минимизации суммы карты заметности  $I_v$  в отрезаемых полосах:

$$x = \underset{x}{\operatorname{argmin}} \left( \sum_{r=0}^{H-1} \sum_{c=0}^x I_v(r, c) + \sum_{r=0}^{H-1} \sum_{c=W-1-N_c-x}^{W-1} I_v(r, c) \right), \quad (6.6)$$

где  $H$  – высота исходного изображения;  $W$  – ширина исходного изображения.

Если среднее из значений карты заметности в отрезаемых полосах превышает предопределённый порог  $T_c$ , то делается вывод, что данное преобразование ведёт к ухудшению изображения, и следует исследовать возможность применения иных преобразований.



а)



б)

Рисунок 6.3– Пример обрезки полос: а) по умолчанию с противоположных сторон обрезаются полосы одинакового размера; б) результат предлагаемого преобразования.

На рисунке 6.3 показан пример обрезки полос. В случае отрезания с противоположных сторон одинакового количества столбцов (рисунок 6.3а) существенно портится композиция фотографии. Заметим, что такое преобразование используется по умолчанию в фотопринтерах при печати без

полей. Предлагаемое преобразование (рисунок 6.3b) отрезает фоновую область с правого края фотографии и улучшает композицию.

### Переформатирование изображений

Для переформатирования изображений целесообразно применять подход «seam carving», где в качестве карты стоимости использовать  $I_v$ . Классический алгоритм из [303] часто находит оптимальные (имеющие минимальную суммарную стоимость) цепочки пикселей на последовательных итерациях примерно в одном и том же месте. Это может приводить к тому, что модификация изображения путём удаления или добавления цепочек пикселей будет заметна. Предпочтительно находить цепочки пикселей в разных местах изображения. Этого можно добиться, если заносить значение  $k_h$  в координаты пикселей цепочки, найденной на итерации  $j$ , в карту истории  $I_{h_j}$  и поиск следующей оптимальной цепочки осуществлять по сумме  $I_v + I_{h_j}$ . При удалении пикселей цепочки из изображения на карту истории добавляются все соседние пиксели цепочки, при добавлении – пиксели добавленной цепочки и все соседние с ними. В результате маловероятно, что новая оптимальная цепочка пикселей будет пересекать цепочки, найденные на предыдущих итерациях, скорее она будет найдена где-то в другой части изображения. Однако, если только добавлять значения в карту истории, то после нескольких десятков итераций цепочки в карте истории начинают играть роль барьеров, которые сложно преодолеть при поиске оптимальной цепочки, проходящей через небольшие значения  $I_v$ . Поэтому вводится механизм релаксации карты истории, при котором на каждой следующей итерации значения в карте истории уменьшаются на константу  $k_r$ ,  $k_r \ll k_h$ :

$$I_{h_j}(r, c) = \max(0, I_{h_{j-1}}(r, c) - k_r). \quad (6.7)$$

Если среднее из значений карты заметности  $I_v$  по пикселям оптимальных цепочек превышает предопределённый порог  $T_r$ , то делается вывод, что

переформатирование заметно и ведёт к ухудшению изображения. В этом случае предпочтительно не выполнять переформатирование.

Схема предлагаемого способа переформатирования изображений показана на рисунке 6.4. Без потери общности опишем увеличение или уменьшение количества столбцов в изображении без изменения числа строк. На шаге 1 происходит инициализация: вычисляется исходная карта заметности, обнуляется карта истории, обнуляется сумма значений заметности пикселей цепочек. На шаге 2 с помощью динамического программирования выполняется поиск оптимальной цепочки пикселей между верхней и нижней сторонами изображения по сумме карт заметности и истории. На шаге 3 увеличивается сумма значений заметности пикселей цепочек путём добавления в неё значений заметности текущей цепочки пикселей. Условие 4 сравнивает среднее из значений карты заметности из пикселей цепочек с порогом  $T_r$ . Если порог превышен, то предполагается, что переформатирование заметно и портит изображение, поэтому от него лучше отказаться. На шаге 5 выполняется модификация изображения, в зависимости от необходимости уменьшить или увеличить количество столбцов. Соответственно, из изображения либо удаляется цепочка пикселей, либо в каждой строке изображения с помощью интерполяции добавляется пиксель справа (для определённости) от пикселя цепочки. На шаге 6 обновляется карта заметности: необходимо установить её размер в соответствии с текущим размером изображения и вычислить значения заметности в пикселях, соседних с удалёнными, или в добавленных и соседних с ними. На шаге 7 обновляется карта истории: необходимо установить её размер в соответствии с текущим размером изображения, добавить на неё пиксели текущей оптимальной цепочки, выполнить процедуру релаксации по формуле 6.7. Условие 8 проверяет, все ли итерации выполнены, то есть достигла ли ширина изображения заданного значения. Если условие ложно, то выполняется переход на шаг 2.



Рисунок 6.4 – Схема предлагаемого способа переформатирования изображений.



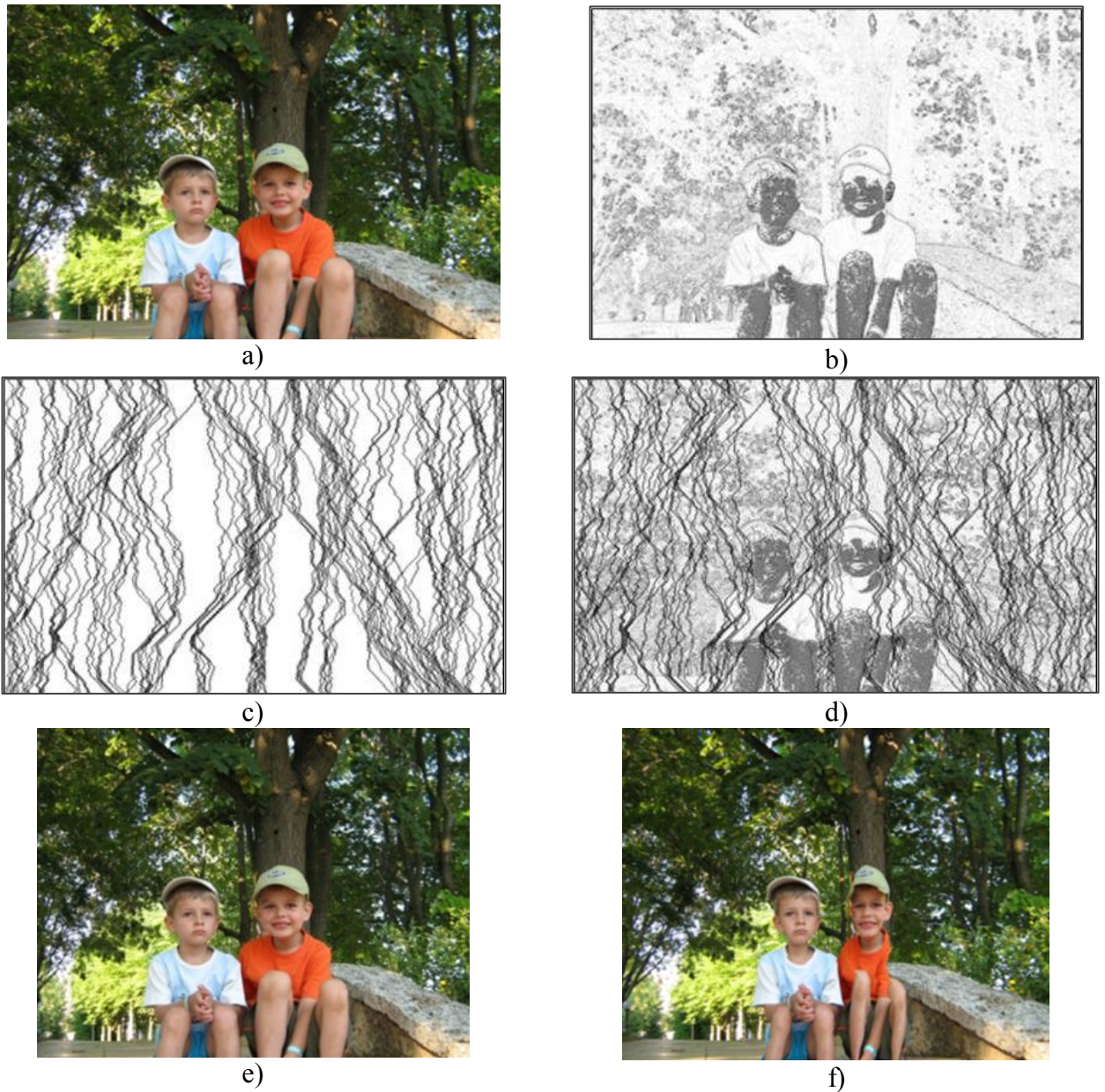


Рисунок 6.5 – а) исходное изображение; б) карта заметности; в) карта истории; г) сумма карт заметности и истории; е) результат предлагаемого переформатирования; ф) результат переформатирования алгоритмом из [303].

Пример предлагаемого переформатирования с помощью удаления цепочек пикселей между верхней и нижней сторонами фотографии, а также вычисляемые карты, показаны на рисунке 6.5. В картах заметности и истории чёрные и тёмно-серые цвета соответствуют большим значениям, а белые и светло-серые – малым. Карта истории показана для сотой итерации. На карте истории можно увидеть, что

в результате релаксации ряд цепочек выглядит светлее. На рисунке 6.5f показано переформатирование с помощью классического алгоритма *seam carving* [303]. Он существенно искажает лица и тела детей на фотографии. Результат изменения соотношения сторон фотографии предлагаемым способом (рисунок 6.5е) выглядит естественно и практически не имеет артефактов.

### Дополнение изображений

Для ряда изображений можно изменить соотношение сторон путём дополнения этого изображения отражёнными полосами с двух противоположных сторон. Иллюстрация такого преобразования показана на рисунке 6.6. Если в отражаемых полосах расположены фоновые области, то отражение будет незаметно для наблюдателя. Например, в сканированном изображении документа с полями равномерного оттенка с помощью отражения можно увеличить поля. Рисунок 6.7 демонстрирует пример дополнения фотографии с рисунка 6.1. отражёнными полосами справа и слева. Заметим, что для этой фотографии добиться необходимого соотношения сторон с помощью кадрирования невозможно, так как обрезается лицо. Дополнение фотографии обеспечивает визуально приятный результат. Похожий результат можно получить с помощью переформатирования с добавлением цепочек от верхнего до нижнего края изображения, но дополнение работает существенно быстрее переформатирования.

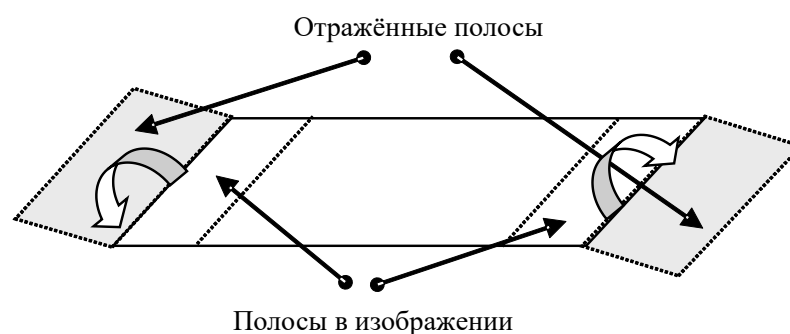


Рисунок 6.6 – Иллюстрация отражения полос изображения для изменения соотношения сторон.

Размеры полос по формуле (6.6) выбираются путём минимизации суммы значений карты заметности в полосах. Если среднее из значений карты заметности в отражаемых полосах превышает предопределённый порог  $T_m$ , то делается вывод, что данное преобразование заметно, ведёт к ухудшению изображения, и лучше его не делать.



Рисунок 6.7 – Дополнение фотографии с рисунка 6.1 зеркально отражёнными полосами.

### **Последовательность преобразований**

Рассмотрим рекомендуемую последовательность преобразований для реализации, в которой обработка выполняется полностью автоматически без участия пользователя. Пусть у изображения и бумаги для печати ширина всегда больше высоты, то есть используется, так называемая альбомная (landscape) ориентация. Пусть требуется увеличить отношение ширины к высоте. Для этого случая схема последовательности преобразований показана на рисунке 6.8.

На шаге 1 выполняется масштабирование изображения с коэффициентом равным отношению ширин бумаги и изображения. На шаге 2 выполняется обрезка полос сверху и снизу изображения, где размер полос получается в результате минимизации суммы значений карты заметности в обрезаемых полосах. Имеет смысл в первую очередь попробовать преобразовать изображение с помощью данной операции, так как обрезка полос равного размера обычно выполняется устройствами печати по умолчанию, таким образом это преобразование является

для пользователя наиболее привычным, процедура минимизации имеет быструю реализацию с помощью интегрального изображения (summed area table) [316].

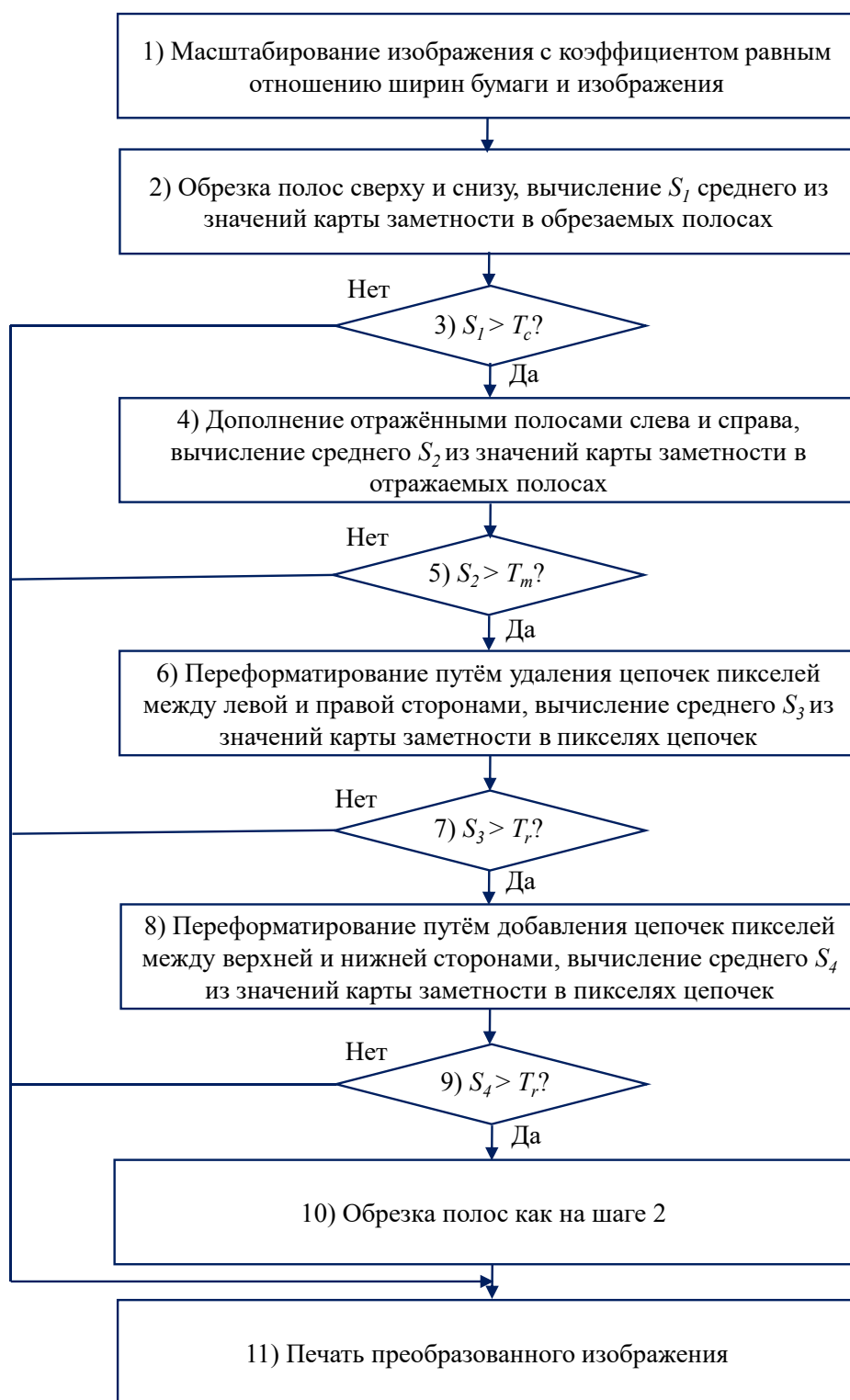


Рисунок 6.8 – Схема последовательности применения преобразований при изменении соотношения сторон изображения.

Если среднее из значений карты заметности в обрезаемых полосах меньше или равно  $T_c$  (условие 2), то результат преобразования печатается (шаг 11). В противном случае применяется другое преобразование. Дополнение отражёнными полосами слева и справа, где размер полос определяется с помощью минимизации суммы значений карты заметности в отражаемых полосах, также имеет быструю реализацию и работает быстрее переформатирования. Поэтому на шаге 4 выполняется отражение полос. Если среднее из значений карты заметности в отражаемых полосах меньше или равно  $T_m$  (условие 5), то результат преобразования масштабируется до размеров бумаги и печатается (шаг 11).

В противном случае выполняется переформатирование путём удаления цепочек пикселей между левой и правой сторонами изображения (шаг 6). Если среднее из значений карты заметности в пикселях удаляемых цепочек меньше или равно  $T_r$  (условие 7), то результат преобразования печатается (шаг 11). Иначе на шаге 8 выполняется переформатирование путём добавления цепочек пикселей между верхней и нижней сторонами. Если среднее из значений карты заметности в пикселях добавляемых цепочек меньше или равно  $T_r$  (условие 9), то результат преобразования масштабируется до размеров бумаги и печатается (шаг 11). Иначе на шаге 10 предполагается, что наименьшие искажения в преобразованном изображении обеспечит обрезка полос сверху и снизу с параметрами, полученными на шаге 2. Далее на шаге 11 изображение печатается.

Для случая, когда требуется уменьшить отношение ширины к высоте в схеме следует заменить низ/верх на лево/право и наоборот, коэффициент масштабирования на шаге 1 равен отношению высот бумаги и изображения.

Выбор значений порогов  $T_c, T_m$  и  $T_r$  выполнялся в ходе эксперимента, в котором 420 фотографий с соотношением сторон 4:3 обрезались, дополнялись и переформатировались для получения соотношения сторон 3:2. Для каждого преобразования отбирались фотографии, где модификация изображения незаметна. Среди таких фотографий вычислялось максимальное значение

соответствующих сумм значений карты заметности. Эти максимальные значения установлены в качестве порогов. Заметим, что порог  $T_m$  значительно ниже порогов  $T_c$  и  $T_r$ , используемых для обрезания полос и переформатирования.

### 6.3. Результаты

Рассмотрим печать без полей набора фотографий, которые имеют соотношение сторон 4:3, на термосублимационном фотопринтере Samsung SPP-2040. Принтер печатает на бумаге 15 на 10 см, то есть требуется преобразование соотношения сторон к 3:2. Разрешение печати 300 dpi. На первом этапе схемы с рисунка 6.8 изображение масштабируется до размера 1800 на 1350 пикселей, из которых печатается 1800 на 1200. Таким образом, требуется либо удалить из изображения 150 строк, либо добавить 225 столбцов, чтобы получить размер 2025 на 1350 пикселей, а затем уменьшить изображение до размера 1800 на 1200.

Набор состоит из 416 фотографий, снятых тремя пользователями в туристических поездках на отдых с семьёй. Эти же три пользователя оценивали качество преобразования фотографий в ходе следующего эксперимента: на экране монитора компьютера показываются исходная фотография размером 1800 на 1350 пикселей (соотношение сторон 4:3) и преобразованная фотография размером 1800 на 1200 пикселей (соотношение сторон 3:2); пользователь должен ответить на вопрос, ухудшается ли композиция фотографии в результате преобразования (да/нет); случайным образом показывается либо результат стандартной обрезки по 75 строк сверху и снизу изображения, либо результат предлагаемой группы преобразований, применяемых по схеме с рисунка 6.8; пользователь не знает, какой способ преобразования использовался; каждое изображение из набора преобразуется обоими способами.

Таблица 6.1 содержит результаты проведённого эксперимента. При использовании стандартного подхода композиция ухудшается у примерно 15 %

фотографий. Использование предлагаемой группы преобразований позволяет сократить это число более чем в 2 раза: ухудшается около 7 % фотографий. Безусловно, доля фотографий, которые ухудшаются при преобразованиях с изменением соотношения сторон, существенно зависит от разницы в соотношении сторон изображения и бумаги, типе сцены на фотографии, привычке фотографа строить сцену (в частности, насколько близко к краям изображения расположены основные объекты съёмки) и ещё целого ряда факторов. Тем не менее, данный эксперимент демонстрирует, что предлагаемые преобразования обеспечивают визуально лучший результат по сравнению со стандартной обрезкой полос одинакового размера с противоположных сторон изображения.

Таблица 6.1 Результаты изменения соотношения сторон фотографий.

Пользователь	% изображений, ухудшившихся при стандартной обрезке	% изображений, ухудшившихся при применении предлагаемых преобразований
1	15,1	6,8
2	14,6	7,1
3	15,3	6,9

Также в проведённом эксперименте определялось, какое из преобразований использовалось для изменения соотношения сторон. В 79,1 % это была обрезка полос оптимального размера, в 2,5 % – дополнение, в 18,4 % – переформатирование.

