

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ МОСКОВСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М.В. ЛОМОНОСОВА

На правах рукописи

Боков ✓

УДК 004.932.4

Боков Александр Александрович

**ИССЛЕДОВАНИЕ И РАЗРАБОТКА АЛГОРИТМОВ
ВОССТАНОВЛЕНИЯ ФОНА В ЗАДАЧЕ ПРЕОБРАЗОВАНИЯ
ВИДЕО В СТЕРЕОСКОПИЧЕСКИЙ ФОРМАТ**

Специальность 05.13.11 —

«Математическое и программное обеспечение вычислительных машин,
комплексов и компьютерных сетей»

Диссертация на соискание учёной степени
кандидата физико-математических наук

Научный руководитель:
кандидат физико-математических наук, старший научный
сотрудник ФГБОУ ВО «Московский государственный
университет имени М.В. Ломоносова»

Ватолин Дмитрий Сергеевич

Москва — 2019

Оглавление

	Стр.
Введение	4
Глава 1. Алгоритм восстановления фона в видео на основе непараметрической модели движения и покадрового уточнения	11
1.1 Постановка задачи	11
1.2 Обзор существующих алгоритмов	14
1.3 Описание предложенного алгоритма	19
1.3.1 Базовый алгоритм построения оптического потока	24
1.3.2 Непараметрическая модель движения фона	29
1.3.3 Базовый алгоритм покадрового восстановления фона	35
1.3.4 Алгоритм покадрового восстановления с уточнением	38
1.3.5 Компенсация изменений условий освещенности	48
1.4 Анализ вычислительной сложности алгоритма	51
1.5 Экспериментальная оценка	54
1.6 Экспертная оценка	57
1.7 Программная реализация	58
1.8 Выводы по главе 1	59
Глава 2. Методика сравнения алгоритмов восстановления фона в видео	60
2.1 Введение	60
2.2 Обзор существующих методов сравнения алгоритмов восстановления фона	61
2.3 Построение тестового набора данных	63
2.4 Функционалы качества восстановленного фона	65
2.5 Экспертная оценка	70

	Стр.
2.6	Выбор функционала качества 74
2.7	Выводы по главе 2 77
Глава 3. Алгоритм построения стереоскопических ракурсов с	
	помощью многослойного восстановления фона 79
3.1	Постановка задачи 79
3.2	Обзор существующих алгоритмов 81
3.3	Описание предложенного алгоритма 85
3.3.1	Модель движения фона с учетом межкадрового изменения диспаратности 88
3.3.2	Алгоритм покадрового восстановления фона с учетом изменения диспаратности 89
3.3.3	Алгоритм построения ракурсов на базе результата многослойного восстановления фона 91
3.4	Анализ результатов работы предложенного алгоритма 92
3.5	Выводы по главе 3 93
Заключение 96	
Список литературы 98	
Список рисунков 106	
Список таблиц 109	

Введение

Актуальность темы. Несмотря на то, что история стереокинематографа насчитывает уже более 90 лет, современный его расцвет начался с выхода кинофильма «Аватар» в 2009 году и обусловлен существенным техническим прогрессом как в области производства стереофильмов, так и в области технологий показа стереоскопического видео. Характерной чертой современного этапа развития данной области является преобладание конвертации в стереоскопический формат как основного способа производства стереофильмов. Например, в 2018 году среди 36 стереофильмов вышедших на экраны кинотеатров 27 были созданы именно путем конвертации в стереоскопический формат. В отличие от стереосъемки, предполагающей наличие специальной установки из камер, позволяющей снять сцену одновременно с двух различных ракурсов, в ходе конвертации новые ракурсы строятся искусственно из обычного моноскопического видео. Конвертация в стереоскопический формат обладает рядом преимуществ по сравнению со стереосъемкой, начиная с повышенного контроля над воспринимаемой зрителем степенью объемности каждого из объектов сцены по отдельности для достижения того или иного художественного эффекта и заканчивая возможностью конвертировать в стереоскопический формат старые популярные фильмы, которые изначально были сняты в моноскопическом формате.

Однако, на сегодняшний день конвертация фильмов в стереоскопический формат остается крайне трудоемким процессом, требующим большой объем ручной работы специально обученных художников. В связи с этим учеными активно исследуются алгоритмы и интерактивные программные системы, которые позволяют упростить процесс конвертации посредством уменьшения объема ручного труда, требуемого для достижения удовлетворительного результата. Задача преобразования видео в стереоскопический формат включает в себя следующие основные подзадачи:

- Построение карты глубины для всех кадров входного видео, которая описывает расстояние от каждой точки сцены до плоскости линзы камеры;
- Заполнение областей открытия, которые содержат фрагменты фона, скрытые за объектом переднего плана в исходном кадре видео, но становящиеся видимыми при изменении ракурса просмотра;
- Обработка полупрозрачных областей и границ объектов сцены.

Данная диссертационная работа посвящена решению подзадачи заполнения областей открытия путем восстановления изображения фона, расположенного за объектами переднего плана. Помимо преобразования видео в стереоскопический формат задача восстановления фона в видео также имеет ряд других приложений, таких как бесследное удаление нежелательных объектов и устранение дефектов пленки в ходе реставрации архивных видеозаписей. Однако стоит отметить, что в данной области до сих пор отсутствует общепринятая методика оценки качества в связи с тем, что традиционные методы оценки качества видео плохо отражают особенности визуального восприятия восстановленного фона. Также, современные алгоритмы восстановления фона в видео формулируют задачу в виде единой задачи оптимизации сразу по всем кадрам, что приводит к чрезмерно высокой вычислительной сложности, затрудняющей применение данных алгоритмов на практике. Таким образом, актуальной является задача выработки специализированной методики оценки качества восстановленного фона, а также задача разработки нового алгоритма восстановления фона с более низкой вычислительной сложностью в сравнении с аналогами.

Степень разработанности темы. Задача восстановления фона активно исследуется учеными с начала 2000-х годов. Наиболее ранние подходы к ее решению на основе моделирования средствами дифференциальных уравнений исследуются в работах Марсело Берталмио (M. Bertalmio). Методы на основе копирования заплаток, позволяющие восстанавливать более крупные области, были развиты в работах Антонио Криминиси (A. Criminisi). Современные методы базируются на явном моделировании движения фона и формулировке единой задачи оптимизации на все кадры видео, что позволяет достичь высокого качества

результата. Данный подход был существенно развит в работе Джиа-Бин Хуанга (J.-В. Huang).

Целью данной работы является исследование и разработка нового решения проблемы заполнения областей открытия на базе восстановления фона за объектами переднего плана, которое бы позволило увеличить степень автоматизации процесса преобразования видео в стереоскопический формат.

Для достижения поставленной цели требуется решить следующие **задачи**:

1. Выработать методику оценки качества восстановленного фона и систематизировать известные алгоритмы его построения.
2. Разработать новый алгоритм восстановления фона в видео, обладающий более низкой вычислительной сложностью по сравнению с аналогами.
3. Создать алгоритм построения новых ракурсов на основе разработанного алгоритма восстановления фона в видео.

Основные положения, выносимые на защиту:

1. Предложен новый подход к решению проблемы заполнения областей открытия на основе проведения многослойного восстановления фона в видео с использованием непараметрической модели движения для каждого слоя.
2. Разработан новый алгоритм восстановления фона в видео с пространственной сложностью, не зависящей от количества кадров во входной видеопоследовательности. По результатам экспериментальной оценки предложенный алгоритм превзошел существующие аналоги по точности найденного решения и не уступил аналогам по результатам экспертной оценки.
3. Предложено два новых функционала качества работы алгоритмов восстановления фона в видео, основанных на критерии локального сходства с эталоном, которые обладают более высокой корреляцией с экспертной оценкой по сравнению с ранее использовавшимися для этой цели функционалами.

4. Все предложенные алгоритмы и функционалы качества были реализованы, а также проведены вычислительные эксперименты, подтверждающие их эффективность. Алгоритм восстановления фона был протестирован на реальных данных, предоставленных коммерческой компанией.

Научная новизна:

1. Впервые предложен алгоритм многослойного восстановления фона в видео, который позволяет моделировать изображение фона в областях перекрытия трех и более объектов.
2. Предложен новый алгоритм восстановления фона в видео, основанный на покадровом уточнении результата восстановления, который обладает более низкой вычислительной сложностью в сравнении с известными аналогами.
3. Предложено два новых функционала качества восстановленного фона, измеряющих локальное сходство с эталоном и постоянство локального сходства с эталоном с течением времени, которые лучше отражают визуальное качество фона, чем ранее известные функционалы.

Теоретическая значимость. В работе задача восстановления фона в видео сформулирована в виде задачи покадровой оптимизации. Показано, что она сводится к решению набора СЛАУ с разреженной симметричной положительно определенной матрицей.

Практическая значимость. Предложенный алгоритм восстановления фона в видео был реализован в виде программного инструмента, позволяющего автоматически восстанавливать фон в произвольных областях видеопоследовательности с более высоким качеством по сравнению с доступными на момент проведения исследования коммерческими аналогами. Данный инструмент может быть применен в киностудиях для удешевления процесса конвертации моноскопического видео в стереоскопический формат за счет автоматизации восстановления фона за объектами переднего плана. Разработанный программный инструмент был апробирован на данных из реальных кинофильмов, которые были предоставлены коммерческой компанией YUVsoft. Предложенные функционалы качества

были включены в разработанную методику оценки качества восстановленного фона. Данная методика вместе с подготовленной базой видеопоследовательностей с эталонными изображениями фона легли в основу открытого для новых участников сравнения алгоритмов восстановления фона в видео, доступного в сети Интернет по адресу <http://videocompletion.org>.

Методология и методы исследования. В диссертационной работе применялись методы линейной алгебры, теории алгоритмов, а также методы численного решения разреженных систем линейных алгебраических уравнений.

Достоверность полученных результатов подтверждается публикациями в рецензируемых научных изданиях, а также апробацией в рамках российских и международных конференций.

Апробация работы. Основные результаты работы докладывались на:

- 8-ой международной конференции по 3D визуализации (IC3D 2018), 5 декабря 2018 г., Брюссель, Бельгия;
- 25-ой международной конференции по обработке изображений (ICIP 2018), 7-10 октября 2018 г., Афины, Греция;
- Международной конференции по мультимедийным системам (ICMEW 2017), 10 июля 2017 г., Гонконг, Китай;
- 5-ой международной научно-практической конференции «Инновационные технологии в кинематографе и образовании», 12-13 ноября 2018 г., Москва, Россия;
- Всероссийской научной конференции «Ломоносовские чтения-2018», 16-27 апреля 2018 г., Москва, Россия;
- Всероссийской научной конференции «Ломоносовские чтения-2017», 17-26 апреля 2017 г., Москва, Россия.

Личный вклад автора. В работах [1—6] научному руководителю Д.С. Ватолину принадлежит постановка задачи, А.А. Бокову – все полученные результаты. В статье [3] М.В. Ерофееву и Ю.А. Гитману принадлежит основа разработанной Интернет-платформы <http://videocompletion.org> для открытого сравнения алгоритмов восстановления фона (стр. 607).

Публикации. Основные результаты по теме диссертации изложены в 6 печатных изданиях, 2 из которых изданы в журналах, рекомендованных ВАК [1; 2], 4 входят в международную базу цитирования Scopus [3—6].

Объем и структура работы. Диссертация состоит из введения, трех глав и заключения. Полный объем диссертации составляет 110 страниц, включая 14 рисунков и 7 таблиц. Список литературы содержит 73 наименования.

Как было упомянуто выше, данная работа посвящена исследованию и разработке алгоритмов восстановления фона в видео и заполнения областей открытия, возникающих в ходе преобразования видео в стереоскопический формат, с помощью восстановленного фона. Для решения задачи заполнения областей открытия было предложено множество различных подходов, обзор которых приведен в разделе 3.2, в том числе техники неявного заполнения, которые устраняют области открытия за счет деформации объекта переднего плана и фона. Ключевым достоинством подхода, основанного на применении алгоритмов восстановления фона в видео, является возможность восстановления истинного изображения фона для всех фрагментов, которые были видимы хотя бы в одном кадре входной видеопоследовательности, и использования этой информации для заполнения областей открытия.

Решению задачи восстановления фона в видео посвящено множество научных работ, обзор которых представлен в разделе 1.2. В разделе 1.3 приводится описание нового алгоритма, который, как показано в разделах 1.5 и 1.6, не уступает известным аналогам и, как показано в разделе 1.4, обладает более низкой пространственной сложностью. Также стоит отметить высокую эффективность созданной программной реализации, которая позволяет увеличить скорость обработки видео более, чем в 100 раз по сравнению с реализациями других алгоритмов, обладающих схожими характеристиками. Предложенный алгоритм основан на непараметрической модели движения фона, которая была построена на базе алгоритма вычисления оптического потока DeepFlow [7], и производит восстановление фона покадрово, что позволило достичь низкой вычислительной сложности алгоритма и высокой скорости работы его реализации.

Авторы многих предложенных в литературе алгоритмов восстановления фона в видео [8—11] не проводят объективную оценку качества и количественное сравнение с аналогами. Это в первую очередь связано со сложностью задачи оценки качества восстановленного фона, так как даже при наличии эталонных изображений фона использование традиционных методов оценки качества видео приводит к низкой корреляции с экспертной оценкой. В разделе 2.2 приводится обзор различных подходов к оценке качества работы алгоритмов восстановления фона в видео, а в разделах 2.3 и 2.4 представлена новая методика сравнения алгоритмов, которая включает два новых функционала качества, которые превзошли ранее известные по корреляции с экспертной оценкой.