Рисунок 6.9 демонстрирует одну из фотографий набора (рисунок 6.9а) и результаты различных преобразований. При обрезке полос сверху и снизу (рисунок 6.9b) от фотографии отрезаются часть головы и хвоста птицы, находящейся на переднем плане. При дополнении изображения (рисунок 6.9c) полоса отражается от правого края изображения. Если внимательно рассмотреть изображение, то отражение заметно. Переформатирование путём добавления цепочек пикселей от верхнего до нижнего края изображения (рисунок 6.9d) создаёт изображение, в



котором модификация практически незаметна. Пороги  $T_c$ ,  $T_m$  и  $T_r$  настроены таким образом, что изображение с рисунка 6.9d выбирается в качестве результата для печати.

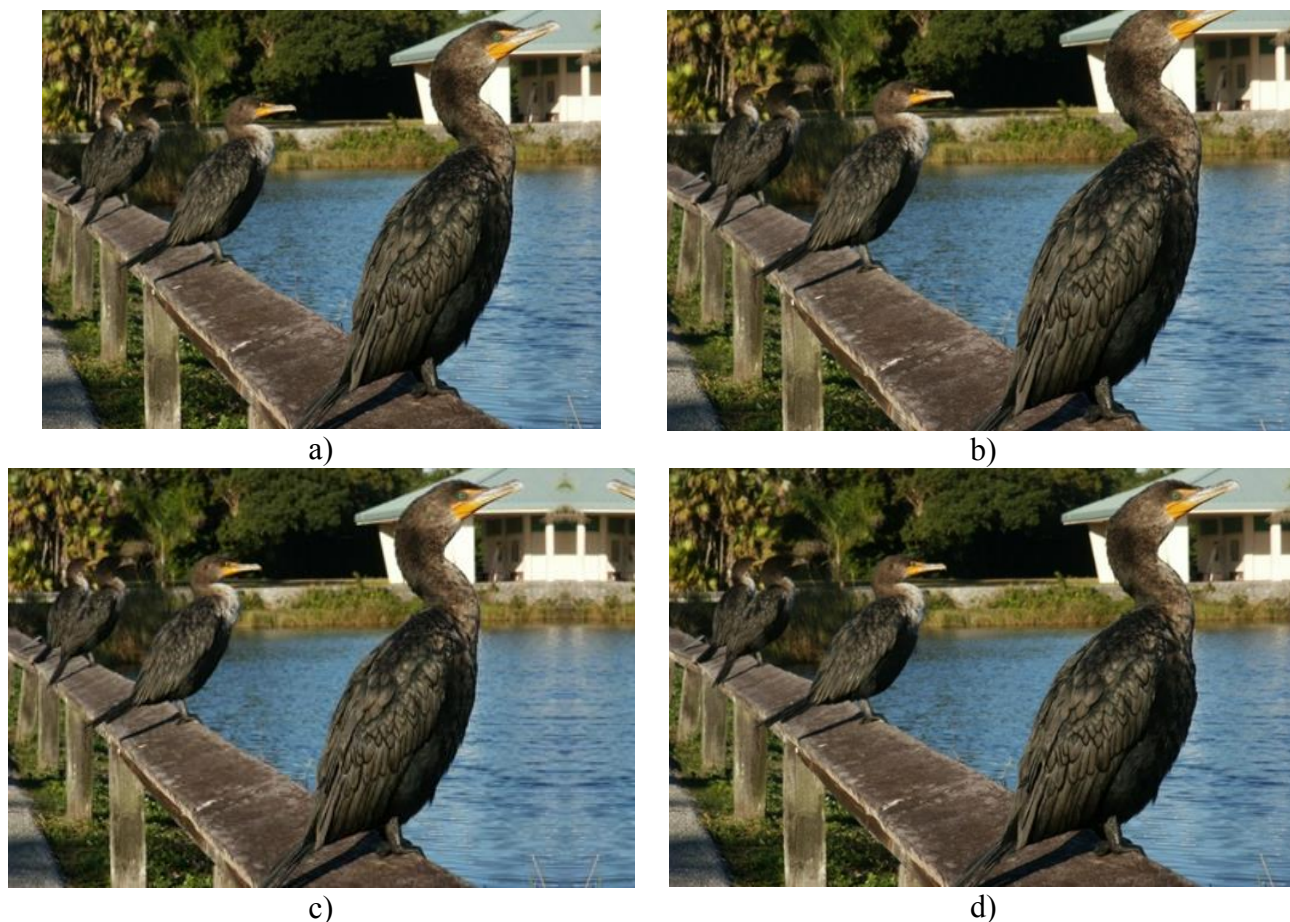


Рисунок 6.9 – а) исходная фотография; б) результат обрезания полос; в) результат дополнения; д) результат переформатирования.

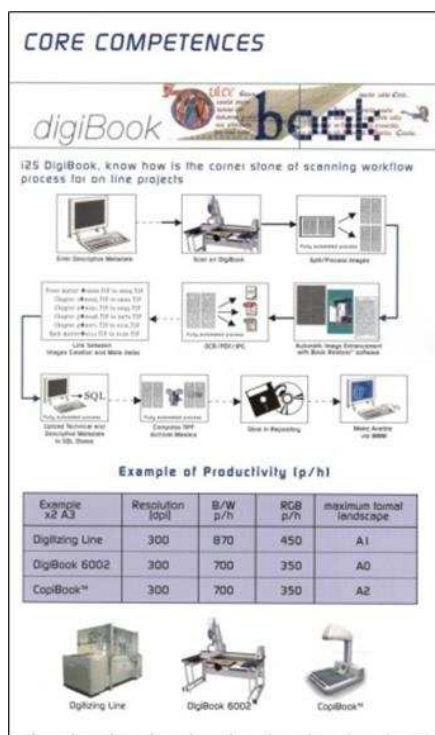
С помощью эмулятора вычислительной платформы фотопринтера оценивалось время выполнения преобразований для изменения соотношения сторон фотографий с 4:3 на 3:2. Первоначальное масштабирование изображений с учётом разрешения и размера бумаги выполняется при улучшении качества фотографий, которое описано в предыдущей главе. Поэтому дополнительное время на эту операцию не тратится. Исходная карта заметности вычисляется один раз для всех преобразований. Время вычисления карты составляет 0,4 с. Обрезка и



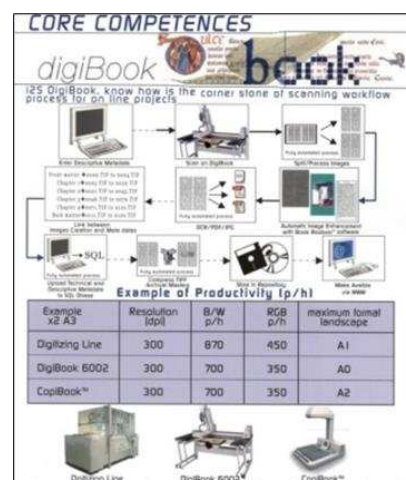
дополнение практически не зависят от количества удаляемых или добавляемых строк/столбцов, так как минимизация суммы значений карты заметности в полосах с помощью интегрального изображения выполняется быстро. Обрезка полос без учёта построения карты заметности занимает около 0.3 с. Дополнение отражёнными полосами требует немного больше времени из-за необходимости ещё одной операции масштабирования.

Время выполнения переформатирования сильно зависит от числа итераций. Поиск цепочки с помощью динамического программирования более-менее быстрая операция, но требуется несколько сотен итераций, а обновление карт заметности и истории с изменением их размера ведёт к дополнительным расходам времени. Переформатирование с удалением цепочек занимает около 5 с. Переформатирование с добавлением цепочек с учетом ещё одной операции масштабирования требует немного больше 6 с. Максимальное время преобразования одной фотографии по схеме с рисунка 6.8 составляет около 12 с., что слишком долго. Учитывая то, что чаще других преобразований используется обрезка полос, принято решение во встроенном ПО фотопринтера реализовать только предлагаемый алгоритм обрезки полос, а всю последовательность преобразований добавить в системное ПО фотопринтера, которое работает на ПК, вычислительные ресурсы которого существенно выше.

Размер изображения документа при копировании с помощью МФП в несколько раз больше размера печатаемой фотографии. Кроме того, требования к скорости копирования выше, чем к печати фотографий. Поэтому переформатирование было реализовано в драйвере сканирования МФП как функция преобразования изображения для размера бумаги, отличающегося от сканированного.



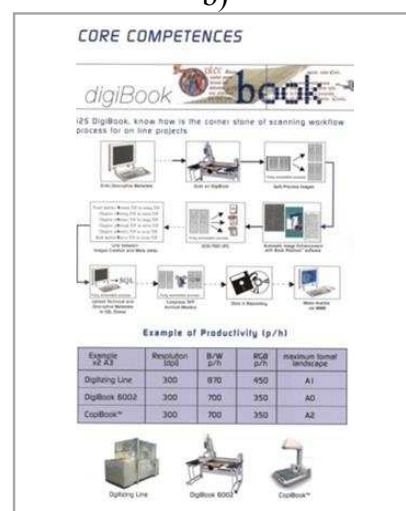
a)



b)



c)



d)

Рисунок 6.10 – а) исходное изображение размера *Legal*; б) изображение переформатировано в размер *Letter*; в) изображение по высоте масштабировано в размер *Letter*; д) изображение масштабировано в размер *Letter* с сохранением соотношения сторон.

На рисунке 6.10 показан пример преобразования изображения, сканированного со страницы размера *Legal* (рисунок 6.10а), для последующей печати на бумаге размера *Letter*. Бумага *Legal* имеет высоту на 27 % больше, чем *Letter*. При масштабировании по высоте (рисунок 6.10с) искажаются соотношения

размеров символов и иллюстраций, что в ряде случаев нежелательно для правильного восприятия информации. При масштабировании с сохранением соотношения сторон исходного документа (рисунок 6.10d) символы и рисунки становятся меньше, часть информации может пропасть, при этом в документе образуются большие белые поля. Переформатирование (рисунок 6.10b) оставляет размеры всех смысловых элементов документа без изменений, удаляются только участки фона. В ряде случаев переформатирование является предпочтительным способом изменения размеров и соотношения сторон изображения документа.

### **Выводы по шестой главе**

Рассмотрены способы преобразования изображений при копировании документов на бумагу отличающегося размера и при печати фотографий без полей. Отмечены недостатки существующих методов и предложена группа преобразований для изменения размеров и соотношения сторон сканированных документов и фотографий без модификации заметных деталей изображения.

Сформулированы следующие принципы определения заметных участков, модификация которых нежелательна:

- на фрагменте изображения присутствует лицо, тело человека или его часть, что определяется с помощью алгоритма детектирования лиц и/или оттенков кожи человека;
- на участке изображения есть протяжённые резкие контурные перепады, которые выделяются с помощью фильтра лапласиана гауссиана или разница гауссиан.

Для изменения размеров и соотношения сторон изображений документов и фотографий предлагаются следующие преобразования, основанные на анализе карты заметности:

- обрезка полос изображения с двух противоположных сторон, где размеры полос выбираются путём минимизации в обрезаемых областях суммы пикселей, которые оцениваются как заметные;
- переформатирование изображений за счёт удаления или добавления малозаметных участков по всему полю изображения;
- дополнение изображений симметрично отраженными полосами с двух противоположных сторон, где размеры полос выбираются путём минимизации в отражаемых областях суммы пикселей, которые оцениваются как заметные.

Предложена последовательность применения преобразований, которая позволяет для более чем 50% фотографий, которые ухудшались при печати без полей, избежать заметных искажений.

## **Глава 7. Принципы автоматического выбора фотографий из набора и размещения изображений на странице**

Некоторые МФП имеют функции печати фотографий. С их помощью можно печатать фотоальбомы/фотокниги, где на каждой странице размещается несколько изображений. Иногда фотографии на странице образуют единое изображение – коллаж.

Благодаря доступности фотокамер, зачастую пользователи ежедневно снимают десятки и даже сотни фотографий. В частности, это происходит в путешествиях, при посещении развлекательных мероприятий, в моменты памятных событий и тому подобное. При желании напечатать несколько фотографий для альбома или сформировать фотокнигу пользователи вынуждены тратить часы для выбора изображений из большой коллекции и размещения их на страницах. В отличие от вставки в альбом уже напечатанных фотографий фиксированного размера, как было в доцифровую эпоху, при создании страниц цифровой фотокниги требуется устанавливать размер каждого добавляемого изображения, таким образом возникает задача определения коэффициента масштабирования. Разработка способов, которые при создании фотоальбома из большого набора фотографий помогают пользователю существенно сократить время, является актуальной.

Безусловно, человек осуществляет выбор изображений и их размещение, исходя из плохо формализуемых факторов, часто на эмоциональном уровне и с учётом личных предпочтений. Тем не менее, можно сформулировать ряд принципов, то есть руководящих положений и установок [3], на основе которых реализуются методы автоматического отбора и размещения фотографий с определением коэффициента масштабирования, способные значительно ускорить создание фотоальбомов.

## 7.1. Выбор фотографий из коллекции

### Существующие способы

Алгоритмы выбора  $N_p$  изображений из набора, состоящего из  $N_s$  фотографий, где  $N_s \gg N_p$ , обсуждается в литературе как для автоматического формирования печатаемых или электронных фотоальбомов, так и для создания элементов интерфейса пользователя для просмотра больших фотоколлекций на экране ПК или мобильных устройств. Большинство существующих способов выбора основаны на кластеризации и ранжировании изображений по некоторому набору признаков. Как правило, фотоальбомы призваны сохранить память об определённых событиях жизни, фотографии в них часто располагают в порядке близком к хронологическому. В англоязычной литературе такой подход к составлению фотоальбомов называют «повествованием» (storytelling) [317]. Разумным способом определения нескольких фотографий, характеризующих всю коллекцию, является кластеризация по времени съёмки и выбор по одному изображению из каждого кластера [318-320]. Иногда используются иные подходы, например в [321] кластеризация выполняется по координатам съёмки, если они доступны из EXIF.

Ряд работ отмечает, что необходимо выбирать только качественные фотографии, поэтому целесообразно до кластеризации исключить из рассмотрения испорченные изображения, в частности размытые, слабоконтрастные, пересвеченные (overexposed) и недосвеченные (underexposed) [322, 323].

Из кластера выбирается наиболее «интересное» изображение. «Интересность» в литературе формулируется различным образом, но большинство авторов полагают, что фотографии, на которых есть лица людей [324], особенно улыбающихся [318] представляют наибольший интерес. В комбинации с количеством детектированных лиц часто используются признаки, характеризующие красочность (colorfulness) и насыщенность фотографий, так как красочное фото при условии сходства композиции, как правило, более

привлекательно. Иногда добавляются проверка «сходства» (similarity) отбираемых изображений, чтобы исключить выбор практически идентичных (near-duplicate) кадров. Например, в [324] оценкой красочности является энтропия двумерной гистограммы цветовых каналов  $ab$  пикселей изображения в цветовом пространстве  $La^*b^*$  [8], а оценкой сходства – сумма абсолютных разностей трёхмерных гистограмм в  $La^*b^*$ .

Ещё одним подходом для выбора изображения из кластера является ранжирование фотографий по их эстетической привлекательности [325]. Несколько лет назад были собраны и размечены наборы фотографий с точки зрения оценки их эстетики, затем на этих наборах обучались глубокие нейронные сети [326]. Однако, эксперименты, проведённые в работе [327], показали, что применение таких сетей для отбора фотографий при создании фотоальбомов слабо коррелирует с результатами человека.

### **Принципы выбора фотографий из набора**

Основываясь на анализе литературы и личных наблюдениях, предлагается формировать «рассказывающую историю» фотокнигу. В исследовании [328] отмечается, что более 90% печатаемых фотоальбомов содержат фотографии, снятые несколькими камерами. Это объясняется тем, что часто в путешествиях или на памятных мероприятиях съёмку ведут несколько человек. При наличии в наборе фотографий, снятых разными фотокамерами, желательно обеспечить в числе отобранных изображений присутствие кадров со всех камер. Целесообразно предварительно исключить некачественные фотографии, отбирать изображения с лицами людей и особенностями, привлекающими внимание.

Предлагаются следующие принципы автоматического выбора фотографий из коллекции для формирования фотокниги [329-334]:

1. оценка качества изображений и исключение из дальнейшего рассмотрения некачественных;
2. кластеризация в двумерном пространстве «время съёмки – номер камеры»;

3. выбор одной фотографии из кластера путём ранжирования изображений с помощью построения карты значимости и детектирования лиц.

### Оценка качества фотографий

При оценке качества предлагается детектировать:

- размытые изображения (способ описан в разделе 5.3.1);
- кадры, повреждённые контровым светом (способ описан в разделе 5.3.1);
- слабоконтрастные фотографии, для которых разность 99 и 1 перцентилей гистограммы яркости меньше порога  $T_c$ ;
- изображения, имеющие заметные артефакты сжатия JPEG;
- фотографии с высоким уровнем шума.

Изображения, испорченные JPEG сжатием, определяются с помощью анализа таблицы квантования  $q$ , предназначенной для канала яркости [335]. Такая таблица есть в заголовке JPEG файла. По значениям элементов левого верхнего угла таблицы квантования  $q$  вычисляется параметр  $Q_J$ :

$$Q_J = \frac{1}{9} \sum_{i=1}^3 \sum_{j=1}^3 q_{ij}. \quad (7.1)$$

Если  $Q_J < T_Q$ , то артефакты JPEG сжатия сильно заметны. В [332] приведены результаты эксперимента, в ходе которого определено  $T_Q = 6,5$ .

В литературе описан алгоритм оценки уровня шума, как наименьшего собственного значения  $\lambda$  матрицы ковариации, построенной для значений пикселей блока изображения [336]. Несмотря на то, что этот алгоритм теоретически обоснован только для аддитивного белого гауссова шума (АБГШ), на практике он применим и для фотографий, модель шума которых отличается от АБГШ [209]. Для изображений с глубиной цвета 8 бит на канал и размера блока 8 на 8 пикселей в ходе экспериментов определено что, если значение минимального собственного значения матрицы  $\lambda > 100$ , то фотография имеет высокий уровень шума. Для такого блока размер матрицы ковариации 64 на 64.



## Умножение матриц 32-разрядных чисел с помощью 16-разрядных векторных операций

Алгоритм оценки уровня шума требуется реализовать на DSP, используемом в системе на кристалле МФП. Сложность данной задачи заключается в том, что рассматриваемый DSP (основан на ядре Tensilica IVP) предназначен для работы с целочисленными значениями пикселей и не имеет инструкций обработки действительных чисел с плавающей точкой. DSP имеет 32-разрядные скалярные и 16-разрядные векторные (single instruction multiple data, SIMD) операции для целых чисел, в векторных регистрах помещается тридцать два 16-разрядных значения. Оценку уровня шума для рассматриваемого диапазона размера матриц (несколько десятков на несколько десятков) можно выполнять в арифметике с фиксированной точкой [97], где числа представлены в формате 16.16, то есть 16 старших бит представляют целую часть числа со знаком, а младшие 16 бит – дробную часть числа.

Предлагается алгоритм умножения прямоугольных матриц 32-разрядных чисел с фиксированной точкой с помощью 16-разрядных векторных операций [337, 338]. Рассмотрим сначала способ умножения  $n$ -разрядных чисел с помощью  $n/2$ -разрядных операций, а затем обобщим его для матриц чисел с фиксированной точкой.

Согласно алгоритму умножения Карацубы [338]:

$$uv = 2^n u_H v_H + 2^{n/2} (u_H v_L + u_L v_H) + u_L v_L, \quad (7.2)$$

где  $u_H$  – старшая или наиболее значимая половина беззнакового целого числа  $u$ ;  $u_L$  – младшая или наименее значимая половина  $u$ ; аналогично  $v_H$  и  $v_L$  половины числа  $v$ .

В [340] рассматриваются следующие способы распространения беззнакового умножения на знаковые операнды:

1. Получить абсолютные значения каждого множителя, выполнить беззнаковое умножение, а затем изменить знак результата, если знаки множителей различаются.
2. В формуле (7.2) при умножении старших половин использовать операцию умножения знакового на знаковое чисел, при умножении старших половин на младшие использовать операцию умножения знакового на беззнаковое чисел с последующим применением операции знакового расширения результата, при умножении младших половин друг на друга использовать операцию умножения беззнаковых чисел.
3. Выполнить беззнаковое умножение, а затем скорректировать его результат следующим образом. Число со знаком  $u$  рассматривается как беззнаковое  $u + 2^n u_{n-1}$ , где  $u_{n-1}$  – единица, если  $u$  отрицательно, и ноль, если  $u$  неотрицательно. Аналогично  $v$  рассматривается как  $v + 2^n v_{n-1}$ . Тогда их произведение:

$$(u + 2^n u_{n-1})(v + 2^n v_{n-1}) = uv + 2^n(vu_{n-1} + uv_{n-1}) + 2^{2n}u_{n-1}v_{n-1}. \quad (7.3)$$

Для получения результата знакового произведения целых чисел  $uv$  требуется вычесть из результата беззнакового умножения второе и третье слагаемые правой части формулы (7.3).

Пусть матрицы  $A$  и  $B$  содержат 32-разрядные числа. В случае матриц, содержащих целые числа без знака, для умножения применима формула (7.2). Требуется учесть, что в результате умножения матриц 16-разрядных чисел получается матрица с числами разрядности более 32. В рассматриваемом DSP результат умножения с помощью SIMD операций помещается в 48-разрядные регистры.