В разделе 3.3 приводится описание предложенного алгоритма построения стереоскопических ракурсов на основе описанного в первой главе алгоритма восстановления фона в видео. Ключевой отличительной особенностью данного алгоритма является проведение многослойного восстановления фона, что позволяет проводить корректное заполнение областей открытия в областях перекрытия трех и более объектов. В разделе 3.4 на базе проведенной экспертной оценки показано превосходство предложенного алгоритма перед аналогами, которые также основаны на использовании восстановленного фона.

В Заключении сформулированы основные результаты диссертационного исследования, рассмотрены возможные варианты их применения и обозначены перспективы дальнейших исследований.

Глава 1. Алгоритм восстановления фона в видео на основе непараметрической модели движения и покадрового уточнения

1.1 Постановка задачи

Задача восстановления фона в видео является подзадачей рассматриваемой в данной работе задачи преобразования видео в стереоскопический формат. Она состоит в восстановлении изображения фона в рамках поданной на вход алгоритму области неизвестного фона, определенной в каждом кадре входной видеопоследовательности (пример восстановленного фона приведен на рисунке 1.1). В зависимости от приложения, в качестве такой области неизвестного фона могут выступать:

- Область, занимаемая определенным объектом переднего плана в видеопоследовательности. Тогда задача состоит в восстановлении фона, расположенного за данным объектом, который может использоваться для заполнения областей открытия в ходе преобразования в стереоскопический формат или для бесследного устранения этого объекта из видеопоследовательности;
- Области, занимаемые логотипами или водяными знаками, которые требуется удалить из входного видео;
- Области видимых дефектов пленки в архивных видеозаписях или искажений, полученных в результате потери пакетов при передаче видео по сети, которые требуется устранить.

Также стоит отметить, что во многих приложениях не требуется восстановление истинного изображения фона, а достаточно лишь получить правдоподобный результат, который имеет высокую экспертную оценку (то есть, зрителю сложно определить какое видео содержит результат работы алгоритма, а какое – настоящее изображение фона). Поэтому задача может иметь множество решений с оди-



(а) Исходный кадр (б) Область неизвестного фона (в) Восстановленный фон

Рисунок 1.1 — Пример кадра входной видеопоследовательности (а) с определенной областью неизвестного фона (б) и результатом работы алгоритма восстановления фона в рамках данной области.

наково высоким уровнем качества и при их оценке нужно отталкиваться от результатов экспертной оценки.

Формально задачу восстановления фона можно поставить следующим образом. Обозначим область определения видеопоследовательности как $I = [1, W] \times [1, H] \times \{1, \dots, T\}$, где W , H – ширина и высота кадра в пикселах, соответственно, T – количество кадров. Тогда входными данными алгоритма являются значения цвета вне требующей восстановления области $\Omega \subset I$. То есть, по значениям цвета $C : I \setminus \Omega \mapsto \mathbb{R}^3$ вне области неизвестного фона требуется определить цвет внутри рассматриваемой области $C(\Omega)$. Если Ω – это область, занимаемая объектом переднего плана, то $C(\Omega)$ описывает фон, находящийся за таким объектом. Далее в описании предложенного алгоритма будем предполагать, что областью значений C является лишь \mathbb{R} , то есть значения яркости пикселей в видео. Построенный алгоритм тривиально обобщается на случай использования полного цветового пространства RGB, как описано в разделе 1.3.5. Очевидно, для корректной постановки задачи требуется ввести ряд дополнительных предположений о природе входных данных. Стоит отметить, что данные предположения существенно различаются между различными алгоритмами, ранее предложенными в научных работах.

Ключевой особенностью задачи восстановления фона в видео, которая отличает ее от аналогичной задачи для изображений, является возможность найти

фрагменты фона, скрытые в текущем кадре, в других кадрах входного видео. Эта особенность существенно упрощает задачу, особенно если для любой скрытой точки фона $p \in \Omega$ можно найти соответствующую точку $q \notin \Omega$, где фон является видимым, однако часто это оказывается слишком сильным предположением. В то же время, работа с видео порождает дополнительные требования к алгоритмам восстановления фона:

- Результат восстановления должен быть устойчив во времени, ввиду высокой чувствительности зрительной системы человека к неестественным различиям между соседними кадрами. Применение алгоритмов восстановления фона в изображениях независимо к каждому кадру будет приводить к неприемлемым результатам именно по этой причине;
- Алгоритм должен иметь адекватную объемам видеоданных временную и пространственную сложность. Как показано в разделе 1.2, многие существующие алгоритмы трудно на практике применять к видео большой продолжительности и высокого разрешения из-за чрезмерно высокой вычислительной сложности.

Для решения поставленной задачи предлагается новый алгоритм восстановления фона в видео на основе непараметрической модели движения фона, которая была построена на базе алгоритма вычисления оптического потока DeepFlow [7], и позволяет увеличить область применимости алгоритма по сравнению с:

- Алгоритмами вычитания фона [12], которые опираются на предположение о статичности камеры и, как следствие, восстанавливаемого фона. Такие алгоритмы применимы, например, к видео, полученному в рамках системы видеонаблюдения, однако в видеопоследовательностях, характерных для киноиндустрии, камера практически всегда является движущейся;
- Алгоритмами, использующими простые параметрические модели движения фона, такие как аффинное [10] или проективное [13] преобразование. Такие алгоритмы применимы к видеопоследовательностям, где весь фон целиком можно достаточно хорошо аппроксимировать одной плоско-

стью, однако будут приводить к существенным погрешностям в случаях, когда это не выполнено.

Также многие алгоритмы [9; 14] опираются на предположение о том, что каждая точка фона в Ω должна быть видима хотя бы в одном кадре входного видео, в то время как предложенный алгоритм позволяет проводить восстановление и в областях фона, которые остаются скрытыми на протяжении всей входной видеопоследовательности. Наиболее современные алгоритмы [11] лишены всех вышеперечисленных недостатков, однако они основаны на оптимизации единого функционала, определенного сразу для всей видеопоследовательности (то есть, количество неизвестных в задаче оптимизации пропорционально количеству пикселей во всех кадрах области Ω). Такой подход приводит к высокой вычислительной сложности алгоритма восстановления фона, в том числе высокой пространственной сложности, что затрудняет обработку продолжительных видео на вычислительных устройствах с ограниченным объемом ОЗУ. Отличительной особенностью предложенного алгоритма является восстановление и уточнение искомого изображения фона в покадровом режиме, что позволяет снизить пространственную сложность (см. раздел 1.4) и построить практическую реализацию со сравнительно высокой скоростью обработки видео (см. раздел 1.7). В то же время, как показано в разделах 1.5 и 1.6, предложенный алгоритм не уступает известным по качеству восстановленного фона.

1.2 Обзор существующих алгоритмов

Одним из классических подходов к восстановлению фона в видео является использование трехмерных пространственно-временных блоков пикселей, вложенных во всю область определения видео I , для оценки и последующей максимизации локального сходства восстановленного фона в Ω с известным фоном в окружающей области $I \setminus \Omega$. В работе 2004 года [8] задача формулируется как еди-

ная задача оптимизации по всей области определения видео, в ходе которой непосредственно минимизируется сумма расстояний от каждого трехмерного блока области неизвестного фона до наиболее схожего трехмерного блока в области известного фона во входном видео. Для нахождения наиболее схожего трехмерного блока в видео используется полный перебор и его простые оптимизации, что ограничивает область применимости метода на практике до коротких последовательностей низкого разрешения (авторы приводят результаты на видео разрешения 360×288). В работе 2014 года [10] был предложен способ значительно ускорить этап поиска наиболее схожего трехмерного блока во входном видео путем адаптации алгоритма PatchMatch [15] для этой цели. Также авторами была модифицирована метрика расстояния между трехмерными блоками путем включения текстурных признаков, что позволило улучшить качество восстановления динамических текстур в видео, таких как волны, огонь или толпа людей. Однако, как было продемонстрировано в работе [11], подходы, основанные на использовании трехмерных блоков, обладают существенными ограничениями при обработке видеопоследовательностей с произвольным движением камеры. В работе [16] эта проблема была частично решена путем использования наклонных параллелепипедов, в которых угол наклона определяется межкадровым движением фона в соответствующей точке. Однако, в связи с формулировкой единой задачи оптимизации по всему видео, полученный алгоритм обладает высокой вычислительной сложностью. Авторы [16] отмечают, что их реализация требует около 50 минут на обработку видеопоследовательности из 90 кадров в разрешении 854×480 .

Некоторые алгоритмы накладывают явные ограничения на допустимый характер движения камеры и/или объектов во входном видео. В работе 2007 года [17] авторы допускают лишь плоскопараллельное движение камеры во входной сцене. Авторы [18; 19] исходят из предположения о том, что входное видео содержит движущиеся объекты переднего плана и статичный фон. В [13] для компенсации движения камеры используется глобальное проективное преобразование, которое неспособно моделировать движение более, чем одной плоскости в сцене при произвольном движении камеры. В работе 2012 года [9] модель движения фо-

на обобщается до кусочно-проективного преобразования (движение камеры компенсируется путем нахождения нескольких плоскостей в сцене и соответствующих им проективных преобразований). Авторы [9] также предлагают обобщение стандартного метода бесшовного монтажа Poisson Blending [20] в рамках задачи восстановления фона в видео для компенсации возможных изменений освещенности. Однако данный алгоритм основан на попарном выравнивании всех входных кадров, что приводит к квадратичной зависимости времени работы от количества кадров во входном видео. Авторы отмечают, что на обработку короткого видео в разрешении 960×720 у созданной ими реализации уходит до 4 часов при параллельной работе на сервере с 64 процессорами. В работе 2015 года [21] используется схожий подход, но радиус поиска ограничивается соседними кадрами для снижения вычислительной сложности (авторы предлагают использовать скользящее окно, состоящее из 21 кадра). Несмотря на это, предложенный алгоритм сохраняет сравнительно низкую скорость работы, которая затрудняет его применение на практике (авторы отмечают, что скорость работы их реализации в среднем равна 0,7 кадрам в минуту при использовании разрешения 1440×1056).

Наиболее схожим с предлагаемым в данной работе алгоритмом является класс алгоритмов восстановления фона, основанных на непараметрических методах восстановления поля векторов движения фона в области Ω , также известного как оптический поток. В некоторых работах [22—24] восстановление поля векторов движения в области неизвестного фона осуществляется путем копирования векторов из области известного фона с использованием трехмерных блоков для оценки локальной сходства поля векторов. Основным ограничением такого подхода является невозможность корректного восстановления поля векторов движения в неизвестной области простым копированием для некоторых типов движения камеры (например, наезд камеры). Авторы работы [25] используют схожую технику с использованием трехмерных блоков, однако дополнительно используют этап итеративного уточнения поля векторов движения в неизвестной области для повышения качества. В работе 2014 года [26] авторы сначала строят поле векторов движения в известной области и затем проводят его покадровую интерполяцию в

неизвестную область с использованием стандартного метода, основанного на модели процесса диффузии. Далее для восстановления фона авторы модифицируют метрику расстояния между двумерными блоками с учетом восстановленного поля векторов движения для поддержания временной стабильности. Необходимость поиска наиболее схожих двумерных блоков по всему входному видео снова приводит к высокой вычислительной сложности (авторы заявляют время работы их реализации в 3,3 часа на обработку видео в разрешении 1280×720 продолжительностью 82 кадра). В другой работе [14] формулируется задача глобальной оптимизации для совместного нахождения восстановленного фона и соответствующего ему поля векторов движения. Авторы описывают итеративный алгоритм решения данной задачи, основанный на чередовании вычисления поля векторов движения во всей видеопоследовательности на базе текущего восстановленного фона и вычисления восстановленного фона на базе текущего поля векторов движения в области Ω . Данный подход был далее развит в работе 2016 года [11], где предложенный авторами алгоритм демонстрирует высокое качество восстановления фона на широком классе видеопоследовательностей с произвольным движением камеры и наличием фрагментов фона, не видимых ни в одном из кадров входного видео. Однако снова вследствие формулировки задачи как задачи глобальной оптимизации по всей области определения видео, данный алгоритм сохраняет проблему большинства известных алгоритмов восстановления фона, которая заключается в чрезмерно высокой вычислительной сложности (созданной авторами реализации требуется порядка 3 часов на обработку видео 854×480 продолжительностью 90 кадров). Стоит отдельно отметить, что все алгоритмы восстановления фона, основанные на решении задачи глобальной оптимизации по всей области определения видео, обладают высокой пространственной сложностью. Количество неизвестных в таких задачах оптимизации оказывается пропорциональным количеству пикселей области Ω во всех кадрах, что приводит к оценке пространственной сложности не ниже, чем $O(WHT)$, где W , H , T – ширина, высота кадра и количество кадров во входном видео, соответственно.

Среди алгоритмов восстановления фона низкой вычислительной сложности стоит особо отметить работу [27], где предлагается специализированный алгоритм для восстановления фона в областях открытия, возникающих при построении стереоскопического видео, основанный на локальной экстраполяции поля векторов движения фона. Однако данный алгоритм обладает рядом ограничений. Во-первых, использование локального копирования векторов движения в область неизвестного фона позволяет корректно восстанавливать движение фона лишь при плоскопараллельном движении камеры и неспособно корректно обрабатывать сцены с существенным наездом или поворотом камеры. Также предложенный авторами алгоритм предполагает многократное копирование восстановленного фона вдоль восстановленных векторов движения с использованием бикубической интерполяции, что приводит к накоплению ошибки и размытию восстановленного фона, если он находится в рамках области Ω в течение длительного времени. Авторы работы 2014 года [28] предлагают алгоритм восстановления фона, способный работать в режиме реального времени, однако данный алгоритм состоит в использовании глобального проективного преобразования для покадрового распространения результатов пространственного восстановления фона. То есть, предложенный авторами подход никак не учитывает то, что восстанавливаемый фон может быть виден в других кадрах входной видеопоследовательности.

Таким образом, существующие алгоритмы восстановления фона либо накладывают сильные ограничения на допустимые входные видеопоследовательности, либо обладают чрезмерно высокой вычислительной сложностью, что затрудняет использование соответствующих алгоритмов на практике. В то же время предлагаемый в данной работе алгоритм совмещает общую непараметрическую модель движения, позволяющую обрабатывать видео с произвольным движением камеры, и сравнительно низкую вычислительную сложность, связанную с отказом от глобальной оптимизации по всей области определения видео, которая является доминирующим подходом среди современных аналогов, в пользу покадровой.

1.3 Описание предложенного алгоритма

Предложенный алгоритм для решения задачи восстановления фона строит отображение V , которое ставит в соответствие каждой точке области неизвестного фона $p \in \Omega$ другую точку q с известным фоном. Такое отображение можно интерпретировать как отображение из области неизвестного фона, где тот или иной фрагмент фона является скрытым в текущем кадре, в области других кадров входной видеопоследовательности, где соответствующий фрагмент фона является видимым. Если выполнено предположение о постоянстве яркости (цвета) фона, которое утверждает, что яркость (цвет) каждой точки фона остается неизменной от кадра к кадру на протяжении всей входной видеопоследовательности, то для проведения восстановления яркости (цвета) фона в $p \in \Omega$ достаточно просто провести копирование соответствующего значения из $V(p)$. Однако вследствие изменений условий освещенности в рамках сцены данное предположение может нарушаться, то есть как цвет, так и яркость фона могут различаться между разными кадрами входного видео. Для компенсации таких изменений освещенности предлагается модификация алгоритма бесшовного монтажа Poisson Blending [20], описанная в разделе 1.3.5, которая позволяет скорректировать значения яркости и цвета перед проведением копирования фона из других кадров в текущий согласно отображению V и, как следствие, предотвратить появление видимых швов в результате восстановления фона. Иллюстрация такого отображения V и основных принципов его построения приведена на рисунке 1.2.

Пусть искомое отображение V в каждой точке можно разложить на композицию отображений между соседними кадрами видео, которые описывают межкадровое движение каждой точки фона. В отличие от традиционного оптического потока [7], описывающего межкадровое движение изображения каждой точки в видеопоследовательности, искомые межкадровые отображения моделируют движение фона как в области $I \setminus \Omega$, так и внутри области неизвестного фона Ω . В контексте задачи преобразования видео в стереоскопический формат искомые

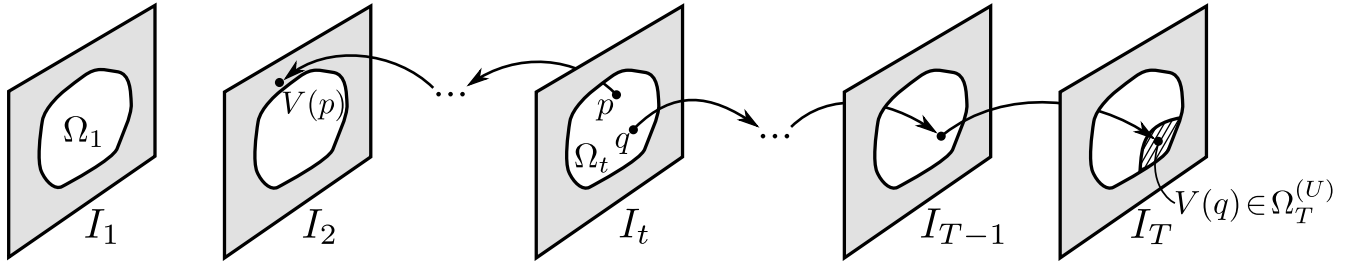
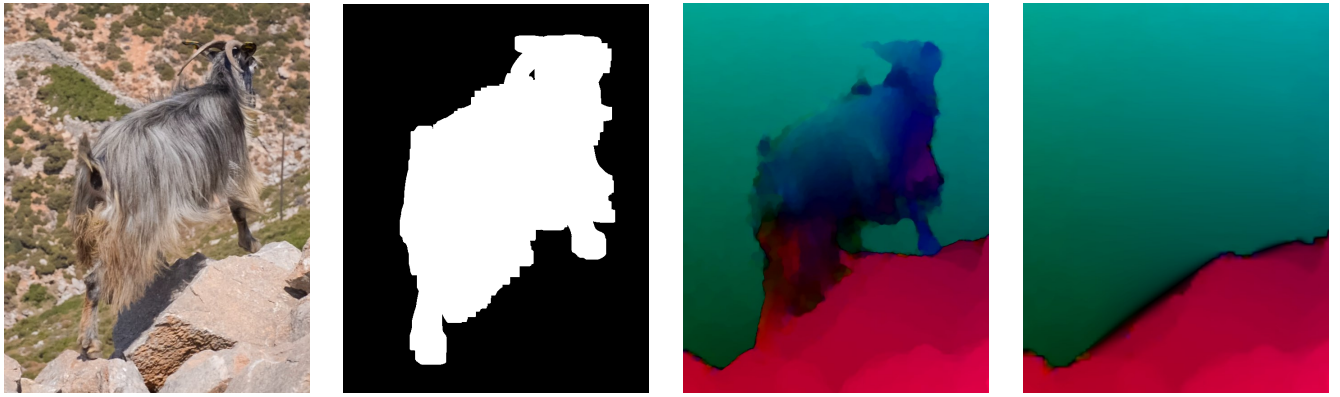


Рисунок 1.2 — Иллюстрация используемого подхода к восстановлению фона. Точка $p \in \Omega^{(K)}$ может быть отображена в область известного фона композицией межкадровых отображений $O^{t \rightarrow t-1}$, что позволяет восстановить фон простым копированием. Точка $q \in \Omega^{(U)}$ не имеет прямого отображения в область известного фона, но цвет в ней может быть восстановлен путем копирования из области $\Omega_T^{(U)} \subset \Omega^{(U)}$ на последнем кадре, к которой был применен алгоритм восстановления фона в изображениях.

межкадровые отображения можно интерпретировать как модель того, как двигается фон, расположенный за объектом переднего плана, занимающим область Ω во входном видео. Разница между оптическим потоком и искомым отображением, моделирующим движение фона, проиллюстрирована на рисунке 1.3. Формально искомое межкадровое отображение можно определить как отображение из одного кадра видеопоследовательности $I_t \subset I$ в соседний: $O^{t \rightarrow t \pm 1} : I_t \mapsto \mathbb{R}^2 \times \{t \pm 1\}$. Областью значений такого отображения является $\mathbb{R}^2 \times \{t \pm 1\}$, так как часть точек фона текущего кадра может выходить за пределы $[1, W] \times [1, H]$ в соседнем кадре. Такие отображения $O^{t \rightarrow t+1}, O^{t \rightarrow t-1}$, моделирующие движение фона для каждой пары соседних кадров с номерами $(t, t+1)$ и $(t, t-1)$, соответственно, в рамках предлагаемого алгоритма вычисляются согласно модели, описанной в разделе 1.3.2.

Для того, чтобы выразить V через $O^{t \rightarrow t \pm 1}$, для каждой точки $p \in \Omega$ определим множества точек $R^+[p], R^-[p] \subset I$, достижимых из нее посредством применения композиции межкадровых отображений вида $O^{t \rightarrow t+1}$ и $O^{t \rightarrow t-1}$, соответственно, оставаясь в пределах кадра $[1, W] \times [1, H]$. Затем представим множество Ω как объединение двух непересекающихся множеств $\Omega^{(K)}$ и $\Omega^{(U)}$, где $\Omega^{(U)} = \{p | \forall q \in R^+[p] \cup R^-[p] : q \in \Omega\}$, то есть множество точек, которые остаются скрытыми на протяжении всей видеопоследовательности, а $\Omega^{(K)} = \Omega \setminus \Omega^{(U)}$. В точках $p \in \Omega^{(K)}$



(а) Исходный кадр

(б) Область
неизвестного
фона(в) Оптический
поток(г) Модель
движения фона

Рисунок 1.3 — Иллюстрация разницы между традиционным оптическим потоком (в) и искомым отображением $O^{t \rightarrow t \pm 1}$ (г), моделирующим движение фона как внутри области неизвестного фона, так и вне ее. В визуализации цвет определяется направлением вектора межкадрового смещения соответствующей точки, а яркость определяется длиной соответствующего вектора.

фон может быть однозначным образом восстановлен как:

$$C(p) = C(V(p)), \text{ где } V(p) = \begin{cases} \arg \min_{q \in R^-[p], q \notin \Omega} \|p - q\|_t, & \exists q \in R^-[p] : q \notin \Omega \\ \arg \min_{q \in R^+[p], q \notin \Omega} \|p - q\|_t, & \text{иначе} \end{cases} \quad (1.1)$$

Здесь, $\|p - q\|_t$ обозначает модуль разницы в номере кадра между p и q . То есть, в точку неизвестной области p копируется значение яркости $C(q)$ из ближайшей (по номеру кадра) точки с известным фоном q из множества достижимых точек в предыдущих кадрах $R^-[p]$, если она существует. Иначе значение яркости копируется из аналогичной точки в будущем кадре.

Для области $\Omega^{(U)}$, остающейся скрытой на протяжении всей входной видеопоследовательности, предположим, что $\forall p \in \Omega^{(U)} : \exists (x_1, y_1, 1) \in R^-[p] \vee \exists (x_T, y_T, T) \in R^+[p]$. Тогда достаточно полностью знать фон в первом и последнем кадрах, чтобы восстановить фон везде в $\Omega^{(U)}$ посредством копирования. Для решения подзадачи восстановления фона в $\Omega^{(U)}$ на первом и последнем кадрах может быть использован любой известный алгоритм пространственного восстановления фона в отдельных изображениях. В данной работе используется алгоритм

[29], основанный на копировании фрагментов фона из того же самого изображения таким образом, чтобы минимизировать заметность возникающих в результате восстановления швов. Таким образом, в точках $p \in \Omega^{(U)}$ фон может быть восстановлен как:

$$C(p) = C(V(p)), \text{ где } V(p) = \begin{cases} (x_T, y_T, T), & \exists (x_T, y_T, T) \in R^+[p] \\ (x_1, y_1, 1), & \text{иначе} \end{cases} \quad (1.2)$$

В работе [11] используется жадный алгоритм выбора опорных (ключевых) кадров для проведения пространственного восстановления, не требующий никаких дополнительных предположений. Однако такой подход приводит к необходимости одновременно хранить в памяти межкадровые отображения $O^{t \rightarrow t \pm 1}$ для всех номеров кадров t , что существенно увеличивает пространственную сложность алгоритма восстановления фона. В то же время, используемое в данной работе предположение часто выполняется на практике и позволяет построить алгоритм покadroвого восстановления фона с пространственной сложностью, не зависящей от количества кадров во входной видеопоследовательности. Если это предположение не выполнено, то в результате работы предлагаемого алгоритма часть области неизвестного фона Ω останется невосстановленной. Если невосстановленная область достаточно мала, то удовлетворительный результат может быть получен применением простого алгоритма пространственного восстановления фона [30] независимо к каждому кадру, содержащему невосстановленную область. Именно такой подход был использован в созданной реализации предлагаемого алгоритма.

Предлагаемый алгоритм состоит из нескольких проходов по входной последовательности кадров в прямом и обратном порядках. В ходе первого прохода в прямом порядке инкрементально строится часть отображения V из уравнения (1.1), соответствующая отображению из области неизвестного фона в текущем кадре в область известного фона $I \setminus \Omega$ в предыдущих кадрах, и одновременно проводится восстановление фона согласно построенному отображению. Затем, на последнем кадре к оставшейся невосстановленной области применяется алгоритм пространственного восстановления [29]. В последующем проходе в об-

ратном порядке схожим образом строится часть отображения V , определенного в уравнениях (1.1) и (1.2), соответствующая отображению из области неизвестного фона в текущем кадре в область известного фона в будущих кадрах и область последнего кадра, к которой был применен алгоритм пространственного восстановления [29]. В конце обратного прохода к оставшейся невосстановленной области первого кадра снова применяется алгоритм [29]. В последнем прямом проходе проводится восстановление фона согласно части V из уравнения (1.2), определяющей отображение в пространственно восстановленную область первого кадра. Итого предложенный алгоритм состоит из трех проходов по входной последовательности кадров. Во всех проходах для каждого кадра с номером t выполняются следующие шаги:

1. Вычислить отображение в предыдущий кадр $O^{t \rightarrow t+1}$ или $O^{t \rightarrow t-1}$ согласно модели движения фона, описанной в разделе 1.3.2.
2. Вычислить отображение V_t , являющееся частью V , определяющей отображение из области неизвестного фона в текущем кадре t в соответствующие области с известным фоном в других кадрах (каких именно зависит от номера прохода, как было описано выше). V_t вычисляется на основе V_{t-1} (V_{t+1}) и $O^{t \rightarrow t-1}$ ($O^{t \rightarrow t+1}$). Описание базового алгоритма вычисления V_t приведено в разделе 1.3.3. Однако использование базового алгоритма приводит к накоплению ошибки модели движения фона по мере увеличения (уменьшения) номера кадра t . Предложенный алгоритм покадрового восстановления с уточнением, описанный в разделе 1.3.4, позволяет компенсировать накопление ошибок $O^{t \rightarrow t-1}$ ($O^{t \rightarrow t+1}$), если они достаточно малы.
3. Скопировать фон из других кадров в текущий согласно V_t и компенсировать возможные изменения освещенности путем применения модификации алгоритма бесшовного монтажа Poisson Blending, описанной в разделе 1.3.5.