Из перечисленных выше трёх способов преобразования беззнакового умножения в знаковое первый неприменим для умножения матриц, второй не может быть использован для данного DSP по причине отсутствия необходимой векторной операции расширения знака, третий способ применим. Формируются

матрицы  $Da$ ,  $Db$  и  $S$ , элементы которых соответствуют второму (сумма в скобках) и третьему слагаемым правой части формулы (7.3):

$$Da(i, j) = \begin{cases} B(i, j), & A(i, j) < 0, \\ 0, & \text{иначе,} \end{cases} \quad (7.4)$$

$$Db(i, j) = \begin{cases} A(i, j), & B(i, j) < 0, \\ 0, & \text{иначе,} \end{cases} \quad (7.5)$$

$$S(i, j) = \begin{cases} 1, & A(i, j) < 0 \wedge B(i, j) < 0, \\ 0, & \text{иначе.} \end{cases} \quad (7.6)$$

Матрицы  $Da$  и  $Db$ , где значения согласно формуле 7.3 для  $n=32$  сдвинуты на 32 разряда влево, и матрица  $S$ , где значения сдвинуты на 64 разряда влево, вычитаются из результата умножения матриц  $A$  и  $B$  чисел, которые рассматриваются как беззнаковые. В соответствии с (7.2) при умножении матриц 16-битных чисел  $A_H B_L$  и  $A_L B_H$  значения 48-битного результата должны быть сдвинуты влево на 16 разрядов, где  $A_H$  и  $B_H$  составлены из старших 16 разрядов чисел из матриц  $A$  и  $B$  соответственно,  $A_L$  и  $B_L$  составлены из старших 16 разрядов чисел. При умножении матриц 16-битных чисел  $A_H B_H$  значения результата должны быть сдвинуты влево на 32 разряда.

	79:64	63:48	47:32	31:16	15:0
	$(A_H B_H)_H$	$(A_H B_H)_M$	$(A_H B_H)_L$		
+		$(A_H B_L)_H$	$(A_H B_L)_M$	$(A_H B_L)_L$	
+		$(A_L B_H)_H$	$(A_L B_H)_M$	$(A_L B_H)_L$	
+			$(A_L B_L)_H$	$(A_L B_L)_M$	$(A_L B_L)_L$
–		$Da_H$	$Da_L$		
–		$Db_H$	$Db_L$		
–	$S$				

Рисунок 7.1 – операции над матрицами 16-разрядных чисел при умножении матриц 32-разрядных целых чисел со знаком.

Рисунок 7.1 иллюстрирует умножение матриц 32-разрядных чисел со знаком с помощью 16-битных операций. В верхней части рисунка отмечены диапазоны по 16 бит, в которых располагаются компоненты, из которых формируется результат умножения. В левой части рисунка отмечены операции сложения или вычитания, применяемые к этим компонентам. У матриц 48-разрядных чисел 16-разрядные слова обозначены нижними индексами  $H$  – старшее,  $M$  – среднее и  $L$  – младшее. В прямоугольную рамку на рисунке 7.1 заключены 16-битные слова, используемые для вычисления результата в формате 16.16 чисел с фиксированной точкой. Можно не вычислять слагаемые за пределами рамки, что ведёт к ускорению работы программы, однако контроль за отсутствием переполнения и корректностью вычислений ложится на программиста.

При использовании SIMD инструкций предпочтительным является реализация матричного умножения не с помощью скалярного (dot или inner product) произведения векторов, а через тензорное произведение векторов (tensor или outer product), так как при этом регистры заполняются элементами матриц, лежащими в памяти один за другим. Тензорное произведение вектора-столбца  $a$  на вектор строку  $b$  образуют матрицу:

$$\begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{pmatrix} \otimes (b_1 \quad b_2 \quad \cdots \quad b_n) = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \cdots & a_1 b_n \\ a_2 b_1 & a_2 b_2 & \cdots & a_2 b_n \\ \vdots & \vdots & \ddots & \vdots \\ a_m b_1 & a_m b_2 & \cdots & a_m b_n \end{pmatrix}, \quad (7.7)$$

При перемножении матриц левый множитель можно рассматривать как вектор-строку, состоящую из векторов столбцов, а матрицу – правый множитель как вектор-столбец, состоящий из векторов-строк. Тогда матричное произведение есть сумма результатов тензорных произведений  $i$ -х векторов множителей:

$$(\vec{a}_1 \quad \vec{a}_2 \quad \cdots \quad \vec{a}_n) \begin{pmatrix} \vec{b}_1 \\ \vec{b}_2 \\ \vdots \\ \vec{b}_n \end{pmatrix} = \sum_{i=1}^n \vec{a}_i \otimes \vec{b}_i. \quad (7.8)$$

Схема умножения матриц 32-разрядных чисел с фиксированной точкой с помощью операций над матрицами 16-разрядных чисел приведена на рисунке 7.2. Блоки Split выполняют декомпозицию матриц 32-разрядных чисел на матрицы младших и старших 16-разрядных слов. Блоки Mul выполняют матричное перемножение для матриц 16-разрядных чисел. BAdd предназначен для заполнения матриц  $Da$  и  $Db$  в зависимости от знака значений в матрицах  $A$  и  $B$  и суммирования  $Da$ ,  $Db$ . Результаты умножения  $A_H B_L$  и  $A_L B_H$  суммируются в блоке Add. В блоках ShiftAdd и ShiftSub происходит соответственно сложение и вычитание матриц, числа в которых сдвинуты в позиции, в зависимости от используемого формата чисел с фиксированной точкой. Для вычислительных платформ, имеющих несколько DSP, блоки, расположенные на одном уровне, не зависят друг от друга и могут выполняться параллельно.

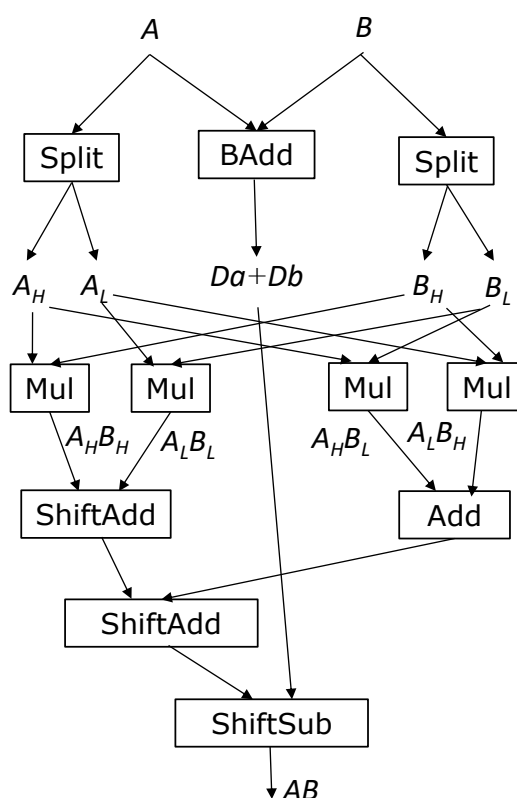


Рисунок 7.2 – Схема умножения матриц 32-разрядных чисел с фиксированной точкой с помощью операций над матрицами 16-разрядных чисел.

На рисунке 7.3 показаны зависимости времени умножения квадратных матриц 32-разрядных чисел с фиксированной точкой от размера матриц  $N$  для следующих способов: путём скалярного произведения векторов с использованием 32-разрядных скалярных инструкций DSP; с помощью тензорного произведения векторов с использованием 32-разрядных скалярных операций; алгоритмом Штрассена [341] с помощью 32-разрядных скалярных инструкций; предлагаемым способом с помощью 16-разрядных векторных инструкций. Несмотря на то, что асимптотическая сложность алгоритма Штрассена и ряда «быстрых» алгоритмов умножения матриц, например [342], ниже, чем сложность умножения с помощью скалярного и тензорного произведения векторов, на практике такие алгоритмы эффективны только для матриц размера несколько десятков тысяч на несколько десятков тысяч и более.

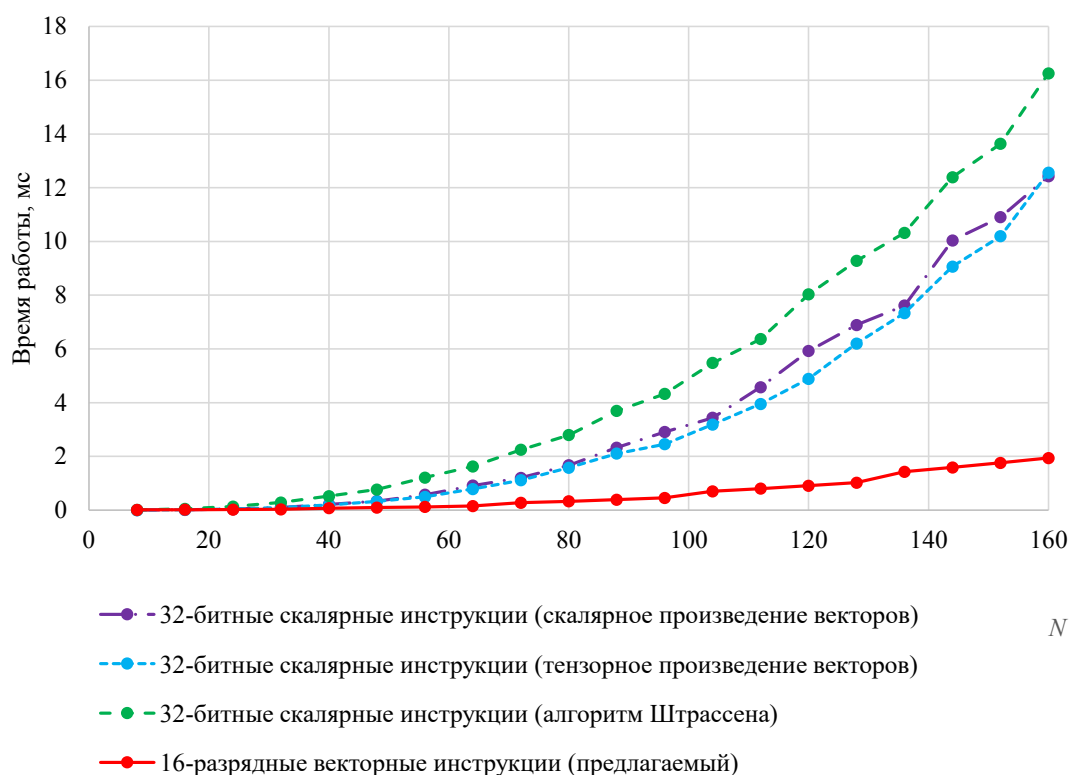


Рисунок 7.3 – Время работы различных способов умножения квадратных матриц 32-разрядных чисел с фиксированной точкой в зависимости от размера матриц.

Для рассматриваемого диапазона размеров матриц предлагаемый алгоритм быстрее чем реализации с помощью 32-разрядных скалярных операций DSP. В частности, для матриц размера 64 на 64 он быстрее более чем в 5 раз. Алгоритм обобщается для умножения матриц  $n$ -разрядных целых со знаком с помощью  $n/2$ -разрядных векторных операций. По аналогии с описанным в данном разделе можно сформулировать способы умножения матриц  $n$ -разрядных чисел с помощью  $n/4$ - $n/8$ -разрядных операций.

### Кластеризация

Чтобы пространство «время съёмки – номер камеры» было метрическим необходимо либо определить для него специальную функцию расстояния, либо использовать одинаковые единицы измерения по осям, что позволит применять известные функции расстояния, например евклидово. Предлагается применить второй вариант: время съёмки указывать в секундах, а по оси «номер камеры» установить шаг  $H_c$  в единицах времени, как показано на рисунке 7.4:

$$H_c = \frac{k_c L_t}{N_p}, \quad (7.9)$$

где  $L_t$  – разница во времени между самым давним и самым новым снимками в коллекции;  $N_p$  – количество отбираемых фотографий; коэффициент  $k_c \in [0, N_p]$ .

При  $k_c = 0$  кластеризация выполняется только по времени, при  $k_c = N_p$  кластеризация осуществляется только по модели камеры, промежуточные значения  $k_c$  позволяют регулировать поведение кластеризации между этими двумя крайними случаями, рекомендуется  $k_c = 2$ .

Если в некоторый промежуток времени было снято больше фотографий, чем в другие периоды, то можно предположить, что в этот момент происходило нечто интересное, и целесообразно из этого промежутка отобрать большее количество фотографий. Поэтому для кластеризации предлагается применить алгоритм срединного сечения (median-cut) [343], используемый для подбора палитры, включающей несколько десятков цветов, из тысяч, присутствующих в

изображении. Этот алгоритм обеспечивает приблизительно одинаковое количество точек в кластерах, большее количество кластеров образуется в местах плотного расположения точек и меньшее в местах, где точки разрежены.

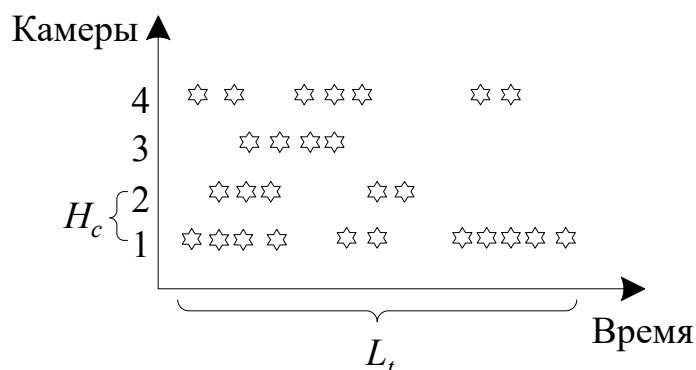


Рисунок 7.4 – Двумерное пространство «время- номер камеры» для кластеризации фотографий.

### Ранжирование изображений

Ранжировать изображения в пределах кластера для выбора одного с наибольшим рангом предлагается за счёт построения карты значимости (saliency map) и детектирования лиц. Для решения обеих задач существует множество методов, которые существенно различаются вычислительной сложностью и объёмом потребляемой памяти. В разделе 6.2 было отмечено, что алгоритм Виолы-Джонса [261] с модификациями из [309] может быть эффективно реализован на встроенных платформах. Классический способ построения карты значимости, описанный в [344], основан на модели раннего (preattentive) зрения. Он не требует значительных вычислительных затрат и обеспечивает разумные интерпретируемые результаты. Пример карты значимости показан на рисунке 7.5.

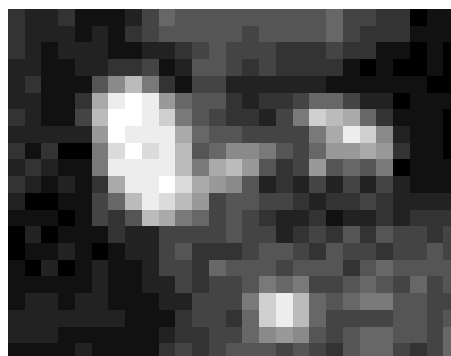
Без ограничения общности, пусть способом из [344] для фотографии построена карта значимости  $I_s$ . На  $I_s$  подавляются пиксели, значение в которых невелико, чтобы такие пиксели не участвовали в вычислении ранга фотографии:



$$I'_s(r, c) = \begin{cases} I_s(r, c), & I_s(r, c) \geq 0,25 \max_{v(r,c)} I_s(r, c), \\ 0, & \text{иначе.} \end{cases} \quad (7.10)$$



a)



b)

Рисунок 7.5 – а) фотография; б) карта значимости.

Ранг  $R$  изображения, т.е. величина по которой выполняется ранжирование, вычисляется по формуле:

$$R = w_f N_f^{a_f} + \frac{1}{WH} \sum_{r=0}^{H-1} \sum_{c=0}^{W-1} I'_s(r, c), \quad (7.11)$$

где  $W, H$  – размеры карты значимости  $I'_s$ ;  $N_f$  – количество обнаруженных на фотографии лиц;  $w_f$  и  $a_f$  – коэффициенты.

Значение  $w_f = 25$  и  $a_f = 0,5$  найдены в ходе эксперимента, в котором 54 изображения (на 38 были лица людей) ранжировались автором этих фотографий, далее максимизировался коэффициент ранговой корреляции Спирмена между результатами ранжирования человеком и по формуле (7.11).

### Результаты

Для оценки качества предлагаемого способа автоматического выбора фотографий был проведён следующий эксперимент [332]. Пяти пользователям было предложено предоставить наборы из 30 фотографий, снятых подряд без какого-либо предварительного отбора и относящихся к некоему определённом событию. Далее из 30 фотографий выбиралось 10 с помощью предлагаемого

метода, способом из [324] и путём случайного выбора. Авторы фотографий также были инструктированы выбрать 10 изображений, которые, по их мнению, могли бы войти в фотоальбом. Кроме того, каждый пользователь для своего набора разделил оставшиеся 20 изображений на две группы «приемлемо» для включения в фотокнигу и «неприемлемо». Результат эксперимента приведены в таблице 7.1. Наибольшее количество совпадений результатов выбора с человеком обеспечил предлагаемый метод, также у него наименьшее количество отобранных фотографий, которые оценены как «неприемлемые». Способ из [324] продемонстрировал результаты заметно лучше случайного выбора, но уступает предлагаемому по количеству совпадений с результатами ручного выбора.

Таблица 7.1 Сравнение способов автоматического выбора фотографий.

Способ	Оценка	Набор 1	Набор 2	Набор 3	Набор 4	Набор 5	Сумма
Случайный выбор	Совпадает	2	2	3	4	4	15
	Приемлемо	5	5	4	2	5	21
	Неприемлемо	3	3	3	4	1	14
Способ из [324]	Совпадает	2	2	2	6	5	17
	Приемлемо	6	7	7	0	4	24
	Неприемлемо	2	1	1	4	1	9
Предлагаемый	Совпадает	6	5	6	5	7	29
	Приемлемо	3	4	4	4	2	17
	Неприемлемо	1	1	0	1	1	4

Рассмотрим подробнее набор 1 из таблицы 7.1 и продемонстрируем, как осуществлялся выбор изображений предлагаемым способом. Набор состоит из 30 фотографий посещения зоопарка. Съёмка выполнялась двумя камерами. Изображения показаны на рисунке 7.6 в пространстве «время съёмки - камера». Под каждой фотографией указан её номер. Четыре фотографии оказались нечёткими и были исключены из рассмотрения. На рисунке 7.6 такие изображения перечёркнуты красным. Получившиеся в результате кластеризации 10 кластеров

обведены синей пунктирной линией. Выбранные из кластеров фотографии обведены красной пунктирной линией.

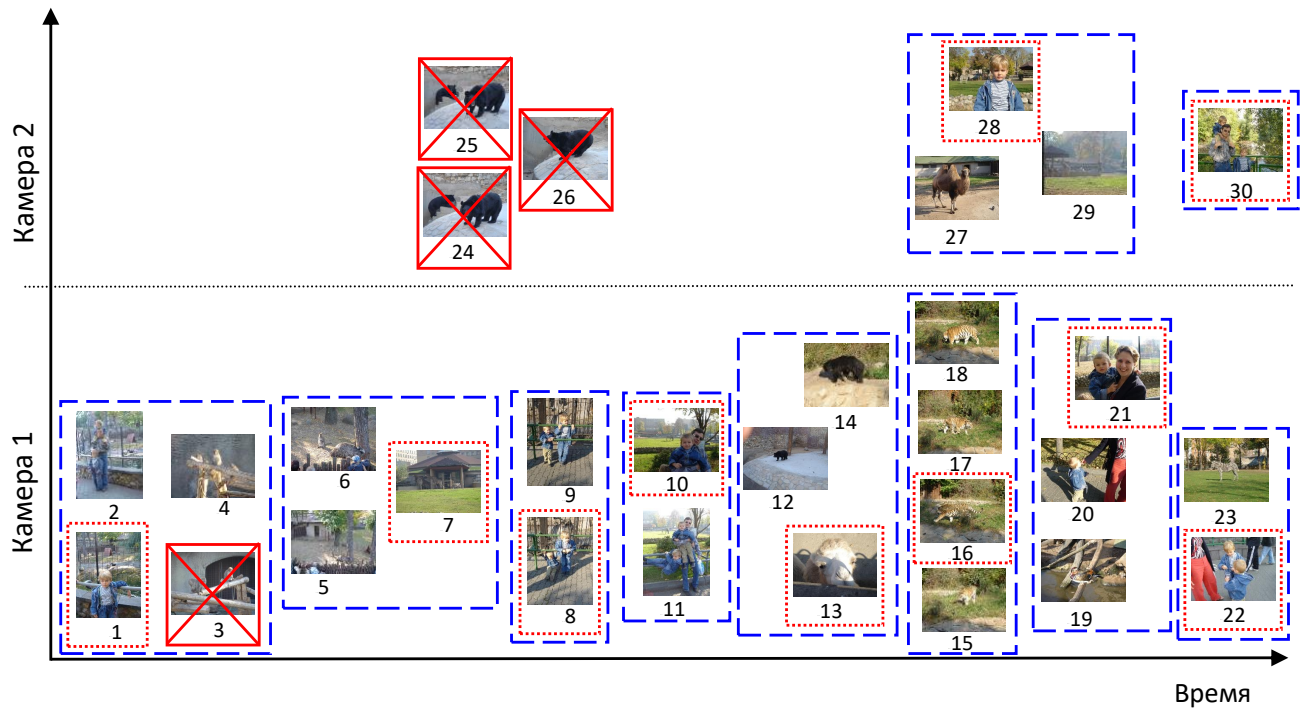


Рисунок 7.6 – Иллюстрация выбора фотографий из набора предлагаемым способом.

Результат отбора совпал с выбором пользователя для 6 фотографий: это изображения с номерами 1, 8, 10, 21, 28, 30. Выбор 3 фотографий пользователь посчитал приемлемым: 13, 16, 22. Одно изображение (номер 7) отмечено, как «неприемлемое» для включения в фотоальбом.

На рисунке 7.7 приведены результаты выбора 10 изображений из набора 1 несколькими способами. Способом из [324] и случайным выбором были отобраны в том числе нечёткие фотографии, которые пользователь отметил, как «неприемлемые» для включения в фотоальбом.

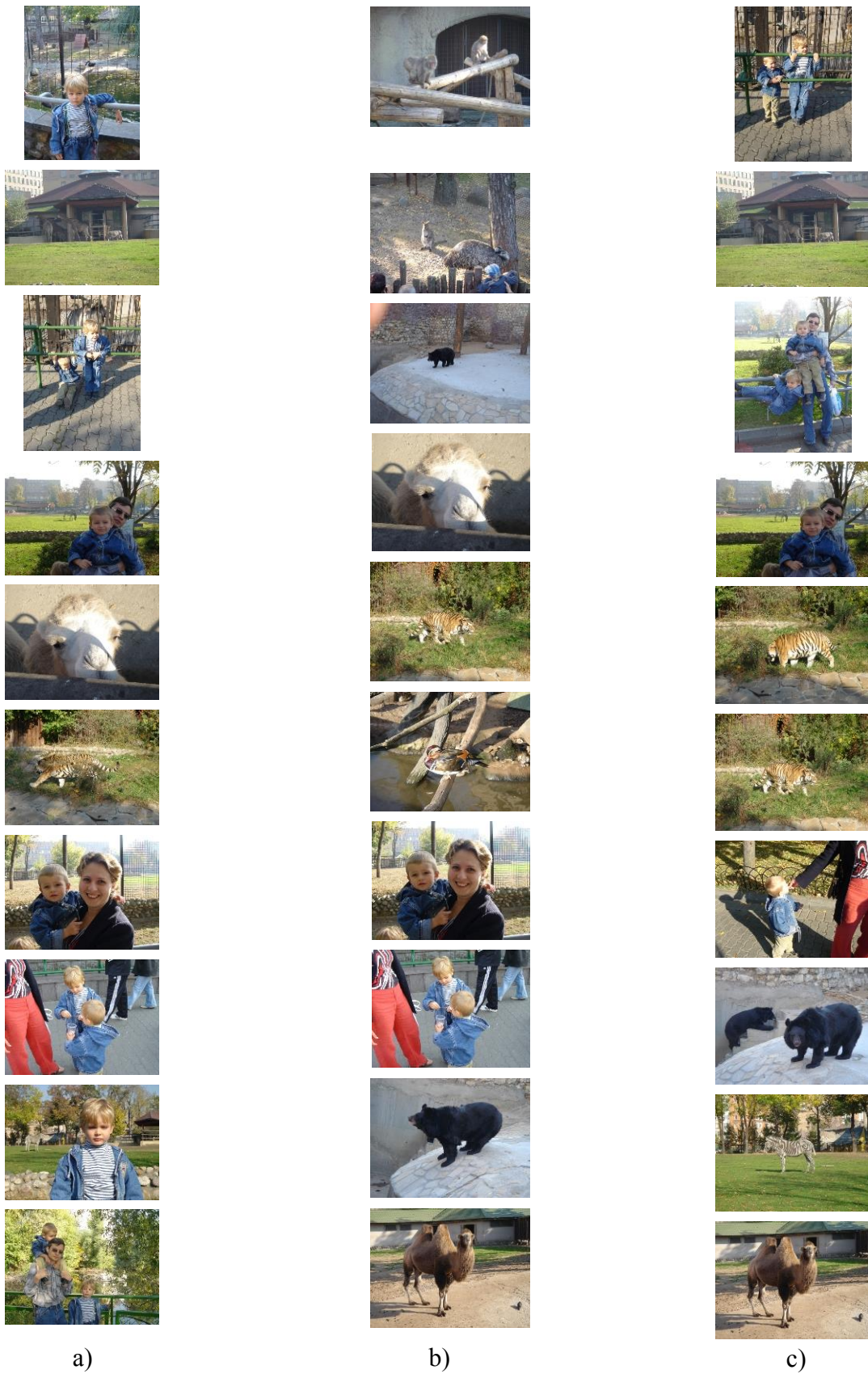


Рисунок 7.7 – Выбранные фотографии: а) предлагаемым методом; б) способ из [324]; в) случайным образом.

## 7.2. Размещение фотографий на странице

### Существующие способы

Широко известны задачи двумерной и трёхмерной упаковки и размещения, такие как упаковка в контейнеры, упаковка в полосу, укладка рюкзака, прямоугольного раскроя/разрезания заготовок [345, 346]. Несмотря на похожие формулировки этих задач, существующие методы их решения применимы для размещения только объектов фиксированного размера и не позволяют вычислять размер изображений на странице.

Программное обеспечение для создания фотокниг, как правило, использует набор шаблонов для размещения фотографий [317]. Иногда пользователю показывается нескольких вариантов размещения и предложению выбрать наиболее понравившийся. Если соотношение сторон фотографий и предназначенных для них позиций в шаблоне существенно различается, то либо изображения обрезаются, при этом может пострадать их композиция, либо изменяются размеры позиций, что может привести к частичному перекрытию изображений.

Исследователями компании HP предложен алгоритм BRIC (blocked recursive image composition) для автоматического размещения фотографий на странице [347, 348]. В этом способе для заданного количества изображений рассматриваются все возможные схемы размещения, задаваемые в виде бинарных деревьев. Каждое дерево строится рекурсивно путём разбиения пополам доступной для размещения прямоугольной области по вертикали или горизонтали. Для каждой схемы размещения коэффициенты масштабирования изображений вычисляются путём решения одной из двух систем уравнений: в первой из них суммы ширин изображений, умноженные на коэффициенты масштабирования, равны ширине страницы; во второй суммы высот изображений, умноженные на коэффициенты масштабирования, равны высоте страницы. Также в системах уравнения учитывается величина промежутка между фотографиями. Из всех возможных схем

размещения с выбирается та, при которой площадь свободного пространства на странице минимальна. Размещённые фотографии всегда полностью заполняют прямоугольную область (без учёта фиксированного промежутка между ними).

В некоторых случаях при решении системы уравнений коэффициенты масштабирования могут получаться отрицательными. В этом случае схема размещения не рассматривается. Иногда исключаются из рассмотрения все возможные схемы, и размещение не может быть выполнено. В таком случае авторы BRIC рекомендуют изменить размеры фотографий и страницы или величину промежутка между фотографиями, после чего запустить алгоритм размещения заново.



Рисунок 7.8 – Пример размещения фотографий алгоритмом BRIC.

Ещё одним недостатком BRIC является то, что в результате размещения размеры фотографий на странице могут сильно отличаться друг от друга, что, как правило, нежелательно. На рисунке 7.8 показан пример размещения алгоритмом BRIC трёх фотографий на странице размера А4 в альбомной ориентации. Фотография, размещённая в правом нижнем углу, получилась слишком маленькой, на странице она в несколько раз меньше, чем два других изображения.

Несмотря на перечисленные недостатки, результаты размещения алгоритмом BRIC, как правило, нравятся многим пользователям. В частности, отмечается, что



компактное размещение в полностью заполненной изображениями прямоугольной области выглядит гармонично [347].

### Принцип размещения фотографий

Предлагается не ограничиваться схемами размещения изображений на странице, которые используются в BRIC, а рассматривать все возможные варианты, определять коэффициенты масштабирования изображений путём минимизации свободного пространства на странице и разности размеров размещённых фотографий [349, 350]. На рисунке 7.9 показаны все возможные схемы размещения на странице трёх изображений без учёта перестановок изображений местами.

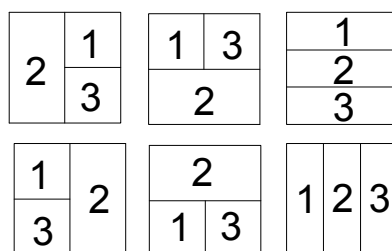


Рисунок 7.9 – Варианты схем размещения для 3-х изображений.

Заметим, что требование минимизации незаполненного пространства на странице необходимо для нахождения оптимальной схемы размещения и соответствующих ей коэффициентов масштабирования изображений, но не ограничивает размещение только в виде плотной укладки «плиткой» (tile). Далее можно дополнительно умножить полученные коэффициенты масштабирования на число меньше единицы для уменьшения изображений, что обеспечит промежутки между фотографиями на странице. Или, наоборот, умножить на число больше единицы для увеличения изображений, что приведёт к созданию примерно равных полос перекрытий при формировании коллажа.

Если нет перекрытий изображений, и они занимают всю страницу, то имеется явная зависимость между размерами фотографий и размером страницы.

Рассмотрим схему размещения, показанную в левом нижнем углу рисунка 7.9. Для неё справедлива следующая система уравнений:

$$\begin{cases} W_c = \tau_1 W_1 + \tau_2 W_2, \\ W_c = \tau_2 W_2 + \tau_3 W_3, \\ H_c = \tau_1 H_1 + \tau_3 H_3, \\ H_c = \tau_2 H_2, \end{cases} \quad (7.12)$$

где  $W_i$  – ширина  $i$ -го изображения;  $H_i$  – высота  $i$ -го изображения;  $\tau_i$  – коэффициент масштабирования  $i$ -го изображения;  $W_c$  – ширина страницы;  $H_c$  – высота страницы.

Аналогичные системы уравнений можно написать для любой схемы размещения. В BRIC аналитически решаются подобные уравнения для ширин или высот, но  $\tau_i$  могут получаться отрицательными, что недопустимо. В [350] предлагается искать решение всей системы уравнений аналогичных (7.12) с помощью МНК, минимизируя суммарную площадь пустого пространства и перекрытий, но такой способ также не исключает возникновения отрицательных  $\tau_i$ . Предлагается рассматривать ограниченный диапазон положительных значений  $\tau_i$  и решать задачу минимизации незанятого пространства на странице, при этом штрафовать за перекрытия фотографий и разницу в размерах размещённых изображений.

В общем виде системы уравнений аналогичных (7.12) можно записать как:

$$g_c = G\tau, \quad (7.13)$$

где  $G$  – матрица размеров (ширин и высот) изображений;  $\tau$  – вектор-столбец коэффициентов масштабирования, количество элементов которого равно числу размещаемых изображений  $n$ ;  $g_c$  – вектор-столбец размеров страницы,  $i$ -й элемент которого  $W_c$ , если в  $i$ -й строке  $G$  расположены значения ширин изображений, или  $i$ -й элемент которого  $H_c$ , если в  $i$ -й строке  $G$  расположены значения высот. Например, для уравнений (7.12) получаются следующие  $g_c$  и  $G$ :

$$g_c = \begin{bmatrix} W_c \\ W_c \\ H_c \\ H_c \end{bmatrix}, G = \begin{bmatrix} W_1 & W_2 & 0 \\ 0 & W_2 & W_3 \\ H_1 & 0 & H_3 \\ 0 & H_2 & 0 \end{bmatrix}. \quad (7.14)$$



Ограничим диапазон изменения  $\tau_i$  от  $\tau_{min}$  до  $\tau_{max}$ :

$$\tau_{min} = 0,2 \frac{W_c H_c}{n \sum_{i=1}^n W_i H_i}, \quad \tau_{max} = 2 \frac{n W_c H_c}{\sum_{i=1}^n W_i H_i}, \quad (7.15)$$

Обозначим  $d$  разность векторов  $g_c$  и  $G\tau$ :

$$d = g_c - G\tau. \quad (7.16)$$

Значения  $d_i < 0$  свидетельствуют о перекрытии изображений. Такие ситуации штрафуются путём присвоения  $d_i$  значения больше площади страницы:

$$d_i = \begin{cases} W_c H_c + |d_i|, & d_i < 0 \\ d_i, & \text{иначе} \end{cases}. \quad (7.17)$$

Минимизация суммы  $d_i$  для  $\tau_i \in [\tau_{min}, \tau_{max}]$  позволяет найти коэффициенты масштабирования, при которых площадь незанятого на странице пространства минимальна. Однако такой подход часто ведёт к решению, при котором существенно отличается размер изображений, размещённых на странице. Чтобы избежать этого, предлагается в минимизируемую функцию добавить сумму абсолютных значений разности площадей изображений на странице. Таким образом, получаем следующую минимизируемую функцию  $f_s$  для схемы  $s$ :

$$f_s = \sum_{i=1}^n d_i + \frac{2\omega}{n(n-1)} \sum_{i=1}^{n-1} \sum_{j=i+1}^n |W_i H_i \tau_i^2 - W_j H_j \tau_j^2| \xrightarrow{\tau_i \in [\tau_{min}, \tau_{max}]} \min, \quad (7.18)$$

где  $\omega$  – коэффициент, с помощью которого регулируется влияние требования к минимальной разности размеров изображений, размещённых на странице.

Функция  $f_s$  не является выпуклой. Для её минимизации применяется алгоритм роя частиц [351]. Из всех схем  $s$  выбирается та, для которой  $f_s$  минимальна.

При увеличении  $n$  число схем растёт факториально. На странице А4 обычно размещают от 2 до 6 изображений. На встроенной платформе МФП время поиска оптимальной  $f_s$  для  $n=6$  составило около 5 с, а для  $n=7$  уже более 220 с. Поэтому для  $n$  больше 6 рассматривается подмножество возможных схем размещения, которое выбирается случайным образом.

### Примеры размещения

На рисунке 7.7 показаны примеры размещения трёх фотографий на странице размера А4 в альбомной ориентации. При установке в формуле (7.11) параметра  $\omega = 0,1$  размеры размещённых фотографий примерно равны (рисунок 7.10а). Если  $\omega = 0$ , то одна из фотографий больше двух других, но страница почти полностью заполнена (рисунок 7.10б). Во втором случае площадь, занятая фотографиями, выше, чем обеспечивает алгоритм BRIC для этих же изображений (рисунок 7.8).

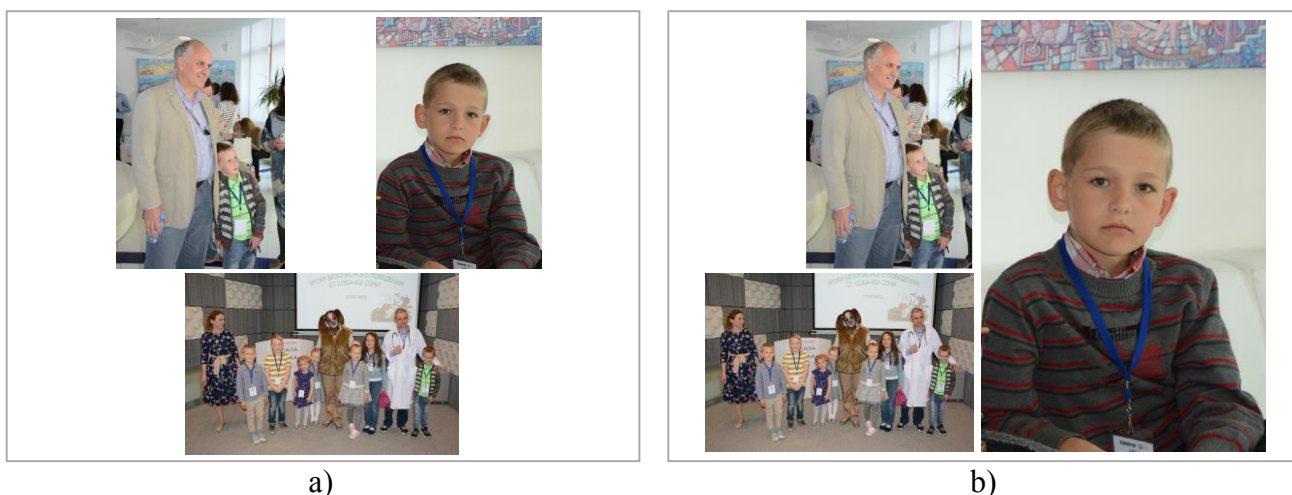


Рисунок 7.10 – Примеры размещения предложенным способом: а) параметр  $\omega = 0,1$ ; б)  $\omega = 0$ .

Учитывая, что пользователи имеют разные предпочтения, целесообразно при реализации, по крайней мере в прикладном ПО для ПК, предоставлять выбор из нескольких вариантов размещения, включая сформированное алгоритмом BRIC, если с помощью BRIC для данных размеров страницы и фотографий размещение возможно.

## 7.3. Формирование коллажа

### Типы коллажей

В большинстве публикаций описывают алгоритмы автоматического создания коллажей одного из следующих типов:

- в виде расположения «плиткой» (tiled), где фотографии касаются друг друга [352-354] (см. рисунок 7.11a); для создания таких коллажей, как правило, используются различные модификации алгоритма BRIC, части изображений, которые не удовлетворяют условию вписывания в прямоугольник, обрезают;
- в виде повёрнутых на небольшой угол относительно горизонтали изображений [355, 356], создаётся впечатление лежащих на столе и частично перекрывающихся фотографий (см. рисунок 7.11b); видимая часть фотографии определяется с помощью построения карты значимости;
- в виде изображений с перекрытием, между которыми обеспечен плавный переход (перетекание) одного изображения в другое [324, 357] (см. рисунок 7.11c); для совмещения перекрывающихся частей изображения также часто используется карта значимости.

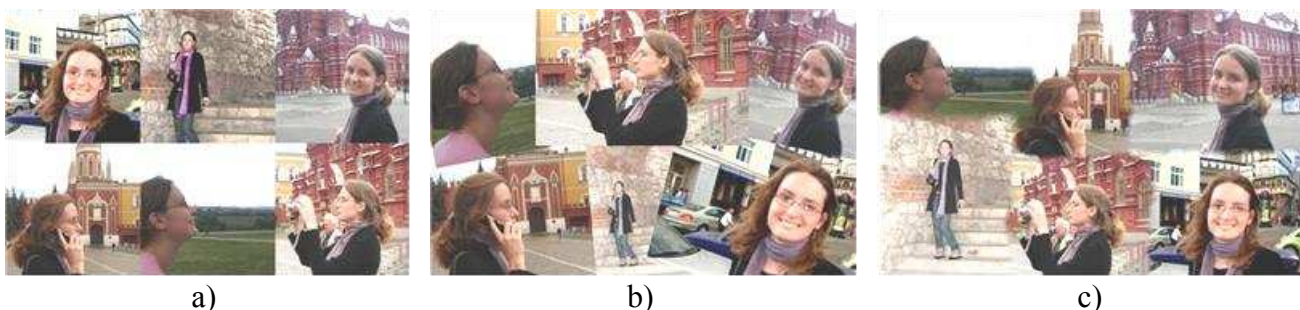


Рисунок 7.11 – Варианты коллажей: а) расположение «плиткой»; б) повёрнутые изображения; в) плавные переходы между фотографиями.

Иные типы коллажей формируются редко. В [358] небольшие фотографии размещаются на незначимых областях изображения. В [359] коллаж строится из большого количества фрагментов фотографий произвольной формы.

#### **Алгоритм создания плавного перехода между изображениями**

В [349] проводился опрос пользователей о предпочтительном типе коллажа. Большинство участников опроса отметило коллажи с плавными переходами между фотографиями как наиболее интересные.

Изображения размещаются на странице с помощью способа, описанного в разделе 7.2. Затем коэффициенты масштабирования увеличиваются, чтобы обеспечить отсутствие незаполненных фрагментов и частичное перекрытие изображений. Для создания в области перекрытий плавного перехода между изображениями предлагается вычислительно простой и компактный алгоритм.

Пусть изображения  $I_1$  и  $I_2$  частично перекрываются по вертикальной стороне как показано на рисунке 7.12. Обозначим  $I_{f1}$  и  $I_{f2}$  перекрывающиеся части изображений  $I_1$  и  $I_2$  соответственно. Введем систему координат, начало которой находится в левом нижнем углу области перекрытия, ось  $x$  направлена горизонтально, ось  $y$  направлена вертикально. Обозначим  $x_{\max}$  и  $y_{\max}$  ширину и высоту области перекрытия  $I_o$ . Необходимо в области перекрытия  $I_o$  получить плавный переход между изображениями.

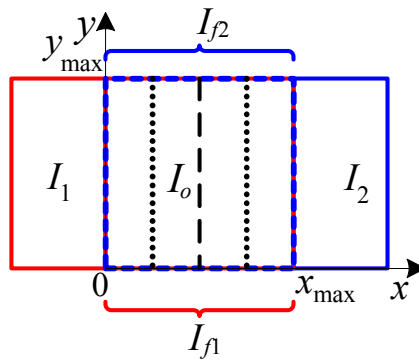


Рисунок 7.12 – Обозначения в области перекрытия двух изображений.

На первом этапе изображения  $I_{f1}$  и  $I_{f2}$  размываются с помощью ФНЧ:

$$I'_{f1} = g_a * I_{f1}, \quad (7.19)$$

$$I'_{f2} = g_a * I_{f2}. \quad (7.20)$$

Затем размытые части изображений  $I'_{f1}$  и  $I'_{f2}$  смешиваются с исходными  $I_{f1}$  и  $I_{f2}$  по альфа-функции  $\alpha_s$  так, чтобы у части левого изображения  $I_1$  была размыта область справа, а у части правого изображения  $I_2$  была размыта область слева:

$$I''_{f1}(x, y) = \alpha_s(x)I'_{f1}(x, y) + (1 - \alpha_s(x))I_{f1}(x, y), \quad (7.21)$$

$$I''_{f2}(x, y) = (1 - \alpha_s(x))I'_{f2}(x, y) + \alpha_s(x)I_{f2}(x, y), \quad (7.22)$$

где функция  $\alpha_s$  слегка меняет свой вид для каждой пары смешиваемых изображений за счёт псевдослучайного параметра  $\beta \in [0, 3]$ :

$$\alpha_s(x) = \frac{2^{x^{(1+\beta)}} - 1}{x_{\max}}. \quad (7.23)$$

Изображение в области перекрытия  $I_o$  образуется в результате смешивания  $I''_{f1}$  и  $I''_{f2}$  с помощью альфа-канала  $\alpha_f$ , который обеспечивает фигурную в виде сплайна границу между смешиваемыми изображениями:

$$I_o(x, y) = \alpha_f(x, y)I''_{f1} + (1 - \alpha_f(x, y))I''_{f2}, \quad (7.24)$$

где

$$\alpha_f(x, y) = \begin{cases} 1, & x \leq S_{cr}(x, y), \\ 0, & \text{иначе,} \end{cases} \quad (7.25)$$

где  $S_{cr}(x, y)$  – составная кубическая кривая Кэтмула-Рома [8] строится по точкам  $P_i, i = 0..N_p$ :

$$\begin{aligned} P_0 &= (0; 0), P_1 = (0,5x_{\max}; 0), \\ P_{N_p-1} &= (0,5x_{\max}, y_{\max}), P_{N_p} = (0, y_{\max}), \end{aligned} \quad (7.26)$$

$$P_{ix} = 0,25x_{\max} + v(0,5x_{\max}), P_{iy} = P_{(i-1)y} + 0,05Hy_{\max} + v(0,05Hy_{\max}),$$

где  $H$  – высота смешиваемых изображений; функция  $v(x)$  возвращает псевдослучайное число в диапазоне от 0 до  $x$ .