Рассмотрим предложенную модель движения фона и основные шаги алгоритма восстановления фона более детально.

1.3.1 Базовый алгоритм построения оптического потока

Рассмотрим упрощенную версию алгоритма построения оптического потока DeerFlow [7] без слагаемого, отвечающего за постоянство градиента яркости, и без компоненты, использующей алгоритм глубокого сопоставления. Искомое отображение из кадра с номером t в кадр с номером $t + 1$, описывающее движение каждой точки, тогда определяется как результат минимизации следующего функционала:

$$O^{t \rightarrow t+1} = \arg \min_O \sum_{\substack{p \in I_t^{(1)} \\ O(p) \in I_{t+1}}} \Phi \left(w_p (C(O(p)) - C(p))^2 \right) + \alpha \sum_{p \in I_t^{(1)}} \Phi \left(\left\| \frac{\partial F_O}{\partial x}(p) \right\|^2 + \left\| \frac{\partial F_O}{\partial y}(p) \right\|^2 \right). \quad (1.3)$$

Первое слагаемое здесь отвечает за минимизацию межкадровой разницы по яркости вдоль искомого отображения O с весами $w_p = (1 + \|\nabla_2 C(p)\|^2)^{-1}$, учитывающими значение модуля пространственного градиента яркости $\nabla_2 C = (\frac{\partial C}{\partial x}, \frac{\partial C}{\partial y})$. Второе слагаемое отвечает за гладкость отображения O посредством минимизации модуля частных производных соответствующего векторного поля смещений $F_O(p) = (O(p) - p)$. Баланс между этими двумя ограничениями контролируется константой α . Оба слагаемых используют устойчивую к выбросам функцию потерь $\Phi(x) = \sqrt{x + \varepsilon^2}$ ($x \geq 0, \varepsilon = 10^{-3}$), которая позволяет увеличить точность вычисления оптического потока около границ объектов. Обе суммы проходят по точкам p пиксельной сетки $I_t^{(1)}$ размера $W \times H$ пикселей в текущем кадре (верхний индекс обозначает шаг в один пиксел).

Вычислим значения $F_O = (U, V, 1)$ на пиксельной сетке $I_t^{(1)}$ с помощью итеративного алгоритма. Обозначим текущее приближение на k -ой итерации как $F_O^k = (U^k, V^k, 1)$ и выразим его через результат предыдущей итерации следующим образом:

$$F_O^k = F_O^{k-1} + \Delta F_O^k, k = 1, \dots, N_{OF}. \quad (1.4)$$

Здесь $\Delta F_O^k = (\Delta U^k, \Delta V^k, 0)$ – искомое приращение векторного поля смещений на k -ой итерации. После подстановки (1.4) в функционал из (1.3), первое слагаемое будет иметь вид:

$$\sum_{\substack{p \in I_t^{(1)} \\ p + F_O^{k-1}(p) \in I_{t+1}}} \Phi \left(w_p (C(p + F_O^{k-1}(p) + \Delta F_O^k(p)) - C(p))^2 \right). \quad (1.5)$$

Для упрощения дальнейших построений перепишем (1.5) как:

$$\sum_{\substack{p \in I_t^{(1)} \\ p + F_O^{k-1}(p) \in I_{t+1}}} \Phi \left(w_p (C(p + F_O^{k-1}(p)) - C(p - \Delta F_O^k(p)))^2 \right). \quad (1.6)$$

В окрестности p разложим C в ряд Тейлора до первого слагаемого в предположении, что ΔF_O^k достаточно мало, получим:

$$\sum_{\substack{p \in I_t^{(1)} \\ p + F_O^{k-1}(p) \in I_{t+1}}} \Phi \left(w_p \left(\Delta C^{k-1}(p) + \frac{\partial C}{\partial x}(p) \Delta U^k(p) + \frac{\partial C}{\partial y}(p) \Delta V^k(p) \right)^2 \right). \quad (1.7)$$

Здесь $\Delta C^{k-1}(p) = (C(p + F_O^{k-1}(p)) - C(p))$ – межкадровая разница по яркости вдоль отображения, полученного на предыдущей итерации.

Введем дискретные координаты $(i, j), i = 1, \dots, H, j = 1, \dots, W$ в рамках пиксельной сетки $I_t^{(1)}$ и добавим в оптимизируемый функционал слагаемое регуляризации с малым весом ε_{reg} , которое при прочих равных отдает приоритет приращениям ΔF_O^k с меньшей нормой. После этого минимизируемый функцио-

нал будет иметь следующий вид:

$$\begin{aligned}
E_{OF} = & \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ 1 \leq i+v_{i,j}^{k-1} \leq H \\ 1 \leq j+u_{i,j}^{k-1} \leq W}} \Phi \left(w_{i,j} (\Delta C_{i,j}^{k-1} + \left(\frac{\partial C}{\partial x} \right)_{i,j} \Delta u_{i,j}^k + \left(\frac{\partial C}{\partial y} \right)_{i,j} \Delta v_{i,j}^k)^2 \right) + \\
& \alpha \sum_{\substack{1 \leq i < H \\ 1 \leq j < W}} \Phi \left((u_{i,j+1}^k - u_{i,j}^k)^2 + (v_{i,j+1}^k - v_{i,j}^k)^2 + \right. \\
& \quad \left. (u_{i+1,j}^k - u_{i,j}^k)^2 + (v_{i+1,j}^k - v_{i,j}^k)^2 \right) + \\
& \alpha \sum_{\substack{i=H \\ 1 \leq j < W}} \Phi \left((u_{i,j+1}^k - u_{i,j}^k)^2 + (v_{i,j+1}^k - v_{i,j}^k)^2 \right) + \\
& \alpha \sum_{\substack{1 \leq i < H \\ j=W}} \Phi \left((u_{i+1,j}^k - u_{i,j}^k)^2 + (v_{i+1,j}^k - v_{i,j}^k)^2 \right) + \\
& \varepsilon_{reg} \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W}} (\Delta u_{i,j}^k)^2 + (\Delta v_{i,j}^k)^2.
\end{aligned} \tag{1.8}$$

Неизвестными здесь являются $\Delta u_{i,j}^k$, $\Delta v_{i,j}^k$, а $u_{i,j}^k = u_{i,j}^{k-1} + \Delta u_{i,j}^k$ и $v_{i,j}^k = v_{i,j}^{k-1} + \Delta v_{i,j}^k$. Важно подчеркнуть, что в первой сумме присутствуют лишь слагаемые, соответствующие точкам, которые после применения смещения с предыдущей итерации F_O^{k-1} остаются в рамках кадра $[1, W] \times [1, H]$.

Так как функция $\Phi(x^2) = \sqrt{x^2 + \varepsilon^2}$ является всюду дифференцируемой и выпуклой относительно x , E_{OF} также выпукла относительно неизвестных и достаточным условием минимума (1.8) является равенство градиента нулю, то есть $(E_{OF})'_{\Delta u_{i,j}^k} = 0$ и $(E_{OF})'_{\Delta v_{i,j}^k} = 0$ для всех $i = 1, \dots, H, j = 1, \dots, W$. Выпишем соответствующие частные производные для внутренних точек с $i \in [2, H-1], j \in [2, W-1]$ и присутствующим слагаемым в первой сумме из (1.8) (для остальных i, j также требуется учесть возможное соседство с границами кадра и выход за

границы кадра после применения смещения с прошлой итерации):

$$\begin{aligned}
(E_{OF})'_{\Delta u_{i,j}^k} = & \\
2 \cdot \left(\Phi_{i,j}^D w_{i,j} \left(\Delta C_{i,j}^{k-1} \left(\frac{\partial C}{\partial x} \right)_{i,j} + \left(\frac{\partial C}{\partial x} \right)_{i,j}^2 \Delta u_{i,j}^k + \left(\frac{\partial C}{\partial x} \right)_{i,j} \left(\frac{\partial C}{\partial y} \right)_{i,j} \Delta v_{i,j}^k \right) + \right. & \\
\alpha \left((2\Phi_{i,j}^S + \Phi_{i-1,j}^S + \Phi_{i,j-1}^S) u_{i,j}^k - \Phi_{i,j}^S u_{i,j+1}^k - \Phi_{i,j}^S u_{i+1,j}^k - \right. & \\
\left. \left. \Phi_{i-1,j}^S u_{i-1,j}^k - \Phi_{i,j-1}^S u_{i,j-1}^k \right) + \varepsilon_{reg} \Delta u_{i,j}^k \right), & \tag{1.9}
\end{aligned}$$

$$\begin{aligned}
(E_{OF})'_{\Delta v_{i,j}^k} = & \\
2 \cdot \left(\Phi_{i,j}^D w_{i,j} \left(\Delta C_{i,j}^{k-1} \left(\frac{\partial C}{\partial y} \right)_{i,j} + \left(\frac{\partial C}{\partial x} \right)_{i,j} \left(\frac{\partial C}{\partial y} \right)_{i,j} \Delta u_{i,j}^k + \left(\frac{\partial C}{\partial y} \right)_{i,j}^2 \Delta v_{i,j}^k \right) + \right. & \\
\alpha \left((2\Phi_{i,j}^S + \Phi_{i-1,j}^S + \Phi_{i,j-1}^S) v_{i,j}^k - \Phi_{i,j}^S v_{i,j+1}^k - \Phi_{i,j}^S v_{i+1,j}^k - \right. & \\
\left. \left. \Phi_{i-1,j}^S v_{i-1,j}^k - \Phi_{i,j-1}^S v_{i,j-1}^k \right) + \varepsilon_{reg} \Delta v_{i,j}^k \right). &
\end{aligned}$$

Здесь $\Phi_{i,j}^D$ и $\Phi_{i,j}^S$ – значения производной $\Phi'(x) = \frac{1}{2\sqrt{x+\varepsilon^2}}$ в соответствующих точках, зависящих от $\Delta u_{i,j}^k, \Delta v_{i,j}^k$, из первого и второго слагаемых минимизируемого функционала. А именно:

$$\begin{aligned}
\Phi_{i,j}^D &= \Phi' \left(w_{i,j} \left(\Delta C_{i,j}^{k-1} + \left(\frac{\partial C}{\partial x} \right)_{i,j} \Delta u_{i,j}^k + \left(\frac{\partial C}{\partial y} \right)_{i,j} \Delta v_{i,j}^k \right)^2 \right), \\
\Phi_{i,j}^S &= \Phi' \left((u_{i,j+1}^k - u_{i,j}^k)^2 + (v_{i,j+1}^k - v_{i,j}^k)^2 + (u_{i+1,j}^k - u_{i,j}^k)^2 + (v_{i+1,j}^k - v_{i,j}^k)^2 \right). \tag{1.10}
\end{aligned}$$

После приравнивания всех частных производных к нулю получим нелинейную систему из $2WH$ уравнений. Решим ее с помощью метода простой итерации в соответствии с алгоритмом IRLS (Iteratively Reweighted Least Squares). Начиная с начального приближения $\Delta u_{i,j}^{k,0} = \Delta v_{i,j}^{k,0} = 0, \forall i,j$ на каждой итерации $l = 1, \dots, N_{fp}$ будем вычислять веса $\Phi_{i,j}^D, \Phi_{i,j}^S$ путем подстановки результата предыдущей итерации: $\Delta u_{i,j}^{k,l-1}, \Delta v_{i,j}^{k,l-1}$. Тогда на каждой итерации получим линейную систему с разреженной матрицей $A_{k,l}$, которую требуется решить:

$$A_{k,l} \begin{bmatrix} \Delta u_{1,1}^{k,l} \\ \Delta v_{1,1}^{k,l} \\ \vdots \\ \Delta u_{H,W}^{k,l} \\ \Delta v_{H,W}^{k,l} \end{bmatrix} = b_{k,l} \tag{1.11}$$

Отметим, что левую часть системы можно представить в виде суммы:

$$A_{k,l} = \begin{bmatrix} A_{k,l,1,1}^D & 0 & \cdots & 0 \\ 0 & A_{k,l,1,2}^D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{k,l,H,W}^D \end{bmatrix} + A_{k,l}^S + \varepsilon_{reg} I, \text{ где} \quad (1.12)$$

$$A_{k,l,i,j}^D = \begin{cases} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, & \text{если } i + v_{i,j}^{k-1} \notin [1, H] \text{ или } j + u_{i,j}^{k-1} \notin [1, W] \\ \Phi_{i,j}^D w_{i,j} \begin{bmatrix} \left(\frac{\partial C}{\partial x}\right)_{i,j}^2 & \left(\frac{\partial C}{\partial x}\right)_{i,j} \left(\frac{\partial C}{\partial y}\right)_{i,j} \\ \left(\frac{\partial C}{\partial x}\right)_{i,j} \left(\frac{\partial C}{\partial y}\right)_{i,j} & \left(\frac{\partial C}{\partial y}\right)_{i,j}^2 \end{bmatrix}, & \text{иначе} \end{cases}$$

Первое слагаемое является неотрицательно определенной матрицей согласно критерию Сильвестра, так как $\Phi_{i,j}^D$ и $w_{i,j}$ всегда положительны. Второе слагаемое $A_{k,l}^S$ – симметричная пятидиагональная матрица, отражающая связи между искомыми значениями приращений для соседних пикселей. По выражению (1.9) можно заметить, что $A_{k,l}^S$ обладает свойством нестроого диагонального преобладания, а все диагональные элементы являются положительными, так как $\Phi_{i,j}^S$ всегда положительно. Поэтому $A_{k,l}^S$ также является неотрицательно определенной матрицей. $\varepsilon_{reg} I$ – положительно определенная диагональная матрица, следовательно сумма этих трех матриц $A_{k,l}$ также является положительно определенной. Таким образом, система (1.11) с симметричной положительно определенной разреженной матрицей имеет единственное решение и может быть численно решена, например, с помощью итеративного метода Гаусса-Зейделя или метода релаксации (достаточным условием сходимости которых является симметричность и положительная определенность матрицы коэффициентов системы).