В параметрическом виде для параметра  $t$  элементарные кубические кривые Кэтмула-Рома, из которых состоит составная кривая, задаются уравнением:

$$S_{cr}(t) = \frac{1}{2} \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -5 & 4 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_{i-2} \\ P_{i-1} \\ P_i \\ P_{i+1} \end{bmatrix}. \quad (7.27)$$

Рисунок 7.13 иллюстрирует этапы создания плавного перехода между изображениями. За счёт использования псевдослучайных параметров области перекрытия различных пар изображений отличаются друг от друга.

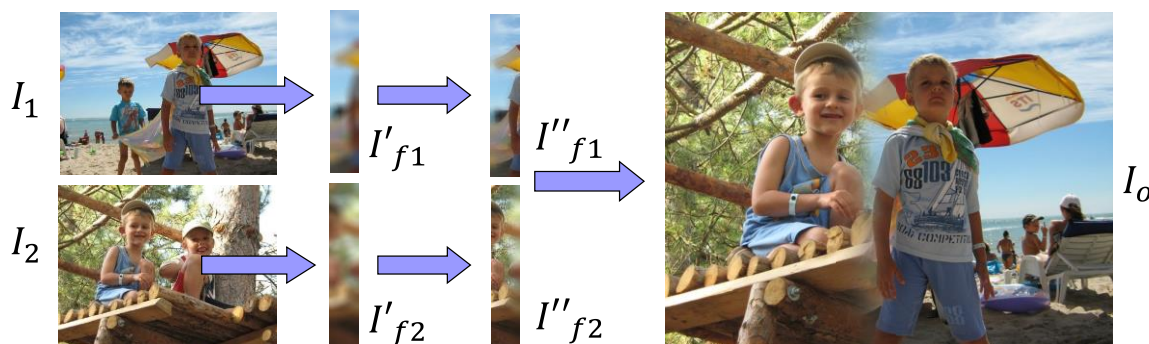


Рисунок 7.13 – Иллюстрация алгоритма создания плавного перехода между изображениями.

Формулы (с 7.19 по 7.22 и 7.24) применяются для каждого из RGB каналов изображения. Плавный переход между перекрывающимися по горизонтальной стороне изображениями создаётся аналогично.

### Результаты

Для сравнительной оценки способов автоматического формирования коллажей для 5 наборов, содержащих 7 или 8 фотографий, были созданы коллажи выкладыванием «плиткой» [353], с поворотами изображения алгоритмом Picture collage [355], с плавными переходами алгоритмом AutoCollage [324] и предлагаемым способом. Был проведён опрос, в котором участвовало 22 человека. Участники опроса были проинструктированы для каждого из пяти наборов фотографий ранжировать пять коллажей в порядке убывания их привлекательности для пользователя. Помимо коллажей интервьюируемые видели исходные фотографии, и им рекомендовалось обращать внимание на ухудшение их композиции в силу обрезания или перекрытия частей фотографий. Участники опроса не были информированы об использованных способах создания коллажей. Самый привлекательный коллаж получал оценку  $E$  в 4 балла, следующий 3, и так далее до 1.

Таблица 7.2 содержит оценки  $\bar{E}$ , усреднённые по всем участвовавшим в опросе и пяти наборам фотографий. Предлагаемый способ имеет самую высокую среднюю оценку. Участвовавшими в опросе было отмечено, что он обеспечивает

визуально приятные коллажи и, как правило, не портит важные элементы сцены. Высокие оценки получили коллажи, сгенерированные алгоритмом Picture collage несмотря на то, что иногда страдала композиция из-за обрезания и перекрытия фрагментов изображения. Формируемая данным способом белая рамка многими пользователями отмечалась как визуально привлекательное решение. Результаты алгоритма AutoCollage значительно различались для разных наборов фотографий. В четырёх случаях важные элементы фотографий были перекрыты или обрезаны, что повлекло низкие оценки. Но для пятого набора коллаж был оценен как самый привлекательный. В целом, результат работы AutoCollage сильно зависит как от содержимого фотографий, так и от соотношения их сторон. Выкладывание «плиткой» отмечено пользователями как наименее привлекательное визуальное решение создания коллажа. Коллажи, сформированные сравниваемыми способами для первого из наборов фотографий, показаны на рисунке 7.14.

Таблица 7.2. Средние оценки способов формирования коллажа.

Способ	$\bar{E}$
Выкладывание «плиткой» [353]	1,6
Picture collage [355]	3,0
AutoCollage [324]	2,0
Предлагаемый	<b>3,4</b>

Время работы во встроенном ПО МФП предложенного алгоритма создания плавного перехода между изображениями для случая 7 фотографий, страницы размера А4 и разрешения печати 600 dpi составляет немногим более 1 с.





a)



b)



c)



d)

Рисунок 7.14 – Коллажи, сформированные различными способами: а) [353]; б) [355]; в) [324]; д) предлагаемый способ.



### Выводы по седьмой главе

Предложены принципы автоматического выбора и размещения фотографий для генерации коллажа и страницы фотокниги. Выбор выполняется путём исключения некачественных фотографий, кластеризации изображений и их ранжирования с помощью построения карты значимости и детектирования лиц.

Для реализации алгоритма безреференсной оценки уровня шума предложен способ перемножения прямоугольных матриц 32-разрядных чисел с фиксированной точкой с помощью 16-разрядных векторных инструкций. Этот алгоритм обобщает способ умножения Карацубы для матриц чисел с фиксированной точкой и использует тензорное произведение векторов вместо скалярного. При умножении матриц размера 64 на 64 предлагаемый алгоритм более чем в 5 раз быстрее реализации с помощью 32-разрядных скалярных операций.

При размещении решается задача оптимизации, в ходе которой определяются коэффициенты масштабирования изображений. Комбинированный критерий оптимизации направлен на минимизацию незанятого на странице пространства и разницы в размерах размещённых на странице фотографий.

Предложен алгоритм создания плавного перехода между изображениями при автоматической генерации коллажа. Создаваемый коллаж выглядит привлекательным для пользователя. За счёт использования псевдослучайных параметров области перекрытия различных пар изображений отличаются друг от друга.

На основе предложенных принципов разработаны новые методы выбора изображений и формирования коллажа, которые по результатам проведённых экспериментов превосходят существующие.

## Заключение

Диссертация является научно-квалификационной работой, в которой *изложены новые научно обоснованные технические и организационные решения по разработке методов обработки изображений для систем сканирования и печати, внедрение которых вносит значительный вклад в развитие страны, что подтверждается 31 патентом на изобретение, выданным Федеральной службой по интеллектуальной собственности Российской Федерации «Роспатент».*

Достигнута цель диссертационного исследования: разработан и внедрён комплекс из более чем 20 режимов работы, удовлетворяющих требованиям к качеству и производительности в условиях ограничений, накладываемых аппаратным обеспечением многофункциональных принтеров, что значительно улучшает потребительские свойства информационных систем сканирования и печати.

В диссертации решена общая научная задача исследования: существенным образом развит научно-методический аппарат разработки и реализации методов обработки изображений для создания комплекса новых и модернизированных решений в системах сканирования и печати в условиях существующих ограничений ресурсов и требований к качеству и производительности. В работе получены следующие результаты, соответствующие частным научным и прикладным задачам:

1. Предложена оригинальная спиральная методология НИР при создании решений для МФП, основанная на анализе и управлении рисками. Впервые выявлены и ранжированы по важности факторы, которые препятствуют реализации результата НИР в продукте. На стадии НИР данные факторы можно рассматривать как риски. Использование методологии позволило в 3 раза повысить количество результатов НИР внедряемых в системное или встроенное ПО МФП.
2. Предложена методика (совокупность методов и последовательность их применения) режимов сканирования, которая включает этапы устранения

основных дефектов сканированных изображений, проведение сегментации, сохранение изображения в метафайл. Предложены способы оценки и коррекции скоса без использования дополнительной памяти, подавления печатного раstra, улучшения локального контраста, сегментации изображения на области фона, текста и картинок. На основе методики разработаны новые режимы:

- сканирования и сохранения документа в PDF с векторизацией контуров символов; режим обеспечивает два цикла повторной печати и сканирования без существенной потери качества распознавания текста; при сходном визуальном качестве режим создаёт файл наименьшего размера, по сравнению с существующими способами;
- копирования двухсторонних карточек на одну страницу с исправлением скоса и идентичной ориентацией обеих сторон; для 99 % карточек, стороны которых развёрнуты относительно друг друга на 180 градусов, ориентация определяется и исправляется.

3. Сформулирован подход к реализации специальных режимов печати путём локальной модификации кода процессора растровых изображений. Определены модули в процессоре растровых изображений Postscript для встраивания режимов печати в существующее программное обеспечение устройств печати. На основе подхода во встроенное ПО существующих устройств печати добавлены следующие новые функции:

- режим печати микротекста, позволяющий автоматически «на лету» генерировать микросимволы размером около 1 типографского пункта для широкого набора языков без использования специальных шрифтов; добавление на страницу микротекста увеличивает время печати на 0,01 с, что незаметно для пользователя;
- режим экономичной печати, позволяющий сократить использование краски на 13,4 % и обеспечивающий лучшее визуальное качество символов по сравнению с альтернативными способами;

- режим черновой печати, в котором картинки преобразуются в цветные эскизы, при этом экономится до 80 % тонера, а время работы режима превышает время обычной печати лишь на 0,5 %;
- режим добавления на страницу скрытых цифровых водяных знаков для сохранения идентификационного кода устройства; за счёт многократного дублирования информации способ позволяет извлечь скрытую информацию из копии и половины страницы.

4. Предложен подход к автоматической оценке и улучшению качества печатаемых фотографий, учитывающий размеры бумажного отпечатка, разрешение и технологию печати; на основе подхода разработаны новые методы оценки и повышения резкости изображения, детектирования и коррекции «красных глаз», оценки и улучшения затемнённых областей; по результатам проведённых экспериментов разработанные методы превосходят существующие, позволяют по меньшей мере в 10 раз сократить время подготовки фотографий к печати, повысить качество напечатанного изображения и предотвратить печать более половины испорченных фотографий.

5. Предложена группа преобразований и последовательность их применения для изменения размеров и соотношения сторон документов и фотографий без модификации основных смысловых элементов изображения; разработанные преобразования предоставляют новые возможности при копировании документов с изменением размера бумаги и печати фотографий без полей; предложены новые алгоритмы для переформатирования изображений за счёт удаления или добавления малозаметных участков, дополнения изображений симметрично отраженными полосами и обрезания полос с двух противоположных сторон, где размеры полос выбираются путём оценки заметности отражения; применение предложенных преобразований позволяет избежать заметных искажений для более чем 50% фотографий, которые ухудшались при печати без полей.

6. Предложены принципы выбора и размещения фотографий для автоматической генерации коллажа и страницы фотокниги, где выбор выполняется путём кластеризации качественных изображений и их ранжирования с помощью карты значимости, при размещении решается задача оптимизации, в ходе которой определяются коэффициенты масштабирования изображений; на основе предложенных принципов разработаны новые методы выбора фотографий и формирования коллажа, которые по результатам проведённых опросов превосходят существующие.

**Благодарности.** Автор глубоко признателен научному консультанту И.А. Матвееву за плодотворные обсуждения работы. Автор благодарен коллегам и соавторам, особенно И.В. Курилину, М.Н. Рычагову, Е.В. Толстой, С.Ю. Яковлеву, И.Р. Аликберову, М.А. Егоровой, Е.В. Потаповой, А.М. Вилькину. Огромное спасибо семье и родителям за поддержку.

### Список сокращений и обозначений

АБГШ – аддитивный белый гауссов шум

ИСО – международная организация по стандартизации

ПК – персональный компьютер

ПО – программное обеспечение

МАИ – метод анализа иерархий

МКО – международная комиссия по освещению

МНК – метод наименьших квадратов

МФП – многофункциональный принтер

МФУ – многофункциональное устройство

НИР – научно-исследовательская работа

ОКР – опытно-конструкторская работа

ОС – операционная система

ТЗ – техническое задание

ФВЧ – фильтр верхних частот

ФНЧ – фильтр нижних частот

ЦВЗ – цифровой водяной знак

3D – three dimensional

ADF – automatic document feeder

AHP – analytic hierarchy process

AIO – all-in-one

ASIC – application-specific integrated circuit

AUC – area under curve

bpp – bits per pixel

BRIC – blocked recursive image composition

CART – classification and regression tree

CIE – международная комиссия по освещению

CMY – cyan, magenta, yellow

CMYK – cyan, magenta, yellow, black

DLA – document layout analysis

DOG – difference of Gaussians

DSS – digital sharpness scale

dpi – dots per inch

DSP – digital signal processor

EFTP – effective throughput

ESAT – estimated saturated throughput

EXIF – exchangeable image file format

FN – false negative

FP – false positive

FSOT – first set out time

GRNN – general regression neural network

HDF – hardware-friendly descreening

HDR – high dynamic range

HPGL – Hewlett-Packard graphics language

HSV – hue, saturation, value

ICDAR – International conference on document analysis and recognition

ID card – identification card

in-place – обозначение алгоритма, работающего без выделения памяти

in-situ – обозначение алгоритма, работающего без выделения памяти

ISO – international standard organization

JFIF – JPEG file interchange format

JPEG – Joint Photographic Experts Group

kb - kilobyte

LED – light-emitting diode

MAE – mean absolute error

MedAE – median absolute error

MFD – multifunctional device  
MFP – multifunctional printer  
MIC – machine identification code  
MOS – mean opinion score  
MRC – mixed raster content  
MSE – mean squared error  
MSR – multiscale Retinex  
OCR – optical character recognition  
PDF – portable document format  
PS – Postscript  
RGB – red, green, blue  
RIP – raster image processor  
ROC – receiver operating characteristic  
SDR – standard dynamic range  
SoC – system-on-chip  
SOHO – small office/home office  
SIMD – single instruction multiple data  
TWAIN – драйвер сканирования (это не аббревиатура)  
USB – universal serial bus  
XPS – XML paper specification



### Список литературы

1. Global printer market: market size, status and forecast to 2030. Verified market research, – 2022. – 150 с.
2. Толковый словарь по вычислительным системам/ Под ред. В. Иллингуорта, Э.Л. Глейзера, И.К. Пайла, М: Машиностроение, – 1991, – 560 с.
3. Большой толковый словарь русского языка / Под ред. С.А. Кузнецова, СПб: Норинт, – 1998, – 1534 с.
4. Shiraishi Y., Imada T., Sato N., Imaizumi H. Multi-function Technology for FAX/MFP Devices //OKI Technical Review, Special Issue on Printers. – 1998. – Т. 64. – С. 81-84.
5. Global printer market (2021 - 2026). Mordor Intelligence. – 2020. – 35 с.
6. ISO 80000-2 : Quantities and Units: Part 2: Mathematical Signs and Symbols to be Used in the Natural Sciences and Technology. International Organization for Standardization, – 2009. – 46 с.
7. Абламейко С.В., Лагуновский Д.М., Минск: Амалфея, – 2000, – 304 с.
8. Foley, J.D., Van, F.D., van Dam, A., Feiner, S.K., Hughes, J.F. Computer graphics: principles and practice (2nd ed.). – Addison-Wesley Professional, – 1995, –1200 с.
9. Воройский Ф.С. Информатика. Энциклопедический словарь-справочник: введение в современные информационные и телекоммуникационные технологии в терминах и фактах. М:ФИЗМАТЛИТ, –2006, –768 с.
10. ISO 32000-2:2020 Document management: Portable document format: Part 2: PDF 2.0. International Organization for Standardization, – 2020. – 986 с.
11. Tai M., Liao S., Yang R. Y. H. Scanning and printing resolution //The 3rd International Conference on Information Sciences and Interaction Sciences. – IEEE, 2010. – с. 708-710.
12. Kipphan H. (ed.). Handbook of print media: technologies and production methods. – Springer Berlin Heidelberg, – 2014. – 1207 с.

13. Lau, D.L., Arce, G.R. Modern Digital Halftoning (2nd ed.). – CRC Press, – 2008. – 686 с.
14. Vrhel M., Saber E., Trussell H. J. Color image generation and display technologies //IEEE Signal Processing Magazine. – 2005. – Т. 22. – №. 1. – с. 23-33.
15. Epson Expression Photo XP-950 specification, Epson America, – 2013. – 4 с.
16. Canon ImagePROGRAF PRO-1000 specification, Canon Europe, – 2015. – 2 с.
17. Sharma A. Understanding Color Management (2nd ed.). – Wiley, – 2018. – 344 с.
18. Specification ICC.1:2022 (Profile version 4.4.0.0): Image technology colour management - Architecture, profile format, and data structure. – International color consortium, – 2022. – 126 с.
19. ISO 216:2007 Writing paper and certain classes of printed matter — Trimmed sizes — A and B series, and indication of machine direction. International Organization for Standardization, – 2007. – 9 с.
20. Microsoft. V4 printer driver. – URL: <https://learn.microsoft.com/en-us/windows-hardware/drivers/print/v4-printer-driver> (дата обращения: 30.01.2023). – Текст: электронный.
21. Adobe Systems Inc. PostScript language reference. – Addison-Wesley Longman Publishing Co., Inc., – 1999. – 897 с.
22. Hewlett-Packard Company. The HP-GL/2 and HP RTL Reference Guide: A Handbook for Program Developers. – Addison Wesley, – 1997. – 516 с.
23. ISO/IEC 10918-1:1994 Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines. International Organization for Standardization, – 1994. – 182 с.
24. Lee D.H., Min S.G. Network Scanning on Multi Function Device //2009 First International Conference on Advances in Multimedia. – IEEE, – 2009. – с. 116-120.
25. ISO/IEC 16485:2000 Information technology — Mixed Raster Content (MRC). International Organization for Standardization, – 2000. – 34 с.

26. Brother All-In-One Mono Laser Printer MFC-L5700DN specification. Brother. – 2016. – 6 с.
27. Yoshikawa E. Image processing method, printing method, image processor, and printing apparatus: пат. 10414167 США. № 20180207947; заявл. 26.07.2018; опубл. 17.09.2019.
28. Samsung Digital Color Laser MFP CLX-3160: Service manual. Samsung Electronics. – 2006. – 204 с.
29. ISO/IEC 24734:2021 Information technology — Office equipment — Method for measuring digital printing productivity. International Organization for Standardization, – 2021. – 61 с.
30. ISO/IEC 29183:2021 Information technology — Office equipment — Method for measuring digital copying productivity for a single one-sided original, – 2021. – 31 с.
31. Richter K. Colour, Colour Vision and Elementary Colours in Colour Information Technology. – Berlin University of Technology, – 2012. 85 с.
32. ISO/IEC 15775:2022 Information technology — Office equipment — Method of specifying image reproduction of colour copying machines and multifunction devices with copying modes by printed test charts. International Organization for Standardization, – 2022. – 65 с.
33. Marius P. Image quality metrics for the evaluation of printing workflows: дис. – Ph. D. thesis (University of Oslo, 2011), – 2011. – 377 с.
34. Clark J.H. The Ishihara test for color blindness //American Journal of Physiological Optics. – 1924.
35. Dugay F., Farup I., Hardeberg J. Y. Perceptual evaluation of color gamut mapping algorithms //Color Research & Application: Endorsed by Inter-Society Color Council. – 2008. – Т. 33. – №. 6. – с. 470-476.
36. Mantiuk R.K., Tomaszewska A., Mantiuk R. Comparison of four subjective methods for image quality assessment //Computer graphics forum. – Oxford, UK: Blackwell Publishing Ltd, 2012. – Т. 31. – №. 8. – с. 2478-2491.

37. Большая советская энциклопедия. в 30-ти т. – 3-е изд. – М.: Советская энциклопедия, 1969 - 1986.
38. Липчиу Н.В., Липчиу К.И. Методология научного исследования: учебное пособие //Краснодар: КубГАУ. – 2013. – 290 с.
39. ГОСТ Р 15.101-2021 Система разработки и постановки продукции на производство. Порядок выполнения научно-исследовательских работ. М.: Российский институт стандартизации, – 2021. – 13 с.
40. ГОСТ Р 15.301-2016 Система разработки и постановки продукции на производство. Порядок разработки и постановки продукции на производство. М.: Стандартиформ, – 2016. – 15 с.
41. Перл И.А., Калёнова О.В. Введение в методологию программной инженерии: Учебное пособие. – СПб.: Университет ИТМО, 2019. – 53 с.
42. Гидроец М.О., Гришанова Л.И. Методологии разработки программного продукта //Системный анализ и логистика. – 2020. – №. 4. – с. 45-53.
43. Boehm B., Hansen W. J. Spiral development: Experience, principles, and refinements. – CARNEGIE-MELLON UNIV, SOFTWARE ENGINEERING INST, 2000. – 37 с.
44. Дмитриева С.И. Управление рисками НИОКР //Актуальные проблемы гуманитарных и естественных наук. – 2012. – №. 9. – с. 79-83.
45. Saaty T.L. Relative measurement and its generalization in decision making why pairwise comparisons are central in mathematics for the measurement of intangible factors the analytic hierarchy/network process //RACSAM-Revista de la Real Academia de Ciencias Exactas, Fisicas y Naturales. Serie A. Matematicas. – 2008. – Т. 102. – №. 2. – с. 251-318.
46. Сафонов И.В., Матвеев И.А. Методология научно-исследовательской работы при создании функций систем сканирования и печати // Системы и средства информатики, 2023. Т. 33. № 3. с. 76-84.

47. Savoia A. Pretotype it: Make sure you are building the right it before you build it right. //pretotyping.org. – 2011. – 71 с.
48. Virzi R.A., Sokolov J.L., Karis D. Usability problem identification using both low- and high-fidelity prototypes //Proceedings of the SIGCHI conference on human factors in computing systems. – 1996. – с. 236-243.
49. Рудаков П.И., Сафонов И.В. Обработка сигналов и изображений. MATLAB 5. х //М.: Диалог-МИФИ. – 2000. – 416 с.
50. Royster P. The Art of Scanning //Digital Commons / University of Nebraska-Lincoln, Institutional Repository Information. Т.67, – 2011. – 28 с.
51. Hull J.J. Document image skew detection: Survey and annotated bibliography //Document Analysis Systems II. – 1998. – с. 40-64.
52. Кузнецов Ю.В. Технология обработки изобразительной информации //СПб.: Петербургский институт печати, – 2002. – 312 с.
53. Safonov, I.V., Kurilin, I.V., Rychagov, M.N., Tolstaya, E.V. Document Image Processing for Scanning and Printing //Springer Nature, Switzerland, – 2019, – 306 с.
54. Zaghetto A., de Queiroz R.L. Pre- and postprocessing for multilayer compression of scanned documents //Journal of Electronic Imaging. – 2011. – Т. 20. – №. 4. – с. 1-10.
55. Гаврилов А.Р., Сафонов И.В. Система и способ копирования: пат. 2308166 Рос. Федерация. № 2004120488/09; заявл. 07.07.2004; опубл. 10.10.2007, Бюл. № 28.
56. Safonov I., Kurilin I. Deskew for card image scanning //Графикон'2011. – 2011. – с. 42-45.
57. Vil'kin A.M., Safonov I.V., Egorova M.A. Bottom-up document segmentation method based on textural features //Pattern Recognition and Image Analysis. – 2011. – Т. 21. – №. 3.– с. 565-568.
58. Vil'kin A.M., Safonov I.V., Egorova M.A. Algorithm for segmentation of documents based on texture features //Pattern recognition and image analysis. – 2013. – Т. 23. – с. 153-159.

59. Vil'kin A.M., Safonov I.V., Egorova M.A. Bottom-up page segmentation based on texture features //10th International conference on Pattern recognition and Image Analysis: New information technologies. – 2010. – Т. 2. с. 373-376.
60. Ефимов С.В., Рычагов М.Н., Сафонов И.В., Курилин И.В. Способ сегментации текста по цветовому признаку в процессе копирования: пат. 2368007 Рос. Федерация. № 2008111139/09; заявл. 25.03.2008; опубл. 20.09.2009, Бюл. № 26.
61. Kurilin I.V., Safonov, I.V., Lee, H., Kim S.H. Descreening of scanned images //Color Imaging XV: Displaying, Processing, Hardcopy, and Applications. – SPIE, 2010. – Т. 7528. – с. 124-135.
62. Сафонов И.В., Курилин И.В. Способ подавления раstra: пат. 2405279 Рос. Федерация. № 2008128581/09; заявл. 15.07.2008; опубл. 27.11.2010, Бюл. № 5.
63. Курилин И.В., Сафонов И.В. Способ адаптивного сглаживания для подавления растровой структуры изображения: пат. 2411584 Рос. Федерация. № 2008134165/28; заявл. 21.08.2008; опубл. 10.02.2011, Бюл. № 4.
64. Kurilin I., Safonov I., Rychagov M., Zavalishin, S., Han D.H., Kim S.H. Fast algorithm for visibility enhancement of the images with low local contrast //Color Imaging XX: Displaying, Processing, Hardcopy, and Applications. – SPIE, 2015. – Т. 9395. – с. 94-102.
65. Сафонов И.В. Система и способ преобразования тонов плоских и объемных изображений: пат. 2509366 Рос. Федерация. № 2012144223/08; заявл. 17.10.2012; опубл. 10.03.2014, Бюл. № 7.
66. Kurilin I.V., Safonov, I.V., Rychagov, M.N., Lee, H., Kim, S.H., Choi, D. Generation of PDF with vector symbols from scanned document //Image Quality and System Performance X. – SPIE, 2013. – Т. 8653. – с. 233-247.
67. Курилин И.В., Сафонов И.В., Ли Хокеун Способ преобразования растрового изображения в метафайл: пат. 2469400 Рос. Федерация. № 2011146617/08; заявл. 17.11.2011; опубл. 10.12.2012, Бюл. № 34.

68. Курилин И.В., Сафонов И.В. Способ и система улучшения текста при цифровом копировании печатных документов: пат. 2520407 Рос. Федерация. № 2012148763/08; заявл. 16.11.2012; опубл. 27.06.2014, Бюл. № 18.
69. Safonov I., Lee, H., Kim, S., Choi, D. Intellectual two-sided card copy //Графикон'2011. – 2011. – с. 38-41.
70. Сафонов И.В., Ли Х.К., Ким С.Х. Способ копирования двухсторонних оригиналов: пат. 2435217 Рос. Федерация. № 2009146134/08; заявл. 14.12.2009; опубл. 27.11.2011, Бюл. № 17.
71. Safonov I.V., Lee Hokeun, Kim SangHo Image forming apparatus and method of copying two-sided card thereof: пат. 8508811 США. № 20110141534; заявл. 16.06.2011; опубл. 13.08.2013.
72. Safonov I.V., Lee Hokeun, Kim SangHo Image forming apparatus and method of copying two-sided card thereof: пат. 8724189 США. № 20130293917; заявл. 07.11.2013; опубл. 13.05.2014.
73. Kurilin I.V., Safonov I.V., Rychagov M.N., Lee H., Kim S.H. High-performance automatic cropping and deskew of multiple objects on scanned images //Image Quality and System Performance XI. – SPIE, 2014. – Т. 9016. – с. 100-109.
74. Курилин И.В., Сафонов И.В., Ли Хокеун Способ редактирования статических цифровых комбинированных изображений, включающих в себя изображения нескольких объектов: пат. 2458396 Рос. Федерация. № 2011129843/08; заявл. 19.07.2011; опубл. 10.08.2012, Бюл. № 22.
75. Kurilin I.V., Lee Hokeun, Safonov I.V. Method of editing static digital combined images comprising images of multiple objects: пат. 9179035 США. № 20130021629; заявл. 24.01.2013; опубл. 03.11.2015.
76. Рычагов М.Н., Сафонов И.В. Система и способ сканирования: пат. 2298292 Рос. Федерация. № 2004129644/09; заявл. 11.10.2004; опубл. 27.04.2007, Бюл. № 12.

77. Сафонов И.В., Рычагов М.Н. Система и способ сканирования и копирования: пат. 2368091 Рос. Федерация № 2004135515/09; заявл. 06.12.2004; опубл. 20.05.2006, Бюл. № 14.
78. Гонсалес Р., Вудс Р. Цифровая обработка изображений. //М: Техносфера, – 2012. – 1104 с.
79. van Beusekom J., Shafait F., Breuel T.M. Resolution independent skew and orientation detection for document images //Document Recognition and Retrieval XVI. – SPIE, 2009. – Т. 7247. – с. 175-182.
80. Yuan B., Tan C.L. Fiducial line based skew estimation //Pattern Recognition. – 2005. – Т. 38. – №. 12. – с. 2333-2350.
81. Chou C.H., Chu S.Y., Chang F. Estimation of skew angles for scanned documents based on piecewise covering by parallelograms //Pattern Recognition. – 2007. – Т. 40. – №. 2. – с. 443-455.
82. Huang K., Chen Z., Yu M., Yan X., Yin A. An efficient document skew detection method using probability model and q test //Electronics. – 2019. – Т. 9. – №. 1. – с. 55.
83. Meng G., Pan C., Zheng N., Sun C. Skew estimation of document images using bagging //IEEE Transactions on image processing. – 2010. – Т. 19. – №. 7. – с. 1837-1846.
84. Ahmad R., Naz S., Razzak I. Efficient skew detection and correction in scanned document images through clustering of probabilistic hough transforms //Pattern Recognition Letters. – 2021. – Т. 152. – С. 93-99.
85. Otsu N. A threshold selection method from gray-level histograms //IEEE transactions on systems, man, and cybernetics. – 1979. – Т. 9. – №. 1. – с. 62-66.
86. Catmull E., Smith A. R. 3-D transformations of images in scanline order //ACM SIGGRAPH Computer Graphics. – 1980. – Т. 14. – №. 3. – с. 279-285.
87. Mao S., Rosenfeld A., Kanungo T. Document structure analysis algorithms: a literature survey //Document recognition and retrieval X. – 2003. – Т. 5010. – с. 197-207.



88. Pak I., Teh P.L. Text segmentation techniques: a critical review //Innovative Computing, Optimization and Its Applications: Modelling and Simulations. – 2018. – с. 167-181.
89. Binmakhashen G.M., Mahmoud S.A. Document layout analysis: a comprehensive survey //ACM Computing Surveys (CSUR). – 2019. – Т. 52. – №. 6. – с. 1-36.
90. Baird H.S., Moll M.A., An C., Casey M.R. Document image content inventories //Document Recognition and Retrieval XIV. – SPIE, 2007. – Т. 6500. – с. 285-296.
91. Cesarini F., Gori M., Marinai S., Soda G. Structured document segmentation and representation by the modified XY tree //Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR'99. – IEEE, 1999. – с. 563-566.
92. Eskenazi S., Gomez-Krämer P., Ogier J.M. A comprehensive survey of mostly textual document segmentation algorithms since 2008 //Pattern Recognition. – 2017. – Т. 64. – с. 1-14.
93. Kosaraju S.C., Masum M., Tsaku N.Z., Patel P., Bayramoglu T., Modgil G., Kang M. DoT-Net: Document layout classification using texture-based CNN //2019 International Conference on Document Analysis and Recognition (ICDAR). – IEEE, 2019. – с. 1029-1034.
94. Safonov I.V., Bucha V.V. Animated thumbnail for still image //ГрафиКон'2010. – 2010. – с. 79-86.
95. Safonov I.V., Kornilov A.S., Reimers I.A. An Animated Graphical Abstract for an Image // Smart Algorithms for Multimedia and Imaging. – Springer, 2021. – с. 351–372.
96. Вилькин А.М., Сафонов И.В., Егорова М.А. Алгоритм сегментации документов //DSPА: Вопросы применения цифровой обработки сигналов. – 2011. – Т. 1. – №. 3. – с. 112-115.
97. Yates R. Fixed-point arithmetic: An introduction //Digital Signal Labs. – 2009. – Т. 81. – №. 83. – 198 с.
98. Шапиро Л., Стокман Д. //М.:БИНОМ. Лаборатория знаний. – 2006. 752 с.

99. Рашка С. Python и машинное обучение. //М: ДМК Пресс, – 2017. – 418 с.
100. Freund Y., Schapire R.E. Experiments with a new boosting algorithm //ICML. – 1996. – Т. 96. – с. 148-156.
101. Sokolova M., Lapalme G. A systematic analysis of performance measures for classification tasks //Information processing & management. – 2009. – Т. 45. – №. 4. – с. 427-437.
102. Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection //Ijcai. – 1995. – Т. 14. – №. 2. – с. 1137-1145.
103. Cawley G. C., Talbot N. L. C. On over-fitting in model selection and subsequent selection bias in performance evaluation //The Journal of Machine Learning Research. – 2010. – Т. 11. – с. 2079-2107.
104. Filzmoser P., Liebmann B., Varmuza K. Repeated double cross validation //Journal of Chemometrics: A Journal of the Chemometrics Society. – 2009. – Т. 23. – №. 4. – с. 160-171.
105. Friedman J., Hastie T., Tibshirani R. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors) //The annals of statistics. – 2000. – Т. 28. – №. 2. – с. 337-407.
106. Vezhnevets A., Vezhnevets V. Modest AdaBoost-teaching AdaBoost to generalize better //Graphicon. – 2005. – Т. 12. – №. 5. – с. 987-997.
107. Неделько В.М. К вопросу об эффективности бустинга в задаче классификации //Сибирский журнал чистой и прикладной математики. – 2015. – Т. 15. – №. 2. – с. 72-89.
108. Вилькин А.М., Сафонов И. В., Егорова М.А. Использование Matlab для оценки качества сегментации документов //Международная конференция Проектирование инженерных и научных приложений в среде MATLAB. – 2011. – Т. 1. – с. 442-445.

109. Antonacopoulos A., Bridson D. Performance analysis framework for layout analysis methods //Ninth International Conference on Document Analysis and Recognition (ICDAR 2007). – IEEE, 2007. – Т. 2. – с. 1258-1262.
110. Siddiqui H., Bouman C.A. Training-based algorithm for moiré suppression in scanned halftone images //Computational Imaging V. – SPIE, 2007. – Т. 6498. – с. 423-431.
111. Aizenberg I.N., Butakoff C. Frequency domain median like filter for periodic and quasi-periodic noise removal //Image Processing: Algorithms and Systems. – SPIE, 2002. – Т. 4667. – с. 181-191.
112. Tholeti T., Ganesh P., Ramanujam P. Frequency domain filtering techniques of halftone images //2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN). – IEEE, 2015. – с. 427-430.
113. Sattva Descreen plug-in for Adobe Photoshop, <http://www.descreen.net/> (дата обращения 02.04.2023)
114. Siddiqui H., Bouman C. A. Training-based descreening //IEEE transactions on image processing. – 2007. – Т. 16. – №. 3. – с. 789-802.
115. Chen F., Li S., Xu L., Sun B., Sun J. A Two-Step Adaptive Descreening Method for Scanned Halftone Image //Pattern Recognition: 6th Chinese Conference, 2014. Proceedings, Part II 6. – Springer, 2014. – с. 97-106.
116. Sun B., Li S., Sun J. Scanned image descreening with image redundancy and adaptive filtering //IEEE transactions on image processing. – 2014. – Т. 23. – №. 8. – с. 3698-3710.
117. Pawar A., Bodkhe S. Scanned color image improvement using Image descreening //2015 International Conference on Pervasive Computing (ICPC). – IEEE, 2015. – C. 1-4.
118. Siddiqui H., Boutin M., Bouman C. A. Hardware-friendly descreening //IEEE transactions on image processing. – 2009. – Т. 19. – №. 3. – с. 746-757.

119. Perona P., Malik J. Scale-space and edge detection using anisotropic diffusion //IEEE Transactions on pattern analysis and machine intelligence. – 1990. – Т. 12. – №. 7. – с. 629-639.
120. Schavemaker J.G.M., Reinders M.J., Gerbrands J.J., Backer E. Image sharpening by morphological filtering //Pattern Recognition. – 2000. – Т. 33. – №. 6. – с. 997-1012.
121. Safonov I.V., Rychagov M.N., Kang K., Kim S.H. Adaptive sharpening of photos //Color Imaging XIII: Processing, Hardcopy, and Applications. – SPIE, 2008. – Т. 6807. – с. 256-267.
122. Кнут Д.Э. Искусство программирования (The Art of Computer Programming) //Получисленные алгоритмы. М.: Вильямс. – 2017. – Т. 2. – с. 832.
123. Tomasi C., Manduchi R. Bilateral filtering for gray and color images //Sixth international conference on computer vision. – IEEE, 1998. – с. 839-846.
124. Safonov I.V. Automatic correction of amateur photos damaged by backlighting //Proc. International Conference on Computer Graphics and Vision (GraphiCon 06). – 2006. – с. 80-89.
125. SilverFast: Scanner and Digital imaging software, <https://www.silverfast.com/> (дата обращения 02.04.2023)
126. Zavalishin S.S., Safonov I.V., Bekhtin Y.S., Gurov V.S. Parameters adaptation framework for local contrast enhancement //6th Mediterranean Conference on Embedded Computing (MECO). – IEEE, 2017. – с. 1-4.
127. Michelson A.A. Studies in optics. – Courier Corporation, – 1995. – с. 218.
128. Rahman Z., Jobson D.J., Woodell G.A. Retinex processing for automatic image enhancement //Journal of Electronic imaging. – 2004. – Т. 13. – №. 1. – с. 100-110.
129. Moroney N. Local color correction using non-linear masking //Color and Imaging conference. – Society for Imaging Science and Technology, – 2000. – Т. 2000. – №. 1. – с. 108-111.

130. Safonov I.V., Rychagov M.N., Kang K., Kim S.H. Automatic correction of exposure problems in photo printer //2006 IEEE International Symposium on Consumer Electronics. – IEEE, 2006. – с. 1-6.
131. Zuiderveld K. Contrast limited adaptive histogram equalization //Graphics gems. – 1994. – с. 474-485.
132. Canon Color Image Runner C2550: Corporate solution brochure // Canon USA. – 2013. – 12 с.
133. Izawa M. Function approximation processing method and image processing method: пат. 7873218 США. № 20050238244; заявл. 27.10.2005; опубли. 18.01.2011.
134. Douglas D.H., Peucker T.K. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature //Cartographica: the international journal for geographic information and geovisualization. – 1973. – Т. 10. – №. 2. – с. 112-122.
135. Shao L., Zhou H. Curve fitting with Bezier cubics //Graphical models and image processing. – 1996. – Т. 58. – №. 3. – с. 223-232.
136. Kolesnikov A., Franti P. A fast near-optimal min-# polygonal approximation of digitized curves //Proc. IASTED Int. Conf. on Automation, Control and Information Technology. – 2002. – с. 418-422.
137. Kolesnikov A., Franti P. Fast algorithm for multiple-objects min-ε problem //Proceedings 2003 International Conference on Image Processing. – IEEE, 2003. – Т. 1. – с. 221-224.
138. Kolesnikov A. Approximation of digitized curves with cubic Bézier splines //2010 IEEE International Conference on Image Processing. – IEEE, 2010. – с. 4285-4288.
139. Местецкий Л.М. Непрерывная морфология бинарных изображений: фигуры, скелеты, циркуляры. –М.:ФИЗМАТЛИТ, 2009. – 288 с.
140. ZLIB, <https://zlib.net/> (дата обращения 02.04.2023).
141. Independent JPEG Group, <http://www.ijg.org/> (дата обращения 02.04.2023).

142. Павлидис Т.С. Алгоритмы машинной графики и обработки изображений: учеб. пособие / М.: Радио и связь, 1986. – 454 с.
143. Михеев С.М., Курилин И.В., Сафонов И.В., Вилькин А.М. Способ и система преобразования моментального снимка экрана в метафайл: пат. 2534005 Рос. Федерация. № 2013104245/08; заявл. 01.02.2013; опубл. 27.11.2014, Бюл. № 33.
144. I.R.I.S. A Canon company, <https://www.irislink.com/> (дата обращения 02.04.2023).
145. Samsung Multi Functional Printer CLX-6200/62400: User's guide. Samsung Electronics. – 2008. – 109 с.
146. ISO 7810:2019 Identification cards — Physical characteristics, – 2019. – 12 с.
147. Usami A., Takaoka M., Funada M., Fukuoka S., Akiba R., Kanatsu T. Image forming apparatus and method: пат. 6473196 США. № 20010013938; заявл. 16.08.2001; опубл. 29.10.2002.
148. Cheriet M., Kharm N., Suen C., Liu C.L. Character recognition systems: a guide for students and practitioners. – John Wiley & Sons, 2007. – 325 с.
149. Abbyy FineReader, <https://pdf.abbyy.com/> (дата обращения 02.04.2023).
150. Wu P.C., Lin C.H. A Green and Practical Color Photograph Printing Technology Based on Color Difference Model and Human Perception //IEEE Access. – 2021. – Т. 10. – с. 649-666.
151. Seto K., Kawana T. Color image forming apparatus which reduces toner consumption during image formation: пат. 5646670 США. № 1994297834; заявл. 30.08.1994; опубл. 08.07.1997.
152. Ebner F.F., Narendranath N.N. Printer pixel size and toner/ink consumption reduction system: пат. 5946450 США. № 08/720101; заявл. 30.08.1994; опубл. 31.08.1999.
153. Robertson K.W., Taylor R.D., Zimmerman G.D. Method and apparatus for adjusting density in digital images: пат. 5483625 США. № 53577; заявл. 26.04.1993; опубл. 09.01.1996.