Переход от (1.6) к (1.7) опирается на предположение о том, что искомые $\Delta u_{i,j}^k, \Delta v_{i,j}^k$ достаточно малы. Поэтому вышеописанный алгоритм напрямую применим лишь для вычисления оптического потока с небольшим межкадровым смещением (не более нескольких пикселей). Однако это ограничение можно обойти путем использования пирамидальной схемы вычисления. То есть, сначала вы-

числим оптический поток для пары соседних кадров в уменьшенном разрешении (достаточно низком, чтобы максимальное смещение не превышало нескольких пикселей) с использованием $F_O^0 \equiv 0$ в качестве начального приближения. Затем увеличим разрешение по ширине и высоте в Q_{pyr} раз и домножим вычисленные вектора смещения на тот же коэффициент. Используя полученное поле векторов смещений в качестве начального приближения, снова применим вышеописанный алгоритм для уточнения оптического потока в текущем разрешении. Затем, будем повторять этот процесс до тех пор, пока не будет достигнуто изначальное разрешение кадра.

Стоит отдельно отметить, что описанный выше алгоритм может сойтись к локальному минимуму функционала (1.3) и в общем случае не гарантирует достижения глобального минимума, что также верно и для полной версии алгоритма DeepFlow. Однако данный алгоритм достаточно хорошо зарекомендовал себя на практике и показывает высокое качество вычисления оптического потока по результатам экспериментальных сравнений алгоритмов.

1.3.2 Непараметрическая модель движения фона

Ранее известные непараметрические модели движения фона либо использовали стандартные алгоритмы построения оптического потока для области известного фона и затем отдельно формулировали задачу интерполяции оптического потока внутрь Ω [23; 31; 32], либо формулировали глобальную задачу оптимизации по всему видео для совместного нахождения цвета и оптического потока в рамках области неизвестного фона Ω [11; 14]. Отличительной особенностью предлагаемого в данной работе подхода является формулирование единой задачи оптимизации для совместного нахождения оптического потока в области известного фона и его гладкой интерполяции в область Ω на текущем кадре. Для этого внесем следу-

ющие основные изменения в базовый алгоритм построения оптического потока, описанный в предыдущем разделе:

- Учет области неизвестного фона Ω в оптимизируемом функционале таким образом, чтобы точки внутри Ω не фигурировали в первом слагаемом из (1.3), отвечающим за минимизацию межкадровой разницы по яркости вдоль искомого отображения. Таким образом, внутри Ω будет учитываться лишь слагаемое, отвечающее за гладкость искомого отображения. Фактически, это приводит к интерполяции векторного поля смещений из окружающей области известного фона внутрь области Ω .
- Будем вычислять один единственный вектор смещения $F_O(p)$ для блока из $s \times s$ пикселей. В результате число неизвестных в задаче оптимизации уменьшается в s^2 раз ценой уменьшения пространственного разрешения получаемого решения. Значение s может быть экспериментально подобрано так, чтобы это изменение не оказывало значимое влияние на качество итогового результата восстановления фона.
- Вычисление одного вектора смещения для целого блока пикселей позволяет также повысить устойчивость итоговой модели к изменениям освещенности фона между соседними кадрами посредством введения нормировки на среднюю яркость блока. То есть, перед вычислением межкадровой разницы по яркости между парой блоков вычтем среднее значение яркости из каждого. Такой подход позволяет уменьшить объем требуемых вычислений по сравнению с использованием дополнительного слагаемого, отвечающего за минимизацию межкадровой разницы градиента яркости вдоль искомого отображения, что ранее использовалось авторами алгоритмов построения оптического потока для этой цели.

Обозначим через $\nabla_{F_O(p)} C(q)$ межкадровую разницу по яркости в точке q со смещением $F_O(p)$: $C(q + F_O(p)) - C(q)$. Тогда оптимизируемый функционал будет

иметь следующий вид:

$$\begin{aligned}
O^{t \rightarrow t+1} = \arg \min_O \sum_{p \in I_t^{(s)}} \Phi \left(\sum_{\substack{q \in P_{s \times s}(p) \\ q \in I_t \setminus \Omega_t \\ q + F_O(p) \in I_{t+1} \setminus \Omega_{t+1}}} w_q \left(\nabla_{F_O(p)} C(q) - \frac{\sum_q w_q \nabla_{F_O(p)} C(q)}{\sum_q w_q} \right)^2 \right) \\
+ \alpha \sum_{p \in I_t^{(s)}} \Phi \left(\left\| \frac{\partial F_O}{\partial x}(p) \right\|^2 + \left\| \frac{\partial F_O}{\partial y}(p) \right\|^2 \right). \tag{1.13}
\end{aligned}$$

Здесь $P_{s \times s}(p)$ обозначает блок размера $s \times s$ пикселей с центром в точке p (то есть, множество из s^2 точек пикселей), а $I_t^{(s)}$ – сетка с шагом s пикселей в кадре I_t . $\Omega_t \subset I_t$ – область неизвестного фона в кадре с номером t . Для упрощения обозначений \sum_q обозначает сумму по тому же множеству значений, что и первая сумма по q (результат суммы \sum_q уже не зависит от q).

В предположении, что s – нечетное, перепишем E_{OF} из (1.8) согласно внешним изменениям (для четного s достаточно соответствующим образом модифицировать диапазон изменения m и n в первой сумме):

$$\begin{aligned}
E_{OF} = \sum_{(j,i,t) \in I_t^{(s)}} \Phi \left(\sum_{\substack{i - \lfloor s/2 \rfloor \leq m \leq i + \lfloor s/2 \rfloor \\ j - \lfloor s/2 \rfloor \leq n \leq j + \lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i,j}^{k-1}, m+v_{i,j}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} \left(\Delta C_{m,n,i,j}^{k-1} - S_{i,j}^{k-1} + \left(\frac{\partial C}{\partial x} \right)_{m,n} \Delta u_{i,j}^k + \right. \right. \\
\left. \left. \left(\frac{\partial C}{\partial y} \right)_{m,n} \Delta v_{i,j}^k \right)^2 \right) + \\
\alpha \sum_{\substack{(j,i,t) \in I_t^{(s)} \\ i+s \leq H \\ j+s \leq W}} \Phi \left((u_{i,j+s}^k - u_{i,j}^k)^2 + (v_{i,j+s}^k - v_{i,j}^k)^2 + \right. \\
\left. (u_{i+s,j}^k - u_{i,j}^k)^2 + (v_{i+s,j}^k - v_{i,j}^k)^2 \right) + \\
\alpha \sum_{\substack{(j,i,t) \in I_t^{(s)} \\ i+s > H \\ j+s \leq W}} \Phi \left((u_{i,j+s}^k - u_{i,j}^k)^2 + (v_{i,j+s}^k - v_{i,j}^k)^2 \right) + \\
\alpha \sum_{\substack{(j,i,t) \in I_t^{(s)} \\ i+s \leq H \\ j+s > W}} \Phi \left((u_{i+s,j}^k - u_{i,j}^k)^2 + (v_{i+s,j}^k - v_{i,j}^k)^2 \right) + \\
\varepsilon_{reg} \sum_{(j,i,t) \in I_t^{(s)}} (\Delta u_{i,j}^k)^2 + (\Delta v_{i,j}^k)^2. \tag{1.14}
\end{aligned}$$

Здесь $\Delta C_{m,n,i,j}^{k-1}$ и $S_{i,j}^{k-1}$, соответственно, обозначают межкадровую разницу по яркости в точке с координатами m,n вдоль отображения с предыдущей итерации

со смещением $(u_{i,j}^{k-1}, v_{i,j}^{k-1})$, и взвешенное среднее разницы по яркости в блоке с центром в i, j вдоль того же отображения:

$$\Delta C_{m,n,i,j}^{k-1} = C(n + u_{i,j}^{k-1}, m + v_{i,j}^{k-1}, t + 1) - C(n, m, t)$$

$$S_{i,j}^{k-1} = \frac{\sum_{\substack{i - \lfloor s/2 \rfloor \leq m \leq i + \lfloor s/2 \rfloor \\ j - \lfloor s/2 \rfloor \leq n \leq j + \lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i,j}^{k-1}, m+v_{i,j}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} \Delta C_{m,n,i,j}^{k-1}}{\sum_{\substack{i - \lfloor s/2 \rfloor \leq m \leq i + \lfloor s/2 \rfloor \\ j - \lfloor s/2 \rfloor \leq n \leq j + \lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i,j}^{k-1}, m+v_{i,j}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n}} \quad (1.15)$$

Значения яркости C в точках с нецелыми координатами определяются путем билинейной интерполяции.

Аналогично предыдущему разделу, приравняв градиент E_{OF} к нулю и введя индекс l итерации IRLS, получим систему:

$$A_{k,l} \begin{bmatrix} \Delta u_{i_1, j_1}^{k,l} \\ \Delta v_{i_1, j_1}^{k,l} \\ \vdots \\ \Delta u_{i_{N_s}, j_{N_s}}^{k,l} \\ \Delta v_{i_{N_s}, j_{N_s}}^{k,l} \end{bmatrix} = b_{k,l} \quad (1.16)$$

Здесь все узлы двумерной сетки $I_t^{(s)}$ с шагом s пикселей пронумерованы от 1 до $N_s = \lceil H/s \rceil \lceil W/s \rceil$. Левая часть системы имеют следующий вид:

$$A_{k,l} = \begin{bmatrix} A_{k,l,i_1,j_1}^D & 0 & \cdots & 0 \\ 0 & A_{k,l,i_2,j_2}^D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{k,l,i_{N_s},j_{N_s}}^D \end{bmatrix} + A_{k,l}^S + \varepsilon_{reg} I \quad (1.17)$$

Первое слагаемое в (1.17) – неотрицательно определенная матрица, так как каждая 2×2 матрица на диагонали неотрицательно определена как сумма неотрицательно определенных матриц (или как нулевая матрица, если множество точек,

по которому осуществляется суммирование, – пустое):

$$\begin{aligned}
A_{k,l,i,j}^D &= \Phi_{i,j}^{D,l} \sum_{\substack{i-\lfloor s/2 \rfloor \leq m \leq i+\lfloor s/2 \rfloor \\ j-\lfloor s/2 \rfloor \leq n \leq j+\lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i,j}^{k-1}, m+v_{i,j}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} \begin{bmatrix} \left(\frac{\partial C}{\partial x}\right)_{m,n}^2 & \left(\frac{\partial C}{\partial x}\right)_{m,n} \left(\frac{\partial C}{\partial y}\right)_{m,n} \\ \left(\frac{\partial C}{\partial x}\right)_{m,n} \left(\frac{\partial C}{\partial y}\right)_{m,n} & \left(\frac{\partial C}{\partial y}\right)_{m,n}^2 \end{bmatrix}, \text{ где} \\
\Phi_{i,j}^{D,l} &= \Phi' \left(\sum_{\substack{i-\lfloor s/2 \rfloor \leq m \leq i+\lfloor s/2 \rfloor \\ j-\lfloor s/2 \rfloor \leq n \leq j+\lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i,j}^{k-1}, m+v_{i,j}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} (\Delta C_{m,n,i,j}^{k-1} - S_{i,j}^{k-1} + \left(\frac{\partial C}{\partial x}\right)_{m,n} \Delta u_{i,j}^{k,l-1} + \left(\frac{\partial C}{\partial y}\right)_{m,n} \Delta v_{i,j}^{k,l-1})^2 \right)
\end{aligned} \tag{1.18}$$