154. Davidson M.L., Hawkins B.C. Image forming device having a reduced toner consumption mode: пат. 6266153 США. № 09/076034; заявл. 12.05.1998; опубл. 24.07.2001.
155. Montrucchio B., Ferrero R. Toner savings based on quasi-random sequences and a perceptual study for green printing //IEEE Transactions on Image Processing. – 2016. – Т. 25. – №. 6. – с. 2635-2646.
156. EcoFont, <https://www.ecofont.com/> (дата обращения 15.06.2023).
157. Ryman Eco, <https://www.rymaneco.com/> (дата обращения 15.06.2023).
158. Imjai T., Wattanapanich, C., Madardam, U., Garcia, R. Analysis of Ink/Toner Savings of English and Thai Ecofonts for Sustainable Printing //Sustainability. – 2021. – Т. 13. – №. 7. – с. 4070.
159. Мересин В.М. Защищённая полиграфия. Справочник: учеб. пособие / М.: Флинта, 2012. – 640 с.
160. Bozhkova T., Spiridonov I., Shterev K. Overview of security printing types and trends in its future development //Bulg. Chem. Commun. – 2017. – Т. 49. – с. 195-201.
161. Choi J.H., Lee H.Y., Lee H.K. Color laser printer forensic based on noisy feature and support vector machine classifier //Multimedia Tools and Applications. – 2013. – Т. 67. – с. 363-382.
162. Low S.H., Maxemchuk N.F., Lapone A.M. Document identification for copyright protection using centroid detection //IEEE Transactions on Communications. – 1998. – Т. 46. – №. 3. – с. 372-383.
163. Huang D., Yan H. Interword distance changes represented by sine waves for watermarking text images //IEEE Transactions on Circuits and Systems for Video Technology. – 2001. – Т. 11. – №. 12. – с. 1237-1245.
164. Обыденков Д.О., Фролов А.Е., Маркин Ю.В., Фомин С.А., Кондратьев, Б.В. Методы маркирования текстовых документов при печати посредством вертикального сдвига и изменения яркости фрагментов слов //Труды Института системного программирования РАН. – 2021. – Т. 33. – №. 5. – с. 65-82.

165. Bhattacharjya A.K., Ancin H. Data embedding in text for a copier system //Proceedings 1999 International Conference on Image Processing. – IEEE, 1999. – T. 2. – c. 245-249.
166. Xiao C., Zhang C., Zheng C. Fontcode: Embedding information in text documents using glyph perturbation //ACM Transactions on Graphics (TOG). – 2018. – T. 37. – №. 2. – c. 1-16.
167. Tan L., Hu K., Zhou X., Chen R., Jiang W. Print-scan invariant text image watermarking for hardcopy document authentication //Multimedia Tools and Applications. – 2019. – T. 78. – c. 13189-13211.
168. Briffa J. A., Culhane C., Treharne H. Imperceptible printer dot watermarking for binary documents //Optics, Photonics, and Digital Technologies for Multimedia Applications. – SPIE, 2010. – T. 7723. – c. 166-174.
169. Hota H., Hikima T. An Overview and Applications of Val-Code®: Digital Watermarking Technology for Printed Documents //Oki Technical Review. – 2004. – T. 71. – №. 192. – c. 20-23.
170. Kuraki V., Nakagata V. Watermarking technologies for security-enhanced printed documents //Fujitsu Science and Technical Journal. – 2007. – T. 43. – №. 2. – c. 197-203.
171. Fan Z., Eschbach R., Wang S., Bala R. Security Printing in Digital Age // Proceedings of International Conference on Imaging Science and Hardcopy. – 2008. – c. 7-10.
172. Hains C.M., Low J.R., Eschbach R., McElvain J. Super resolution encoded microprinting through postscript: пат. 7965422 США. № 11/414,576; заявл. 28.04.2006; опубли. 21.06.2011.
173. Wang S., Liu C., Xu B. Glossmark technology: beyond halftone frequencies //Color Imaging X: Processing, Hardcopy, and Applications. – SPIE, 2005. – T. 5667. – c. 544-553.



174. Fan Z. Information embedding using two-layer conjugate screening //Optical Security and Counterfeit Deterrence Techniques V. – SPIE, 2004. – Т. 5310. – с. 170-175.
175. Bugnon T., Brichon M., Hersch R. D. Model-based deduction of CMYK surface coverages from visible and infrared spectral measurements of halftone prints //Color Imaging XII: Processing, Hardcopy, and Applications. – SPIE, 2007. – Т. 6493. – с. 324-333.
176. Richter T., Escher, S., Schönfeld, D., Strufe, T. Forensic analysis and anonymisation of printed documents //Proceedings of the 6th ACM Workshop on information hiding and multimedia security. – 2018. – с. 127-138.
177. Сафонов И.В., Яковлев С.Ю. Способ формирования символов для микропечати: пат. 2365510 Рос. Федерация. № 2008101807/12; заявл. 23.01.2008; опубл. 27.08.2009, Бюл. № 24.
178. Safonov I.V., Yakovlev S.Yu. Method of forming characters for microprint and image forming apparatus using the same: пат. 8248622 США. № 20090185210; заявл. 23.07.2009; опубл. 21.09.2012.
179. Safonov I.V., Yakovlev S.Yu. Method of forming characters for microprint and image forming apparatus using the same: пат. 8493575 США. № 20120268754; заявл. 25.10.2012; опубл. 23.07.2013.
180. Safonov I.V., Yakovlev S.Yu. Method of forming characters for microprint and image forming apparatus using the same: пат. 8736853 США. № 20130278971; заявл. 24.10.2013; опубл. 27.05.2014.
181. Safonov I.V., Kurilin I.V., Rychagov M.N., Lee H., Kim S., Choi D. Fast approach for toner saving //Color Imaging XVI: Displaying, Processing, Hardcopy, and Applications. – SPIE, 2011. – Т. 7866. – с. 369-378.
182. Сафонов И.В., Яковлев С.Ю., Аликберов И.Р., Курилин И.В. Способ и система для снижения расхода краски при печати: пат. 2405201 Рос. Федерация. № 2009122721/09; заявл. 16.06.2009; опубл. 27.11.2010, Бюл. № 33.

183. Safonov I.V., Tolstaya E.V., Rychagov M.N., Lee H., Kim S.H., Choi D. Bio-inspired color sketch for eco-friendly printing //Color Imaging XVII: Displaying, Processing, Hardcopy, and Applications. – SPIE, 2012. – Т. 8292. – с. 369-378.
184. Сафонов И.В., Толстая Е.В. Способ черновой печати посредством преобразования растровых изображений в эскизы (варианты): пат. 2445677 Рос. Федерация. № 2010137868/08; заявл. 13.09.2010; опубл. 20.03.2012, Бюл. № 8.
185. Safonov I.V., Tolstaya E.V. Printing control device, imaging forming apparatus, and imaging forming method: пат. 8730526 США. № 20120062957; заявл. 15.03.2012; опубл. 20.05.2014.
186. Tolstaya E.V., Safonov I.V. Printing control device, imaging forming apparatus, and imaging forming method: пат. 9075548 США. № 20140218769; заявл. 07.08.2014; опубл. 07.07.2015.
187. Данилевич А.Б., Сафонов И.В. Система и способ черновой печати посредством преобразования картинок в контурные эскизы: пат. 2469399 Рос. Федерация. № 2011138830/08; заявл. 22.09.2011; опубл. 10.12.2012, Бюл. № 34.
188. Danilevich A.B., Safonov I.V. Print control apparatus, image forming apparatus, and method for forming image: пат. 9036203 США. № 20130077106; заявл. 28.03.2013; опубл. 19.05.2015.
189. Kurilin I.V., Safonov I.V., Rychagov M.N., Lee H., Kim S.H., Choi D.C. Embedding positional-independent hidden data into hardcopy //Pattern Recognition and Image Analysis. – 2011. – Т. 21. – с. 511-514.
190. Сафонов И.В., Курилин И.В. Способ внедрения скрытого цифрового сообщения в печатаемые документы и извлечения сообщения: пат. 2431192 Рос. Федерация. № 2010100795/08; заявл. 12.01.2010; опубл. 10.10.2011, Бюл. № 28.
191. Сафонов И.В., Курилин И.В. Способ и система встраивания и извлечения скрытых данных в печатаемых документах: пат. 2446464 Рос. Федерация. № 2010117994/08; заявл. 06.05.2010; опубл. 27.03.2012, Бюл. № 32.

192. Carter R., Meggs P.B., Day B., Maxa S., Sanders M. *Typographic Design: Form and Communication*, 6th ed. //Wiley, New Jersey, – 2014, – 352 c.
193. Rougier N.P. Higher quality 2D text rendering //Journal of Computer Graphics Techniques. – 2013. – T. 2. – №. 1. – c. 50-64.
194. Saeed K., Tabędzki M., Rybnik M., Adamski M. K3M: A universal algorithm for image skeletonization and a review of thinning techniques //International Journal of Applied Mathematics and Computer Science. – 2010. – T. 20. – №. 2. – c. 317-335.
195. Saha P.K., Borgefors G., di Baja G.S. A survey on skeletonization algorithms and their applications //Pattern recognition letters. – 2016. – T. 76. – c. 3-12.
196. Ahmed M., Ward R. A rotation invariant rule-based thinning algorithm for character recognition //IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2002. – T. 24. – №. 12. – c. 1672-1678.
197. Elarian Y., Ahmad I., Awaida S., Al-Khatib W., Zidouri A. Arabic ligatures: analysis and application in text recognition //2015 13th International Conference on Document Analysis and Recognition (ICDAR). – IEEE, 2015. – c. 896-900.
198. Bernal E., Trask J., Allebach J. Model-based memory-efficient algorithm for compensation of toner overdevelopment in electrophotographic printers //Journal of Imaging Science and Technology. – 2008. – T. 52. – №. 6. – c. 60504-1-60504-15.
199. Park H.J., Pizlo Z., Allebach J.P. Determination of optimal coring values from psychophysical experiments //Image Quality and System Performance VI. – SPIE, 2009. – T. 7242. – c. 191-200.
200. Mould D., Grant K. Stylized black and white images from photographs //Proceedings of the 6th international symposium on Non-photorealistic animation and rendering. – 2008. – c. 49-58.
201. Xu J., Kaplan C.S. Artistic thresholding //Proceedings of the 6th international symposium on Non-photorealistic animation and rendering. – 2008. – c. 39-47.

202. Wen F., Luan Q., Liang L., Xu Y.Q., Shum H.Y. Color sketch generation //Proceedings of the 4th international symposium on Non-photorealistic animation and rendering. – 2006. – C. 47-54.
203. Vonikakis V., Andreadis I., Papamarkos N., Gasteratos A. Adaptive document binarization-a human vision approach //International Conference on Computer Vision Theory and Applications. – SCITEPRESS, 2007. – T. 2. – c. 104-109.
204. Kang H., Lee S., Chui C.K. Coherent line drawing //Proceedings of the 5th international symposium on Non-photorealistic animation and rendering. – 2007. – C. 43-50.
205. Son M., Kang H., Lee Y., Lee S. Abstract line drawings from 2D images //15th Pacific Conference on Computer Graphics and Applications. – IEEE, 2007. – c. 333-342.
206. Zhang W., Li G., Ma H., Yu Y. Automatic color sketch generation using deep style transfer //IEEE computer graphics and applications. – 2019. – T. 39. – №. 2. – C. 26-37.
207. Liu Z.S., Siu W.C., Chan H.A. Learn to Sketch: A fast approach for universal photo sketch //2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). – IEEE, 2021. – c. 1450-1457.
208. Samsung ML-4550 Series: Monochrome Laser Printer, User's guide // Samsung Electronics, – 2006, – 113 c.
209. Safonov I.V., Kurilin I.V., Rychagov M.N., Tolstaya E.V. Adaptive Image Processing Algorithms for Printing //Springer Nature, Singapore, – 2018, – 304 c.
210. Ciancio A., da Silva E.A., Said A., Samadani R., Obrador P. No-reference blur assessment of digital pictures based on multifeature classifiers //IEEE Transactions on image processing. – 2010. – T. 20. – №. 1. – c. 64-75.
211. Keelan B. Handbook of image quality: characterization and prediction // CRC Press, –2002, – 544 c.
212. Ebrahimi F., Chamik M., Winkler S. JPEG vs. JPEG 2000: an objective comparison of image encoding quality //Applications of Digital Image Processing XXVII. – SPIE, 2004. – T. 5558. – c. 300-308.

213. Ватолин Д., Ракушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. — М.: ДИАЛОГ-МИФИ, 2003. — 384 с.
214. ISO 5800:1987 Photography — Colour negative films for still photography — Determination of ISO speed. International Organization for Standardization, — 1987. — 4 с.
215. Wang Z., Bovik A.C. Modern Image Quality Assessment //Springer Nature, Switzerland, — 2006, — 146 с.
216. Digital Still Camera Image File Format Standard (Exchangeable image file format for Digital Still Cameras: Exif) Version 2.1. Japan Electronic Industry Development Association (JEIDA) — 1998. — 166 с.
217. Yen J., Wu P., Tretter D. Knowledge discovery for better photographs //Multimedia Content Access: Algorithms and Systems. — SPIE, 2007. — Т. 6506. — с. 107-117.
218. Толстая Е.В., Сафонов И.В., Курилин И.В., Ефимов С.В., Рычагов М.Н. Способ повышения качества цифрового изображения на основе метаданных: пат. 2374688 Рос. Федерация. № 2008113577/09; заявл. 10.04.2008; опубл. 27.11.2009, Бюл. № 33.
219. Kurilin I.V., Safonov I.V., Tolstaya E.V., Efimov S.V., Rychagov M.N. EXIF-based Image Enhancement Pipeline // Proceedings of 9<sup>th</sup> International conference on Pattern recognition and Image Analysis: New information technologies. 2008. — с. 375-378.
220. Сафонов И.В. Система и способ коррекции изображения при выводе на устройство печати: пат. 2298293 Рос. Федерация. № 2004116860/09; заявл. 04.06.2004; опубл. 27.04.2007, Бюл. № 12.
221. Сафонов И.В. Способ предотвращения печати размытых фотографий: пат. 2338252 Рос. Федерация. № 2007108381/09; заявл. 06.03.2007; опубл. 10.11.2008, Бюл. № 31.

222. Сафонов И.В. Способ адаптивного повышения резкости цифровых изображений в процессе печати: пат. 2383924 Рос. Федерация. № 2007113703/09; заявл. 12.04.2007; опубл. 10.03.2010, Бюл. № 7.
223. Сафонов И.В. Способ автоматической коррекции эффекта красных глаз: пат. 2320011 Рос. Федерация. № 2006123847/09; заявл. 05.07.2006; опубл. 20.03.2008, Бюл. № 8.
224. Сафонов И.В. Способ автоматической коррекции эффекта красных глаз: пат. 2324225 Рос. Федерация. № 2006132154/09; заявл. 07.09.2006; опубл. 10.05.2008, Бюл. № 13.
225. Сафонов И.В., Гаврилов А.Р., Аликберов И.Р. Способ устранения эффекта красных глаз на цифровых цветных изображениях для пары глаз: пат. 2338255 Рос. Федерация. № 2006137150/09; заявл. 20.10.2006; опубл. 10.11.2008, Бюл. № 31.
226. Safonov I.V. Automatic red eye detection //17th International Conference on Computer Graphics and Vision GRAPHICON-2007, 2007. – с. 112-119.
227. Safonov I.V., Rychagov M.N., Kang K., Kim S.H. Automatic red eye correction and its quality metric //Color Imaging XIII: Processing, Hardcopy, and Applications. – SPIE, 2008. – Т. 6807. – с. 276-285.
228. Сафонов И.В. Система и способ коррекции темных оттенков на цифровых фотографиях: пат. 2298223 Рос. Федерация. № 2005112311/09; заявл. 25.04.2005; опубл. 27.04.2007, Бюл. № 12.
229. Safonov I.V. Method and apparatus for adjusting brightness of image: пат. 7702177 США. № 20060239583; заявл. 26.10.2006; опубл. 20.04.2010.
230. Сафонов И.В. Способ улучшения цифровых изображений: пат. 2298226 Рос. Федерация. № 2005133120/09; заявл. 28.10.2005; опубл. 27.04.2007, Бюл. № 12.
231. Рычагов М.Н., Сафонов И.В., Толстая Е.В., Ефимов С.В., Канг Ки-Мин, Ким Санг-Хо Способ повышения качества цифрового фотоизображения: пат. 2400815 Рос. Федерация. № 2006135550/09; заявл. 09.10.2006; опубл. 27.09.2010, Бюл. № 11.

232. Safonov I.V., Rychagov M.N., Kang K., Kim S.H. Automatic correction of exposure problems in photo printer //2006 IEEE International Symposium on Consumer Electronics. – IEEE, 2006. – с. 1-6.
233. Goncharova A.V., Safonov I.V., Romanov I.A. The regression model for the procedure of correction of photos damaged by backlighting //CEUR Workshop Proceedings. – 2019. – с. 326-333.
234. Патана Е.И., Сафонов И.В. Адаптивный способ создания и печати цветных анаглифных изображений: пат. 2481635 Рос. Федерация. № 2011132463/08; заявл. 02.08.2011; опубл. 10.05.2013, Бюл. № 13.
235. Patana E., Safonov I., Rychagov M. Adaptive generation of color anaglyph //Transactions on Computational Science XIX: Special Issue on Computer Graphics. – Springer Berlin Heidelberg, 2013. – с. 33-47.
236. Patana E., Safonov I., Rychagov M. Adaptive 3D color anaglyph generation for printing //The 22-nd International Conference on Computer Graphics and Vision: GraphiCon. – 2012. – с. 1-5.
237. Zhu M., Yu L., Wang Z., Ke Z., Zhi C. A Survey on Objective Evaluation of Image Sharpness //Applied Sciences. – 2023. – Т. 13. – №. 4. – с. 2652.
238. Zhang B., Allebach J. P., Pizlo Z. An investigation of perceived sharpness and sharpness metrics //Image Quality and System Performance II. – SPIE, 2005. – Т. 5668. – С. 98-110.
239. Zhan Y., Zhang R. No-reference image sharpness assessment based on maximum gradient and variability of gradients //IEEE Transactions on Multimedia. – 2017. – Т. 20. – №. 7. – с. 1796-1808.
240. Hu H., De Haan G. Low cost robust blur estimator //2006 International Conference on Image Processing. – IEEE, 2006. – с. 617-620.
241. Crete F., Dolmiere T., Ladret P., Nicolas M. The blur effect: perception and estimation with a new no-reference perceptual blur metric //Human vision and electronic imaging XII. – SPIE, 2007. – Т. 6492. – с. 196-206.

242. De K., Masilamani V. No-reference image sharpness measure using discrete cosine transform statistics and multivariate adaptive regression splines for robotic applications //Procedia computer science. – 2018. – Т. 133. – с. 268-275.
243. Baig M.A., Moinuddin A.A., Khan E., Ghanbari M. DFT-based no-reference quality assessment of blurred images //Multimedia Tools and Applications. – 2022. – Т. 81. – №. 6. – с. 7895-7916.
244. Li D., Jiang T., Jiang M. Exploiting high-level semantics for no-reference image quality assessment of realistic blur images //Proceedings of the 25th ACM international conference on Multimedia. – 2017. – с. 378-386.
245. Li L., Wu D., Wu J., Li H., Lin W., Kot A.C. Image sharpness assessment by sparse representation //IEEE Transactions on Multimedia. – 2016. – Т. 18. – №. 6. – с. 1085-1097.
246. Vu C.T., Phan T.D., Chandler D.M. S<sub>3</sub>: A spectral and spatial measure of local perceived sharpness in natural images //IEEE transactions on image processing. – 2011. – Т. 21. – №. 3. – с. 934-945.
247. Topper T.N., Jernigan M.E. On the informativeness of edges //Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics. – IEEE, 1989. – с. 909-914.
248. Lim S.H., Yen J., Wu P. Detection of out-of-focus digital photographs //Hewlett-Packard Laboratories Technical Report HPL. – 2005. 5 с.
249. Samsung Photo Printer User's Guide: SPP-2040/2020 series // Samsung Electronics, – 2005, – 81 с.
250. Nik Sharpener, <https://nikcollection.dxo.com/nik-sharpener/> (дата обращения 25.04.2023).
251. FocusMagic, <https://www.focusmagic.com/> (дата обращения 25.04.2023).
252. Ioffe S. Red eye detection with machine learning //Proceedings 2003 International Conference on Image Processing. – IEEE, 2003. – Т. 2. – с. II-871.



253. Zhang L., Sun Y., Li M., Zhang H. Automated red-eye detection and correction in digital photographs //2004 International Conference on Image Processing, 2004. ICIP'04. – IEEE, 2004. – T. 4. – с. 2363-2366.
254. Corcoran P., Bigioi P., Steinberg E., Pososin A. Automated in-camera detection of flash-eye defects //IEEE Transactions on Consumer Electronics. – 2005. – T. 51. – №. 1. – с. 11-17.
255. Hardeberg J.Y. Red eye removal using digital color image processing //PICS. – 2001. – с. 283-287.
256. Gasparini F., Schettini R. Automatic redeye removal for smart enhancement of photos of unknown origin //International Conference on Advances in Visual Information Systems. –Springer Berlin Heidelberg, 2005. – с. 226-233.
257. Smolka B., Czubin K., Hardeberg J.Y., Plataniotis K.N., Szczepanski M., Wojciechowski K. Towards automatic redeye effect removal //Pattern Recognition Letters. – 2003. – T. 24. – №. 11. – с. 1767-1785.
258. Luo H., Yen J., Tretter D. Detecting and correcting redeye in an image: пат. 7333653 США. № 20050047655; заявл. 29.08.2003; опубл. 19.02.2008.
259. Kumar V., Bhooshan S., Sood A., Shahi R., Mendiratta S. A novel technique for automated red eye detection and correction //2009 XXII International Symposium on Information, Communication and Automation Technologies. – IEEE, 2009. – с. 1-5.
260. Gaubatz M., Ulichney R. Automatic red-eye detection and correction //Proceedings. International Conference on Image Processing. – IEEE, 2002. – T. 1. – с. I-I.
261. Viola P., Jones M.J. Robust real-time face detection //International journal of computer vision. – 2004. – T. 57. – с. 137-154.
262. Schildkraut J.S., Gray R.T. A fully automatic redeye detection and correction algorithm //Proceedings. International Conference on Image Processing. – IEEE, 2002. – T. 1. – с. I-I.

263. Adams R., Bischof L. Seeded region growing //IEEE Transactions on pattern analysis and machine intelligence. – 1994. – Т. 16. – №. 6. – с. 641-647.
264. Yoo S., Park R.H. Red-eye detection and correction using inpainting in digital photographs //IEEE Transactions on Consumer Electronics. – 2009. – Т. 55. – №. 3. – с. 1006-1014.
265. Willamowski J., Csurka G. Probabilistic automatic red eye detection and correction //18th International Conference on Pattern Recognition (ICPR'06). – IEEE, 2006. – Т. 3. – с. 762-765.
266. Vandrotti B.S., Veldandi M., Govindarao K.A., Uliyar M., Mishra P. An efficient red eye reduction technique //2012 IEEE International Conference on Consumer Electronics (ICCE). – IEEE, 2012. – с. 37-40.
267. Харт К. Учимся рисовать лица и эмоции. Руководство по рисованию головы человека. Москва: Бомбора, – 2022, – 144 с.
268. Volken F., Terrier J., Vandewalle P. Automatic red-eye removal based on sclera and skin tone detection //Proc. IS&T; Third European Conference on Color in Graphics, Imaging and Vision (CGIV). – 2006. – с. 359-364.
269. Ali T., Khattak S., Kim I. Automatic red-eye effect removal using combined intensity and colour information //The Imaging Science Journal. – 2011. – Т. 59. – №. 1. – с. 8-16.
270. Ulichney R., Gaubatz M., Van Thong J.M. RedBot-A tool for improving red-eye correction //Proceedings of the IS&T/SID Eleventh Color Imaging Conference. Color Science and Engineering Systems, Technologies Applications. – 2003.
271. Steinberg E. Method and apparatus for the automatic real-time detection and correction of red-eye defects in batches of digital images or in handheld appliances: пат. 6873743 США. № 10/113871; заявл. 29.03.2002; опубл. 29.03.2005.
272. HP Photosmart 470 series: User's guide, Hewlett-Packard Development Company. – 2005. 70 с.

273. Fawcett T. An introduction to ROC analysis //Pattern recognition letters. – 2006. – Т. 27. – №. 8. – с. 861-874.
274. Методы компьютерной обработки изображений / под. ред. В.А. Сойфера. – 2-е изд. – М: ФИЗМАТЛИТ, 2003. – 784 с.
275. Horn B. Robot vision. – MIT press, 1986. – 480 с.
276. Land E.H. The retinex theory of color vision. — Scientific America., 1977. — с. 2-17.
277. Orsini G., Ramponi G., Carrai P., Di Federico R. A modified retinex for image contrast enhancement and dynamics control //Proceedings 2003 International Conference on Image Processing (Cat. No. 03CH37429). – IEEE, 2003. – Т. 3. – с. III-393.
278. Tao L., Asari V. Modified luminance based MSR for fast and efficient image enhancement //32nd Applied Imagery Pattern Recognition Workshop, 2003. Proceedings. – IEEE, 2003. – с. 174-179.
279. Wang W., Wu X., Yuan X., Gao Z. An experiment-based review of low-light image enhancement methods //IEEE Access. – 2020. – Т. 8. – с. 87884-87917.
280. Sobol R. Improving the Retinex algorithm for rendering wide dynamic range photographs //Journal of Electronic Imaging. – 2004. – Т. 13. – №. 1. – с. 65-74.
281. Brajovic V. Brightness perception, dynamic range and noise: a unifying model for adaptive image sensors //Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. – IEEE, 2004. – Т. 2. – с. II.
282. Fattal R., Lischinski D., Werman M. Gradient domain high dynamic range compression //Seminal Graphics Papers: Pushing the Boundaries, Volume 2. – 2023. – с. 671-678.
283. Ou Y., Ambalathankandy P., Takamaeda S., Motomura M., Asai T., Ikebe M. Real-time tone mapping: a survey and cross-implementation hardware benchmark //IEEE Transactions on Circuits and Systems for Video Technology. – 2021. – Т. 32. – №. 5. – с. 2666-2686.

284. Lv X., Zhang S., Liu Q., Xie H., Zhong B., Zhou H., BacklitNet: A dataset and network for backlit image enhancement //Computer Vision and Image Understanding. – 2022. – Т. 218. – С. 103403.
285. Bose S., Nawale S., Khut D., Kolekar M.H. LumiNet: Multi-Spatial Attention Generative Adversarial Network for Backlit Image Enhancement //IEEE Transactions on Instrumentation and Measurement. – 2023.
286. Зоркальцев В. И. Метод наименьших квадратов. – Новосибирское отделение издательства Наука, 1995. – 220 с.
287. Drucker H., Burges C.J., Kaufman L., Smola A., Vapnik V. Support vector regression machines //Advances in neural information processing systems. – 1996. – Т.9.
288. Breiman L. Classification and regression trees. – Routledge, 2017. – 368 с.
289. Specht D.F. A general regression neural network //IEEE transactions on neural networks. – 1991. – Т. 2. – №. 6. – с. 568-576.
290. Breiman L. Random forests //Machine learning. – 2001. – Т. 45. – с. 5-32.
291. Alkhamash E.H. An Optimized Gradient Boosting Model by Genetic Algorithm for Forecasting Crude Oil Production //Energies. – 2022. – Т. 15. – №. 17. – с. 6416.
292. Moorthy A.K., Bovik A.C. A two-step framework for constructing blind image quality indices // IEEE Signal processing letters. – 2010. – Т. 17. – №. 5. – с. 513-516.
293. Mittal A., Moorthy A.K., Bovik A.C. No-reference image quality assessment in the spatial domain // IEEE Transactions on Image Processing. – 2012. – Т. 21. – №. 12. – с. 4695-4708.
294. Mittal A., Soundararajan R., Bovik A.C. Making a completely blind image quality analyzer // IEEE Signal Process. Lett. – 2013. – Т. 20. – №. 3. – с. 209-212.
295. Zhang L., Zhang L., Bovik A.C. A feature-enriched completely blind image quality evaluator // IEEE Transactions on Image Processing. – 2015. – Т. 24. – №. 8. – с. 2579-2591.

296. Liu L., Hua Y., Zhao Q., Huang H., Bovik A.C. Blind image quality assessment by relative gradient statistics and adaboosting neural network // Signal Processing: Image Communication. – 2016. – T. 40. – с. 1-15.
297. Ma M., Guo J. K. Automatic image cropping for mobile device with built-in camera //First IEEE Consumer Communications and Networking Conference, 2004. CCNC 2004. – IEEE, 2004. – с. 710-711.
298. Yan J., Lin S., Bing Kang S., Tang X. Learning the change for automatic image cropping //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2013. – с. 971-978.
299. Jaiswal N., Meghrajani Y.K. Saliency based automatic image cropping using support vector machine classifier //2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS). – IEEE, 2015. – с. 1-5.
300. Xu Y., Xu W., Wang M., Li L., Sang G., Wei P., Zhu L. Saliency aware image cropping with latent region pair //Expert Systems with Applications. – 2021. – T. 171. – с. 114596.
301. Jia G., Huang H., Fu C., He R. Rethinking image cropping: Exploring diverse compositions from global views //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – с. 2446-2455.
302. Lin S.S., Yeh I.C., Lin C.H., Lee T.Y. Patch-based image warping for content-aware retargeting //IEEE transactions on multimedia. – 2012. – T. 15. – №. 2. – с. 359-368.
303. Avidan S., Shamir A. Seam carving for content-aware image resizing //Seminal Graphics Papers: Pushing the Boundaries, Volume 2. – 2023. – с. 609-617.
304. Кормен, Т., Лейзерсон, Ч., Ривест, Р., Штайн, К. Алгоритмы: построение и анализ, 2-е изд. – М.: Вильямс, 2005. – 1296 с
305. Kiess J., Kopf S., Guthier B., Effelsberg W. Seam carving with improved edge preservation //Multimedia on Mobile Devices 2010. – SPIE, 2010. – T. 7542. – с. 147-157.

306. Chen Y., Pan Y., Song M., Wang M. Improved seam carving combining with 3D saliency for image retargeting //Neurocomputing. – 2015. – Т. 151. – С. 645-653.
307. Lin W., Zhang F., Lian R., Xu L., Chen X., Kuang L. Seam Carving Algorithm Based on Saliency //Advances in Smart Vehicular Technology, Transportation, Communication and Applications: Proceedings of the First International Conference on Smart Vehicular Technology, Transportation, Communication and Applications, – Springer International Publishing, 2018. – с. 282-291.
308. Hasan M.K., Ahsan, M.S., Newaz S.S., Lee G.M. Human face detection techniques: A comprehensive review and future research directions //Electronics. – 2021. – Т. 10. – №. 19. – с. 2354.
309. Egorova M.A., Murynin A.B., Safonov I.V. An Improvement of face detection algorithm for color photos //Pattern Recognition and Image Analysis. – 2009. – Т. 19. – с. 634-640.
310. Gomez G., Morales E. Automatic feature construction and a simple rule induction algorithm for skin detection //Proc. of the ICML workshop on Machine Learning in Computer Vision. – 2002. – Т. 31. с. 31-38.
311. Буча В.В., Сафонов И.В., Рычагов М.Н. Способ и система адаптивного переформатирования цифровых изображений: пат. 2368006 Рос. Федерация. № 2008100088/09; заявл. 10.01.2008; опубл. 20.09.2009, Бюл. № 26.
312. Bucha V.V., Safonov I.V., Rychagov M.N. Method and system of adaptive reformatting of digital image: пат. 8472753 США. № 20090180713; заявл. 16.07.2009; опубл. 25.06.2013.
313. Bucha V., Safonov I., Rychagov M., Hong J.K., Kim S.H. Retargeting of digital photos and documents //Digital Photography V. – SPIE, 2009. – Т. 7250. – с. 308-317.
314. Safonov I., Tolstaya E. Matching aspect ratio of digital photo via complementing //17th International Conference on Computer Graphics and Vision GRAPHICON. – 2007. – с. 154-157.

315. Ли Енмин, Сафонов И.В., Курилин И.В. Способ и система для просмотра изображения на устройстве отображения: пат. 2509377 Рос. Федерация. № 2012122959/08; заявл. 05.06.2012; опубл. 10.03.2014, Бюл. № 7.
316. Lewis J.P. Fast template matching //Vision interface. – 1995. – Т. 95. – №. 120123. – с. 15-19.
317. Corrigan-Kavanagh E., Frohlich D.M., Scarles C. Re-invigorating the photo album: Augmenting printed photobooks with digital media //Personal and ubiquitous computing. – 2023. – Т. 27. – №. 2. – с. 467-480.
318. Obrador P., Moroney N. Automatic image selection by means of a hierarchical scalable collection representation //Visual Communications and Image Processing. – SPIE, 2009. – Т. 7257. – с. 285-296.
319. Graham A., Garcia-Molina H., Paepcke A., Winograd T. Time as essence for photo browsing through personal digital libraries //Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries. – 2002. – с. 326-335.
320. Platt J.C., Czerwinski M., Field B.A. PhotoTOC: Automatic clustering for browsing personal photographs //Fourth international conference on information, communications, and signal processing. – IEEE, 2003. – Т. 1. – с. 6-10.
321. Naaman M., Song, Y.J., Paepcke A., Garcia-Molina H. Automatic organization for digital photographs with geographic coordinates //Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries. – 2004. – с. 53-62.
322. Zhang X., Gao Y., Atkins C.B., Cheatle P., Xiao J., Chao H., Willis C. Autophotobook: using technology to streamline photobook creation //Imaging and Printing in a Web 2.0 World; and Multimedia Content Access: Algorithms and Systems IV. – SPIE, 2010. – Т. 7540. – с. 149-163.
323. Chu W.T., Chen J.C., Wu J.L. Tiling slideshow: an audiovisual presentation method for consumer photos //IEEE Multimedia. – 2007. – Т. 14. – №. 3. – с 36-45.
324. Rother C., Bordeaux L., Hamadi Y., Blake A. Autocollage //ACM transactions on graphics (TOG). – 2006. – Т. 25. – №. 3. – с. 847-852.

325. Yeh C.H., Barsky B.A., Ouhyoung M. Personalized photograph ranking and selection system considering positive and negative user feedback //ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM). – 2014. – Т. 10. – №. 4. – с. 1-20.
326. Talebi H., Milanfar P. NIMA: Neural image assessment //IEEE transactions on image processing. – 2018. – Т. 27. – №. 8. – с. 3998-4011.
327. Kuzovkin D., Pouli T., Cozot R., Le Meur O., Kervec J., Bouatouch K. Image selection in photo albums //Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval. – 2018. – с. 397-404.
328. Fageth R., Debus C., Sandhaus P. Memory preservation made prestigious but easy //Color Imaging XVI: Displaying, Processing, Hardcopy, and Applications. – SPIE, 2011. – Т. 7866. – С. 128-135.
329. Egorova M.A., Safonov I.V. Algorithms for photobook creation //Proc. of 10th International conference on Pattern recognition and Image Analysis: New information technologies. – 2010. – Т. 2, с.265-268.
330. Egorova M.A., Safonov I.V. Photobook generation algorithms //Pattern Recognition and Image Analysis. – 2011. – Т. 21. – с. 477-480.
331. Egorova M.A., Safonov I.V. Algorithms for photo book authoring //Pattern recognition and image analysis. – 2013. – Т. 23. – с. 57-67.
332. Potapova E., Egorova M., Safonov I. Automatic photo selection for media and entertainment applications //19th International Conference on Computer Graphics and Vision GRAPHICON-2009. – 2009. – с. 117-124.
333. Egorova M., Safonov I. Selection of photos for album building applications //2010 20th International Conference on Pattern Recognition. – IEEE, 2010. – с. 3256-3259.
334. Потапова Е.В., Сафонов И.В. Ранжирование фотографий с использованием карты важности //12-я Международная конференция и выставка Цифровая обработка сигналов и ее применение (DSPA), – 2010. – Т.2, с. 78-81.



335. Foi A., Katkovnik V., Egiazarian K. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images //IEEE transactions on image processing. – 2007. – Т. 16. – №. 5. – с. 1395-1411.
336. Pyatykh S., Hesser J., Zheng L. Image noise level estimation by principal component analysis //IEEE transactions on image processing. – 2012. – Т. 22. – №. 2. – с. 687-699.
337. Сафонов И., Аюпов А., Бёрнс С. Умножение матриц  $n$ -разрядных чисел с фиксированной точкой с помощью  $n/2$ -разрядных векторных инструкций //Проблемы разработки перспективных микро-и наноэлектронных систем (МЭС). – 2016. – №. 3. – с. 141-148.
338. Safonov I., Kornilov A., Makienko D. An Approach for Matrix Multiplication of 32-Bit Fixed Point Numbers by Means of 16-Bit SIMD Instructions on DSP //Electronics. – 2022. – Т. 12. – №. 1. – с. 78-94.
339. Карацуба А.А. Сложность вычислений //Труды математического института имени ВА Стеклова. – 1995. – Т. 211. – с. 186-202.
340. Уоррен Г.С. Алгоритмические трюки для программистов, 2-е изд. – М: Вильямс, 2013. – 512 с.
341. Strassen V. Gaussian elimination is not optimal //Numerische mathematik. – 1969. – Т. 13. – №. 4. – с. 354-356.
342. Duan R., Wu H., Zhou R. Faster matrix multiplication via asymmetric hashing //2023 IEEE 64th Annual Symposium on Foundations of Computer Science (FOCS). – IEEE, 2023. – с. 2129-2138.
343. Heckbert P. Color image quantization for frame buffer display //ACM Siggraph Computer Graphics. – 1982. – Т. 16. – №. 3. – с. 297-307.
344. Itti L., Koch C., Niebur E. A model of saliency-based visual attention for rapid scene analysis //IEEE Transactions on pattern analysis and machine intelligence. – 1998. – Т. 20. – №. 11. – с. 1254-1259.

345. Головистиков А.В. Задачи двумерной прямоугольной упаковки и раскроя: обзор //Информатика. – 2008. – №. 4 (20). – с. 18-33.
346. Данилкина М.Д., Куликов В.Р. О приближенном решении задачи об упаковке на основе эвристик для организации складского пространства //Системы управления, информационные технологии и математическое моделирование. – 2023. – с. 49-55.
347. Atkins C.B. Blocked recursive image composition //Proceedings of the 16th ACM international conference on Multimedia. – 2008. – с. 821-824.
348. Chao H., Tretter D.R., Zhang X., Atkins C.B. Blocked recursive image composition with exclusion zones //Proceedings of the 10th ACM symposium on Document engineering. – 2010. – с. 111-114.
349. Egorova M., Safonov I., Korobkov N. Collage for cover of Photobook //18th International Conference on Computer Graphics and Vision GRAPHICON-2008. – 2008. – с. 160-163.
350. Егорова М.А., Сафонов И.В. Размещение фотографий на холсте с определением коэффициентов масштабирования //31-я конференция молодых ученых и специалистов ИППИ РАН Информационные технологии и системы (ИТиС-2008). –2008. – с. 90-96.
351. Саймон Д. Алгоритмы эволюционной оптимизации /М.: ДМК Пресс, 2020. – 940 с.
352. Chu W.T., Chen J.C., Wu J.L. Tiling slideshow: an audiovisual presentation method for consumer photos //IEEE MultiMedia. – 2007. – Т. 14. – №. 3. – с. 36-45.
353. Wu Z., Aizawa K. Very fast generation of content-preserved photo collage under canvas size constraint //Multimedia Tools and Applications. – 2016. – Т. 75. – с. 1813-1841.
354. Yu J., Chen L., Zhang M., Li M. SoftCollage: A Differentiable Probabilistic Tree Generator for Image Collage //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. – 2022. – с. 3729-3738.

355. Wang J., Sun J., Quan L., Tang X., Shum H.Y. Picture collage //2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). – IEEE, 2006. – T. 1. – c. 347-354.
356. Zhang M., Li M., Yu J., Chen L. Aesthetic photo collage with deep reinforcement learning //IEEE Transactions on Multimedia. – 2023. – T. 25. – c. 4653-4664.
357. Lee M.H., Singhal N., Cho S., Park I.K. //2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops. – IEEE, 2010. – c. 24-30.
358. Tian A., Zhang X., Tretter D.R. Content-aware photo-on-photo composition for consumer photos //Proceedings of the 19th ACM international conference on Multimedia. – 2011. – c. 1549-1552.
359. Liu L., Zhang H., Jing G., Guo Y., Chen Z., Wang W. Correlation-preserving photo collage //IEEE transactions on visualization and computer graphics. – 2017. – T. 24. – №. 6. – c. 1956-1968.

## Приложение А. Акты о внедрении результатов работы



Общество с ограниченной ответственностью  
**“Исследовательский Центр Самсунг”**  
 127018, Российская Федерация, Москва,  
 ул. Двинцев, 12, корп.1, офис № 1500  
 Тел: +7 (495) 797-2500; Факс: +7 (495) 797-2501

### АКТ

о внедрении результатов диссертационной работы Сафонова Ильи Владимировича по теме «Методы обработки изображений для систем сканирования и печати», представленной на соискание ученой степени доктора технических наук по специальности 2.3.8. «Информатика и информационные процессы»

В период работы Сафонова И.В. в ООО “Исследовательский центр Самсунг” им был разработан ряд методов обработки изображений для систем сканирования и печати, что нашло свое отражение в следующих патентах РФ: 2298293, 2308166, 2298292, 2368091, 2298223, 2298226, 2310912, 2320011, 2324225, 2400815, 2338255, 2338252, 2383924, 2368006, 2365510, 2368007, 2374688, 2405279, 2411584, 2405201, 2435217, 2431192, 2446464, 2445677, 2458396, 2481635, 2469399, 2469400, 2509377, 2509366, 2520407, 2534005.

Разработка методов выполнена на основе предложенных:

- подхода к подготовке фотографий для печати, при котором учитываются размеры бумажного отпечатка, разрешение и технология печати;
- методики сканирования и размещения информации на странице, которая обеспечивает эффективное хранение, передачу, печать и включает методы сегментации на фон, текст и графику; устранение скоса; подавление муара; повышение качества изображения; конвертацию в метафайловый формат языка описания страниц;
- подхода для реализации специальных режимов печати за счет локальной модификации кода процессора растровых изображений;
- технических решений для печати элементов защиты от копирования и скрытой информации.

26.12.2022



Начальник управления  
**НАВАСАРДЯН С.В.**, по  
 Дов-сти № 1 от 2.03.20

Компания Шлюмберже  
Ленинградское шоссе, д. 16А стр.3  
Россия, Москва, 125171  
Тел.: +7 495 935 8200 / Факс: +7 495 935 8780

АКТ  
о внедрении результатов диссертационной работы  
Сафонова Ильи Владимировича  
по теме  
«Методы обработки изображений для систем сканирования и печати»,  
представленной на соискание ученой степени  
доктора технических наук

Настоящим актом подтверждается, что следующие результаты диссертационной работы Сафонова И.В. «Методы обработки изображений для систем сканирования и печати» были использованы в научно-исследовательском центре компании «Шлюмберже» при разработке новых технологий для нефтегазовой индустрии:

- спиральная методология научно-исследовательской работы на основе анализа рисков;
- улучшение видимости затемнённых и слабоконтрастных областей при анализе сейсмических данных;
- повышение контраста и сегментации рентгеновских изображений при измерении межфазного натяжения оптически непрозрачных флюидов;
- оценки резкости и контраста изображений рентгеновской компьютерной томографии и растрового электронного микроскопа с фокусированным ионным пучком.

Начальник исследовательского отдела  
НИЦ Шлюмберже

Хан В.К.



27.09.2023