Второе слагаемое в (1.17) – неотрицательно определенная матрица по свойству нестроганого диагонального преобладания, так как $(A_{k,l}^S)_{2r-1, 2c-1} = (A_{k,l}^S)_{2r, 2c} = a_{r,c}^S$, где:

$$a_{r,c}^S = \begin{cases} \mathbf{1}[j_r + s \leq W] \Phi_{i_r, j_r}^{S,l} + \mathbf{1}[i_r + s \leq H] \Phi_{i_r, j_r}^{S,l} + \\ \mathbf{1}[j_r - s \geq 1] \Phi_{i_r, j_r - s}^{S,l} + \mathbf{1}[i_r - s \geq 1] \Phi_{i_r - s, j_r}^{S,l}, & c = r \\ -\Phi_{i_r, j_r}^{S,l}, & (i_c, j_c) = (i_r, j_r + s) \\ -\Phi_{i_r, j_r}^{S,l}, & (i_c, j_c) = (i_r + s, j_r) \\ -\Phi_{i_r, j_r - s}^{S,l}, & (i_c, j_c) = (i_r, j_r - s) \\ -\Phi_{i_r - s, j_r}^{S,l}, & (i_c, j_c) = (i_r - s, j_r) \\ 0, & \text{иначе} \end{cases} \tag{1.19}$$

Значения весов $\Phi_{i,j}^{S,l}$ на l -ой итерации определяются следующим образом:

$$\begin{aligned}
\Phi_{i,j}^{S,l} &= \alpha \Phi' \left(\mathbf{1}[j + s \leq W] (u_{i,j+s}^{k-1} + \Delta u_{i,j+s}^{k,l} - u_{i,j}^{k-1} - \Delta u_{i,j}^{k,l})^2 + \right. \\
&\quad \mathbf{1}[j + s \leq W] (v_{i,j+s}^{k-1} + \Delta v_{i,j+s}^{k,l} - v_{i,j}^{k-1} - \Delta v_{i,j}^{k,l})^2 + \\
&\quad \mathbf{1}[i + s \leq H] (u_{i+s,j}^{k-1} + \Delta u_{i+s,j}^{k,l} - u_{i,j}^{k-1} - \Delta u_{i,j}^{k,l})^2 + \\
&\quad \left. \mathbf{1}[i + s \leq H] (v_{i+s,j}^{k-1} + \Delta v_{i+s,j}^{k,l} - v_{i,j}^{k-1} - \Delta v_{i,j}^{k,l})^2 \right).
\end{aligned} \tag{1.20}$$

Третье слагаемое в (1.17) – положительно определенная диагональная матрица. Следовательно левая часть системы (1.16) – симметричная положительно определенная матрица и у системы существует единственное решение. Для полного

определения системы (1.16) также явно выпишем правую часть (отдельно для четных и нечетных строк):

$$\begin{aligned}
(b_{k,l})_{2r-1} = & \Phi_{i,j}^{D,l} \sum_{\substack{i_r - \lfloor s/2 \rfloor \leq m \leq i_r + \lfloor s/2 \rfloor \\ j_r - \lfloor s/2 \rfloor \leq n \leq j_r + \lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i_r,j_r}^{k-1}, m+v_{i_r,j_r}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} \left(\frac{\partial C}{\partial x} \right)_{m,n} (S_{i,j}^{k-1} - \Delta C_{m,n,i,j}^{k-1}) + \\
& \mathbf{1}[j_r + s \leq W] \Phi_{i_r,j_r}^{S,l} \cdot (u_{i_r,j_r+s}^{k-1} - u_{i_r,j_r}^{k-1}) + \mathbf{1}[i_r + s \leq H] \Phi_{i_r,j_r}^{S,l} \cdot (u_{i_r+s,j_r}^{k-1} - u_{i_r,j_r}^{k-1}) + \\
& \mathbf{1}[j_r - s \geq 0] \Phi_{i_r,j_r-s}^{S,l} \cdot (u_{i_r,j_r-s}^{k-1} - u_{i_r,j_r}^{k-1}) + \mathbf{1}[i_r - s \geq 0] \Phi_{i_r-s,j_r}^{S,l} \cdot (u_{i_r-s,j_r}^{k-1} - u_{i_r,j_r}^{k-1})
\end{aligned} \tag{1.21}$$

$$\begin{aligned}
(b_{k,l})_{2r} = & \Phi_{i,j}^{D,l} \sum_{\substack{i_r - \lfloor s/2 \rfloor \leq m \leq i_r + \lfloor s/2 \rfloor \\ j_r - \lfloor s/2 \rfloor \leq n \leq j_r + \lfloor s/2 \rfloor \\ (n,m,t) \in I_t \setminus \Omega_t \\ (n+u_{i_r,j_r}^{k-1}, m+v_{i_r,j_r}^{k-1}, t+1) \in I_{t+1} \setminus \Omega_{t+1}}} w_{m,n} \left(\frac{\partial C}{\partial y} \right)_{m,n} (S_{i,j}^{k-1} - \Delta C_{m,n,i,j}^{k-1}) + \\
& \mathbf{1}[j_r + s \leq W] \Phi_{i_r,j_r}^{S,l} \cdot (v_{i_r,j_r+s}^{k-1} - v_{i_r,j_r}^{k-1}) + \mathbf{1}[i_r + s \leq H] \Phi_{i_r,j_r}^{S,l} \cdot (v_{i_r+s,j_r}^{k-1} - v_{i_r,j_r}^{k-1}) + \\
& \mathbf{1}[j_r - s \geq 0] \Phi_{i_r,j_r-s}^{S,l} \cdot (v_{i_r,j_r-s}^{k-1} - v_{i_r,j_r}^{k-1}) + \mathbf{1}[i_r - s \geq 0] \Phi_{i_r-s,j_r}^{S,l} \cdot (v_{i_r-s,j_r}^{k-1} - v_{i_r,j_r}^{k-1})
\end{aligned} \tag{1.22}$$

В авторской реализации размер блока $s = 8$ вне зависимости от разрешения и уровня пирамиды, $\alpha = 2.5$, $\varepsilon_{reg} = 10^{-2}$. Таким образом, на каждом уровне пирамиды $l_{pyr} = N_{pyr} \dots 1$ (в авторской реализации $N_{pyr} = 8$), где пространственное разрешение по горизонтали и вертикали уменьшено в $(Q_{pyr})^{l_{pyr}-1}$ раз относительно оригинала, для уточнения векторного поля смещений, моделирующего движение фона, требуется выполнить следующие шаги:

1. Увеличить пространственное разрешение результата с предыдущего уровня в Q_{pyr} раз с помощью билинейной интерполяции и домножить каждый полученный вектор смещения на Q_{pyr} для инициализации F_O^0 на текущем уровне. В случае $l_{pyr} = N_{pyr}$ векторное поле смещений F_O^0 инициализируется нулевыми смещениями. В авторской реализации $Q_{pyr} = 2$.
2. Для каждой итерации $k = 1 \dots N_{OF}$ (в авторской реализации $N_{OF} = 10$) найти приращение $\Delta F_O^k = (\Delta U^k, \Delta V^k, 0)$ и вычислить $F_O^k = F_O^{k-1} + \Delta F_O^k$, что включает в себя следующие шаги:

- 2.1. Инициализировать нулем приращения $\Delta u_{i,j}^{k,0}$, $\Delta v_{i,j}^{k,0}$ для всех точек сетки $I_t^{(s)}$.
- 2.2. Для каждой итерации $l = 1 \dots N_{fp}$ (в авторской реализации $N_{fp} = 5$) вычислить текущие значения весов $\Phi_{i,j}^{D,l}$, $\Phi_{i,j}^{S,l}$ и коэффициенты системы (1.16) по смещениям с $(k-1)$ -ой итерации $u_{i,j}^{k-1}$, $v_{i,j}^{k-1}$ и приращениям с $(l-1)$ -ой итерации $\Delta u_{i,j}^{k,l-1}$, $\Delta v_{i,j}^{k,l-1}$. Затем вычислить решение системы линейных уравнений (1.16) для получения результата l -ой итерации: $\Delta u_{i,j}^{k,l}$, $\Delta v_{i,j}^{k,l}$. В авторской реализации для численного решения системы используется метод релаксации с параметром $\omega = 1.6$ и количеством итераций $N_{SOR} = 25$ (его сходимость обеспечивается тем, что $A_{k,l}$ – симметричная положительно определенная матрица).
- 2.3. В результате k -ой итерации получим приращения $\Delta u_{i,j}^k = \Delta u_{i,j}^{k,N_{fp}}$, $\Delta v_{i,j}^k = \Delta v_{i,j}^{k,N_{fp}}$ и обновленные смещения $u_{i,j}^k = u_{i,j}^{k-1} + \Delta u_{i,j}^k$, $v_{i,j}^k = v_{i,j}^{k-1} + \Delta v_{i,j}^k$.

После вычисления финального векторного поля смещений F_O^{NOF} на сетке $I_t^{(s)}$ в оригинальном разрешении, вычислим соответствующее попиксельное векторное поле на $I_t^{(1)}$ с помощью билинейной интерполяции. Полученный результат затем используется в алгоритме покадрового восстановления фона, описанном в следующих разделах.

1.3.3 Базовый алгоритм покадрового восстановления фона

Как уже было упомянуто выше, предлагаемый алгоритм восстановления фона состоит из трех проходов по входной последовательности кадров. В каждом кадре t в каждом проходе обновляется отображение $V_t : \Omega_t^{(K)} \mapsto I$, которое ставит в соответствие каждой точке p из текущей области определения $\Omega_t^{(K)} \subset \Omega_t$

в рамках области неизвестного фона в текущем кадре $\Omega_t \subset I_t$ точку $q \in I$ с известным фоном. Таким образом, базовый алгоритм восстановления фона состоит из следующих шагов:

1. Инициализировать области определения отображений V_t как $\Omega_t^{(K)} := \emptyset$ для всех номеров кадров t .
2. **Прямой проход.** Для каждого номера кадра $t = 2 \dots T$ расширить область определения V_t путем присваивания:

$$V_t(p) := \begin{cases} O^{t \rightarrow t-1}(p), & O^{t \rightarrow t-1}(p) \in I_{t-1} \setminus \Omega_{t-1} \\ V_{t-1}(O^{t \rightarrow t-1}(p)), & O^{t \rightarrow t-1}(p) \in \Omega_{t-1}^{(K)} \end{cases} \quad (1.23)$$

То есть, если результат отображения в предыдущий кадр согласно модели движения фона $O^{t \rightarrow t-1}(p)$ оказывается в области известного фона $I_{t-1} \setminus \Omega_{t-1}$ или в области определения V_{t-1} , то определим отображение из p в известный фон. Затем требуется восстановить фон в $\Omega_t^{(K)}$ посредством копирования согласно V_t и компенсации возможных изменений освещенности с помощью алгоритма, описанного в разделе 1.3.5.

3. **Пространственное восстановление.** Применить алгоритм пространственного восстановления [29] к оставшейся невосстановленной области на последнем кадре $\Omega_T^{(U)} = \Omega_T \setminus \Omega_T^{(K)}$. Затем присвоить $V_T(p) := p$ для $\forall p \in \Omega_T^{(U)}$. После присваивания получим, что область определения на последнем кадре $\Omega_T^{(K)} = \Omega_T$.
4. **Обратный проход.** Для каждого номера кадра $t = T - 1 \dots 1$ расширить область определения V_t путем присваивания для $p \in \Omega_t^{(U)} = \Omega_t \setminus \Omega_t^{(K)}$:

$$V_t(p) := \begin{cases} O^{t \rightarrow t+1}(p), & O^{t \rightarrow t+1}(p) \in I_{t+1} \setminus \Omega_{t+1} \\ V_{t+1}(O^{t \rightarrow t+1}(p)), & O^{t \rightarrow t+1}(p) \in \Omega_{t+1}^{(K)} \end{cases} \quad (1.24)$$

Затем восстановить фон во всех p , где в текущем кадре в ходе обратного прохода было определено отображение V_t .

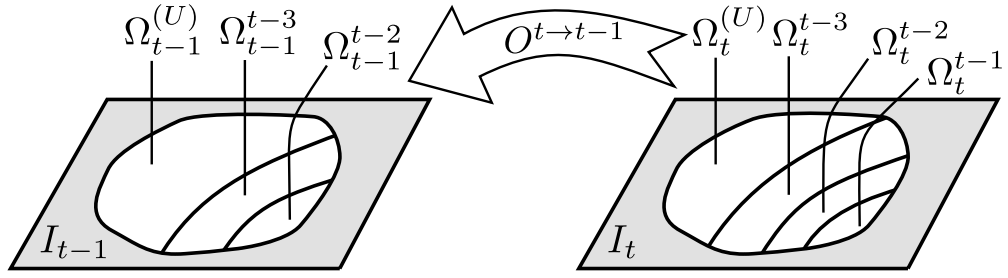


Рисунок 1.4 — Иллюстрация разложения области Ω_t на набор непересекающихся областей определения отображений в различные кадры $\Omega_t^{t_i}$ и область $\Omega_t^{(U)}$, где отображение еще не определено, в рамках базового алгоритма покадрового восстановления фона.

5. **Пространственное восстановление.** Применить алгоритм пространственного восстановления [29] к оставшейся невосстановленной области на первом кадре $\Omega_1^{(U)}$ и присвоить $V_1(p) := p$ для $\forall p \in \Omega_1^{(U)}$.
6. **Прямой проход.** Для каждого номера кадра $t = 2 \dots T$ расширить область определения V_t путем идентичного первому прямому проходу присваивания для точек с еще не восстановленным фоном $p \in \Omega_t^{(U)}$. Затем восстановить фон во всех точках, где в ходе второго прямого прохода было определено V_t .

Если выполнено описанное в начале главы предположение о том, что для каждой точки p из остающейся скрытой на протяжении всей видеопоследовательности области найдется композиция межкадровых отображений $O^{t \rightarrow t \pm 1}$, отображающая p в первый или последний кадр, то описанный выше алгоритм полностью восстановит фон в Ω . То есть, результат пространственного восстановления в первом и последнем кадрах будет распространен путем копирования (и компенсации условий освещенности) по всей видеопоследовательности и не останется других областей, требующих пространственного восстановления.

Стоит отметить, что на каждом шаге описанного выше алгоритма область Ω_t можно представить в виде объединения непересекающихся областей:

$$\Omega_t = \Omega_t^{(U)} \cup \Omega_t^{t_1} \cup \dots \cup \Omega_t^{t_n}, \text{ где } \Omega_t^{t_i} = \{p | V_t(p) \in I_{t_i}\}. \quad (1.25)$$

Данное объединение включает область $\Omega_t^{(U)}$, где отображение V_t на данном шаге еще не определено, и набор областей определения отображений в различные ис-

ходные кадры с номерами $t_1 \dots t_n$. То есть, отображение V_t можно разложить на отображения в отдельные исходные кадры $V_t^{t_i} : \Omega_t^{t_i} \mapsto I_{t_i}$. Иллюстрация данного разложения приведена на рисунке 1.4.

Для проведения восстановления значений яркости (цвета) фона во всех точках пиксельной сетки каждого кадра $I_t^{(1)}$ в рамках области Ω_t достаточно знать лишь значения V_t в узлах той же сетки. Поэтому будем вычислять V_t в узлах пиксельной сетки $I_t^{(1)}$; если результат отображения в предыдущий кадр оказывается между узлов сетки, то композиция $V_{t\pm 1}(O^{t \rightarrow t\pm 1}(p))$ вычисляется путем предварительного округления координат точки $q = O^{t \rightarrow t\pm 1}(p)$ до ближайшего узла сетки. Точнее, вектор пространственного смещения согласно $O^{t \rightarrow t\pm 1}(p)$ прибавляется к такому же вектору смещения, соответствующему значению $V_{t\pm 1}$ в точке q с округленными координатами, что определяет итоговое значение $V_t(p)$. В рамках предложенного алгоритма такой подход был выбран в связи с тем, что, как было отмечено выше, V_t состоит из набора отображений в отдельные исходные кадры, и на границах между областями определения отображений в различные кадры невозможно применить традиционные методы интерполяции значений между узлами сетки.

1.3.4 Алгоритм покадрового восстановления с уточнением

Основной проблемой базового алгоритма восстановления фона, описанного в предыдущем разделе, является возможность накопления ошибки модели движения фона в значениях V_t . Если ближайший кадр, где видим скрытый в текущем кадре фрагмент фона, отстоит от текущего на n кадров, то для его восстановления требуется вычислить композицию n межкадровых отображений $O^{t \rightarrow t\pm 1}$. Малые погрешности $O^{t \rightarrow t\pm 1}$ и погрешности, вносимые описанным в предыдущем разделе округлением координат при вычислении композиции отображений, могут накапливаться в значениях V_t и, как следствие, приводить к появлению ви-

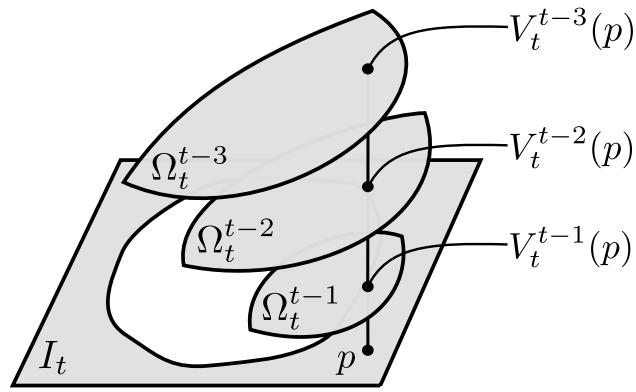


Рисунок 1.5 — Иллюстрация многозначного отображения V_t , используемого в алгоритме покадрового восстановления с уточнением, которое допускает наличие нескольких отображений в различные исходные кадры в каждой точке p . Его можно разложить на набор отображений в различные исходные кадры $V_t^{t_i}$ с возможно пересекающимися областями определения $\Omega_t^{t_i}$, вложенными в расширенную область неизвестного фона $\bar{\Omega}_t$.

димых швов в результате восстановления. Для компенсации такого накопления ошибки предлагается алгоритм покадрового восстановления фона с уточнением. Ключевой идеей предложенного подхода является введение перекрытий между областями определения отображений в различные исходные кадры, что позволяет определить оптимизационную задачу по нахождению оптимального выравнивания между фрагментами фона, копируемыми из различных исходных кадров в текущий.

Формально, определим V_t как многозначную функцию, то есть для каждой точки p : $V_t(p) \subset I$, где мощность множества $|V_t(p)| = N(p)$, причем $\forall q_1, q_2 \in V_t(p) : q_1 \neq q_2 \Rightarrow \exists t : q_1, q_2 \in I_t$, то есть в каждой точке p есть $N(p)$ отображений в различные исходные кадры. Также расширим допустимую область определения V_t до $\bar{\Omega}_t$, которая является результатом морфологического расширения Ω_t с квадратным структурным элементом размера $(2d + 1) \times (2d + 1)$, где d – параметр алгоритма (в авторской реализации $d = 6$ пикселям). Такое расширение области определения позволяет также ввести перекрытия между областями определения отображений в различные исходные кадры и известным фоном в текущем кадре, окружающим область Ω_t , что дает возможность также учитывать выравнивание с известным фоном в вышеупомянутой задаче оптимизации. Так же, как и в преды-

дущем разделе, $\Omega_t^{t_i} \subset \overline{\Omega}_t$ обозначает область определения отображения из текущего кадра t в кадр с номером t_i (однако теперь допускается $\Omega_t^{t_i} \cap \Omega_t^{t_j} \neq \emptyset$ для $i \neq j$), а $V_t^{t_i} : \Omega_t^{t_i} \mapsto I_{t_i}$ обозначает само отображение. Поясняющая иллюстрация приведена на рисунке 1.5.

Алгоритм будет состоять из трех проходов по входной последовательности кадров, таких же как в базовом алгоритме, описанном в предыдущем разделе. Однако для упрощения последующих построений положим, что после каждого из шагов применения алгоритма пространственного восстановления выполняется присваивание $\Omega_t := \Omega_t^{(U)}$ и $\Omega_t^{(K)} := \emptyset$ для всех номеров кадров t . То есть, после завершения каждого из проходов области неизвестного фона Ω_t обновляются таким образом, чтобы не включать подобласти с уже восстановленным на предыдущих шагах фоном, и области определения отображений V_t , соответственно, становятся пустыми. После данной модификации обработку каждого кадра с номером t в каждом из проходов в алгоритме восстановления фона с уточнением можно описать следующим набором шагов:

1. Вычислить отображение в известный фон в предыдущем кадре с областью определения $\Omega_t^{t_{\pm 1}} \subset \Omega_t$ аналогично базовому алгоритму. Затем расширить эту область определения так, чтобы она имела перекрытие с известным фоном. Формально, пусть $\overline{\Omega}_t^{t_{\pm 1}}$ – результат морфологического расширения этой области с квадратным структурным элементом размера $(2d + 1) \times (2d + 1)$. Тогда расширим область определения путем присваивания $\Omega_t^{t_{\pm 1}} := \Omega_t^{t_{\pm 1}} \cup (\overline{\Omega}_t^{t_{\pm 1}} \setminus \Omega_t)$; в добавленных точках $V_t^{t_{\pm 1}}$ точно так же определяется путем копирования значений из отображения $O^{t \rightarrow t_{\pm 1}}$. Отметим, что в итоговой области определения $\Omega_t^{t_{\pm 1}}$ могут присутствовать точки p , которые отображаются за пределы области известного фона, то есть $V_t^{t_{\pm 1}}(p) \notin I \setminus \Omega$. Также по построению $\Omega_t^{t_{\pm 1}} \subset \overline{\Omega}_t$.
2. Для точек $p \in \overline{\Omega}_t$ вычислить отображения в другие исходные кадры с помощью $V_t(p) := V_{t_{\pm 1}}(O^{t \rightarrow t_{\pm 1}}(p))$, если $O^{t \rightarrow t_{\pm 1}}(p) \in \Omega_{t_{\pm 1}}^{(K)}$. Важно подчеркнуть, что здесь также рассматриваются $p \in \Omega_t^{t_{\pm 1}}$ (тогда в p будет опре-

делено отображение как в предыдущий, так и в другие исходные кадры) и что $V_{t\pm 1}(O^{t \rightarrow t\pm 1}(p))$ может включать более, чем одну точку. В общем случае это приводит к получению многозначного отображения V_t с текущей областью определения $\Omega_t^{(K)} \subset \bar{\Omega}_t$. Аналогично базовому алгоритму, V_t вычисляется лишь в узлах пиксельной сетки $I_t^{(1)}$, а композиция отображений вычисляется с использованием округления координат.

3. Скорректировать форму полученного на предыдущем шаге многозначного отображения V_t . Для каждого $t_i, i = 1 \dots n$ (номера кадров, в которые есть отображения в рамках V_t , в порядке увеличения модуля разницы с t):

- 3.1. Если для всех точек $p \in I_t^{(1)}$ таких, что $p \in \Omega_t^{t_i}$ и $V_t^{t_i}(p) \in I \setminus \Omega$, существует кадр $j < i$, что $p \in \Omega_t^{t_j}$ и $V_t^{t_j}(p) \in I \setminus \Omega$, то целиком удалить отображение в кадр t_i (каждое следующее отображение должно позволять восстановить фон хотя бы в одном дополнительном пикселе, который невозможно было восстановить ни одним из предыдущих отображений). Обозначим множество всех таких пикселей p , которые можно восстановить с помощью отображения в кадр t_i , но нельзя восстановить ни одним из предыдущих отображений, как $\Delta\Omega_t^{t_i}$.

- 3.2. Если отображение в кадр с номером t_i не было удалено на предыдущем шаге, то скорректировать область определения этого отображения. Если оно не было удалено, то $\Delta\Omega_t^{t_i} \neq \emptyset$; возьмем $\overline{\Delta\Omega_t^{t_i}}$ – результат морфологического расширения этой области с тем же структурным элементом, что использовался выше. Тогда присвоим области определения $\Omega_t^{t_i} := \overline{\Delta\Omega_t^{t_i}} \cap (\bigcup_{j=1}^i \Delta\Omega_t^{t_j} \cup (\bar{\Omega}_t \setminus \Omega_t))$, чтобы обеспечить перекрытие с ранее восстановленными областями и известным фоном в текущем кадре. В новых точках, добавленных в $\Omega_t^{t_i}$, значение отображения $V_t^{t_i}$ определяется простой экстраполяцией посредством копирования.

Несложно заметить, что после корректировки формы V_t , количество отображений в различные исходные кадры $N(p)$ в любой точке $p \in I_t^{(1)}$ ни при каких условиях не может превосходить $(2d + 1)^2$.

4. Уточнить V_t путем решения задачи оптимизации (1.26), используя результат предыдущего шага в качестве начального приближения. В следующем шаге и при вычислении отображения из следующего кадра $V_{t\pm 1}$ уже использовать результат уточнения V_t .
5. Восстановить фон в текущем кадре с использованием V_t . В каждой точке p из всех $N(p)$ доступных отображений в различные исходные кадры для восстановления фона использовать отображение в ближайший по номеру кадр (с дополнительным условием, что результат отображения должен лежать в области известного фона).

Таким образом, результатом третьего шага будет многозначное отображение V_t с перекрытиями ширины d как между отображениями в различные исходные кадры, так и между отображением в каждый исходный кадр и известным фоном, окружающим Ω_t в текущем кадре. Важно подчеркнуть, что, в отличие от базового алгоритма, здесь результат отображения не всегда оказывается в области известного фона. Положим $V_t(p) = \{V_t^{v_1(p)}(p), \dots, V_t^{v_{N(p)}(p)}(p)\}$, где $v_i(p), i = 1 \dots N(p)$ – упорядоченные по возрастанию модуля разницы с t номера исходных кадров, в которые определены отображения в точке p . Если $V_t^{v_i(p)}(p)$ для всех i и p из области определения всегда лежит в области известного фона, то вышеупомянутую задачу оптимизации можно сформулировать следующим образом:

$$V_t = \arg \min_{V_t} (E_D(V_t) + \lambda E_S(V_t)), \text{ где} \quad (1.26)$$

$$E_D(V_t) = \sum_{\substack{p \in I_t^{(1)} \\ p \in \bar{\Omega}_t \setminus \Omega_t}} \sum_{i=1}^{N(p)} \Phi \left(\left(C(V_t^{v_i(p)}(p)) - C(p) \right)^2 \right) +$$

$$\sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t}} \sum_{i=1}^{N(p)-1} \Phi \left(\left(C(V_t^{v_{i+1}(p)}(p)) - C(V_t^{v_i(p)}(p)) \right)^2 \right),$$

$$E_S(V_t) = \sum_{i=1}^n \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t^{t_i}}} \left\| \frac{\partial F_{V_t^{t_i}}(p)}{\partial x} \right\|^2 + \left\| \frac{\partial F_{V_t^{t_i}}(p)}{\partial y} \right\|^2.$$

Иначе из первой и второй сумм в E_D нужно исключить все слагаемые, содержащие отображение за пределы области известного фона. Как и ранее, t_1, \dots, t_n – множество номеров кадров, в которые определены отображения в рамках V_t , а $F_{V_t^{t_i}}(p) = (V_t^{t_i}(p) - p)$ – вектор смещения согласно отображению $V_t^{t_i}$. $\Phi(x) = \sqrt{x + \varepsilon^2}$ ($\varepsilon = 10^{-3}$) – устойчивая к выбросам функция потерь. Первая сумма в E_D отвечает за выравнивание копируемых из других кадров в текущий фрагментов фона с окружающим фоном в текущем кадре $\bar{\Omega}_t \setminus \Omega_t$. Вторая сумма в E_D отвечает за выравнивание копируемых из различных кадров фрагментов между собой. E_S отвечает за гладкость каждого из отображений $V_t^{t_i}$ из текущего кадра в кадр с номером t_i .

Для решения задачи (1.26) применим схожую технику, которая ранее использовалась для решения задачи построения оптического потока. Воспользуемся тем, что доступно начальное приближение $F_{V_t^{v_i(p)}}(p) = (U_x^{v_i(p)}(p), U_y^{v_i(p)}(p), v_i(p) - t)$ и будем искать приращение $\Delta F_{V_t^{v_i(p)}}(p) = (\Delta U_x^{v_i(p)}(p), \Delta U_y^{v_i(p)}(p), 0)$. Тогда после разложения C в ряд Тейлора до первого члена в соответствующих окрестностях (в предположении, что искомое приращение достаточно мало) и подстановки в E_D получим:

$$\begin{aligned} & \sum_{\substack{p \in I_t^{(1)} \\ p \in \bar{\Omega}_t \setminus \Omega_t}} \sum_{i=1}^{N(p)} \Phi \left(\left(\Delta C^{v_i(p)}(p) + \frac{\partial C}{\partial x}(p_{v_i(p)}) \Delta U_x^{v_i(p)}(p) + \frac{\partial C}{\partial y}(p_{v_i(p)}) \Delta U_y^{v_i(p)}(p) \right)^2 \right) + \\ & \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t}} \sum_{i=1}^{N(p)-1} \Phi \left(\left(\frac{\partial C}{\partial x}(p_{v_{i+1}(p)}) \Delta U_x^{v_{i+1}(p)}(p) + \frac{\partial C}{\partial y}(p_{v_{i+1}(p)}) \Delta U_y^{v_{i+1}(p)}(p) - \right. \right. \\ & \left. \left. \frac{\partial C}{\partial x}(p_{v_i(p)}) \Delta U_x^{v_i(p)}(p) - \frac{\partial C}{\partial y}(p_{v_i(p)}) \Delta U_y^{v_i(p)}(p) + \Delta C^{v_{i+1}(p), v_i(p)}(p) \right)^2 \right) \end{aligned} \quad (1.27)$$

Здесь $p_l = p + F_{V_t^l}(p) \in I_l$ – результат отображения точки p в кадр с номером l согласно начальному приближению, а $\Delta C^l(p) = C(p_l) - C(p)$ – межкадровая

разница по яркости между кадром с номером l и текущим кадром t в точке p , $\Delta C^{l_1, l_2}(p) = C(p_{l_1}) - C(p_{l_2})$ – разница по яркости между кадрами с номерами l_1 и l_2 в точках, полученных в результате отображения точки p в эти кадры согласно начальному приближению многозначного отображения V_t .

Введем дискретные координаты (i, j, l) , где (i, j) – пространственные координаты на сетке $I_t^{(1)}$, а l – номер исходного кадра, в который определено отображение из (i, j) . Тогда оптимизируемый функционал можно переписать как:

$$\begin{aligned}
E_{ref} = & \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \bar{\Omega}_t \setminus \Omega_t}} \sum_{l=v_1(j, i, t)}^{v_N(j, i, t)(j, i, t)} \Phi \left(\left(\Delta C_{i, j}^l + \left(\frac{\partial C}{\partial x} \right)_{i, j, l} \Delta u_{x, i, j}^l + \left(\frac{\partial C}{\partial y} \right)_{i, j, l} \Delta u_{y, i, j}^l \right)^2 \right) + \\
& \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \Omega_t}} \sum_{(l_1, l_2) \in P(j, i, t)} \Phi \left(\left(\Delta C_{i, j}^{l_1, l_2} + \left(\frac{\partial C}{\partial x} \right)_{i, j, l_1} \Delta u_{x, i, j}^{l_1} + \left(\frac{\partial C}{\partial y} \right)_{i, j, l_1} \Delta u_{y, i, j}^{l_1} - \right. \right. \\
& \left. \left. \left(\frac{\partial C}{\partial x} \right)_{i, j, l_2} \Delta u_{x, i, j}^{l_2} - \left(\frac{\partial C}{\partial y} \right)_{i, j, l_2} \Delta u_{y, i, j}^{l_2} \right)^2 \right) + \\
& \lambda \sum_{l=t_1}^{t_n} \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \Omega_t^l \\ (j+1, i, t) \in \Omega_t^l}} (u_{x, i, j+1}^l + \Delta u_{x, i, j+1}^l - u_{x, i, j}^l - \Delta u_{x, i, j}^l)^2 + \\
& \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \Omega_t^l \\ (j, i+1, t) \in \Omega_t^l}} (u_{y, i, j+1}^l + \Delta u_{y, i, j+1}^l - u_{y, i, j}^l - \Delta u_{y, i, j}^l)^2 + \\
& \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \Omega_t^l \\ (j, i+1, t) \in \Omega_t^l}} (u_{x, i+1, j}^l + \Delta u_{x, i+1, j}^l - u_{x, i, j}^l - \Delta u_{x, i, j}^l)^2 + \\
& \sum_{\substack{1 \leq i \leq H \\ 1 \leq j \leq W \\ (j, i, t) \in \Omega_t^l \\ (j, i+1, t) \in \Omega_t^l}} (u_{y, i+1, j}^l + \Delta u_{y, i+1, j}^l - u_{y, i, j}^l - \Delta u_{y, i, j}^l)^2
\end{aligned} \tag{1.28}$$

Здесь $P(j, i, t)$ – множество всех пар номеров кадров $(v_{k+1}(j, i, t), v_k(j, i, t))$, $1 \leq k < N(j, i, t)$ (возможно пустое), а $\left(\frac{\partial C}{\partial x} \right)_{i, j, l}$, $\left(\frac{\partial C}{\partial y} \right)_{i, j, l}$ – частные производные C в точке l -ого кадра, являющейся результатом отображения из точки (i, j) текущего кадра согласно начальному приближению V_t . Так же, как и в описанном в разделе 1.3.2 алгоритме построения оптического потока, добавим в оптимизируемый функционал L_2 -регуляризацию на все $\Delta u_{x, i, j}^l, \Delta u_{y, i, j}^l$ с малым весом $\varepsilon_{reg} = 10^{-2}$, чтобы при прочих равных отдавать предпочтение приращениям с меньшей нормой.

Аналогично E_{OF} из (1.8), достаточным условием минимума E_{ref} является равенство градиента нулю, что эквивалентно системе уравнений: $(E_{ref})'_{\Delta u_{x, i, j}^l} = 0$, $(E_{ref})'_{\Delta u_{y, i, j}^l} = 0$ для всех (i, j, l) из области определения многозначного отображе-

ния. Система снова является нелинейной, поэтому введем индекс k метода простой итерации согласно алгоритму IRLS. На каждой итерации получим линейную систему уравнений:

$$A_k \begin{bmatrix} \Delta u_{x,i_1,j_1}^{k,l_1} \\ \Delta u_{y,i_1,j_1}^{k,l_1} \\ \vdots \\ \Delta u_{x,i_{N_{ref}},j_{N_{ref}}}^{k,l_{N_{ref}}} \\ \Delta u_{y,i_{N_{ref}},j_{N_{ref}}}^{k,l_{N_{ref}}} \end{bmatrix} = b_k \quad (1.29)$$

Здесь все (i,j,l) из области определения пронумерованы от 1 до N_{ref} так, что диапазон $[1, N_{l_1}]$ содержит все точки из области определения отображения в кадр с номером l_1 , диапазон $[N_{l_1} + 1, N_{l_1} + N_{l_2}]$ содержит все точки из области определения отображения в кадр l_2 и так далее для всех исходных кадров, в которые есть отображения в рамках V_t , в порядке увеличения модуля разницы с t . Матрицу из левой части можно выразить как сумму:

$$A_k = \begin{bmatrix} A_{k,i_1,j_1,l_1}^D & 0 & \cdots & 0 \\ 0 & A_{k,i_2,j_2,l_2}^D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & A_{k,i_{N_{ref}},j_{N_{ref}},l_{N_{ref}}}^D \end{bmatrix} + A_k^R + \lambda A_k^S + \varepsilon_{reg} I \quad (1.30)$$

Первое слагаемое в (1.30) аналогично первому слагаемому в (1.17) является неотрицательно определенной матрицей согласно критерию Сильвестра, так как все 2×2 матрицы на диагонали неотрицательно определены:

$$A_{k,i,j,l}^D = \begin{cases} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, & \text{если } (j,i,t) \in \Omega_t \text{ или } (j + u_{x,i,j}^l, i + u_{y,i,j}^l) \notin I \setminus \Omega \\ \Phi_{i,j,l}^{D,k} \begin{bmatrix} \left(\frac{\partial C}{\partial x}\right)_{i,j,l}^2 & \left(\frac{\partial C}{\partial x}\right)_{i,j,l} \left(\frac{\partial C}{\partial y}\right)_{i,j,l} \\ \left(\frac{\partial C}{\partial x}\right)_{i,j,l} \left(\frac{\partial C}{\partial y}\right)_{i,j,l} & \left(\frac{\partial C}{\partial y}\right)_{i,j,l}^2 \end{bmatrix}, & \text{иначе} \end{cases} \quad (1.31)$$

$$\Phi_{i,j,l}^{D,k} = \Phi' \left(\left(\Delta C_{i,j}^l + \left(\frac{\partial C}{\partial x}\right)_{i,j,l} \Delta u_{x,i,j}^{k-1,l} + \left(\frac{\partial C}{\partial y}\right)_{i,j,l} \Delta u_{y,i,j}^{k-1,l} \right)^2 \right)$$

Третье слагаемое в (1.30) аналогично второму слагаемому в (1.17) является неотрицательно определенной матрицей согласно критерию нестрогого диаго-

нального преобладания, так как $(A_k^S)_{2r-1,2c-1} = (A_k^S)_{2r,2c} = a_{r,c}^S$, где:

$$a_{r,c}^S = \begin{cases} N_{i_r, j_r, l_r}^{\text{nbr}}, & c = r \\ -1, & (i_c, j_c, l_c) = (i_r, j_r + 1, l_r) \\ -1, & (i_c, j_c, l_c) = (i_r + 1, j_r, l_r) \\ -1, & (i_c, j_c, l_c) = (i_r, j_r - 1, l_r) \\ -1, & (i_c, j_c, l_c) = (i_r - 1, j_r, l_r) \\ 0, & \text{иначе} \end{cases} \quad (1.32)$$

Здесь $N_{i,j,l}^{\text{nbr}}$ – количество пространственных соседей (i,j) , лежащих в пределах области определения отображения в l -ый кадр Ω_t^l (число в диапазоне $[0,4]$).

Второе слагаемое в (1.30) существенным образом отличается от всех матриц, рассмотренных ранее в разделах 1.3.1 и 1.3.2. Оно соответствует второй сумме в (1.28), которая отвечает за выравнивание между собой фрагментов фона, копируемых из различных исходных кадров в текущий. Покажем неотрицательную определенность матрицы A_k^R посредством ее разложения на сумму неотрицательно определенных матриц:

$$A_k^R = \sum_{(j,i,t) \in \Omega_t} \sum_{\substack{(l,l') \in P(j,i,t) \\ (j+u_{x,i,j}^l, i+u_{y,i,j}^l) \in I \setminus \Omega \\ (j+u_{x,i,j}^{l'}, i+u_{y,i,j}^{l'}) \in I \setminus \Omega}} \Phi_{i,j,l,l'}^{R,k}(a_{i,j,l,l'}^R)(a_{i,j,l,l'}^R)^T, \quad \text{где} \quad (1.33)$$

$$\Phi_{i,j,l,l'}^{R,k} = \Phi' \left(\left(\Delta C_{i,j}^{l,l'} + \left(\frac{\partial C}{\partial x} \right)_{i,j,l} \Delta u_{x,i,j}^{k-1,l} + \left(\frac{\partial C}{\partial y} \right)_{i,j,l} \Delta u_{y,i,j}^{k-1,l} - \left(\frac{\partial C}{\partial x} \right)_{i,j,l'} \Delta u_{x,i,j}^{k-1,l'} - \left(\frac{\partial C}{\partial y} \right)_{i,j,l'} \Delta u_{y,i,j}^{k-1,l'} \right)^2 \right),$$

$$a_{i,j,l,l'}^R = \begin{bmatrix} a_1 \\ \vdots \\ a_{2N_{ref}} \end{bmatrix}, a_{2r-1} = \begin{cases} \left(\frac{\partial C}{\partial x}\right)_{i,j,l}, & (i_r, j_r, l_r) = (i, j, l) \\ -\left(\frac{\partial C}{\partial x}\right)_{i,j,l'}, & (i_r, j_r, l_r) = (i, j, l') \\ 0, & \text{иначе} \end{cases}$$

$$a_{2r} = \begin{cases} \left(\frac{\partial C}{\partial y}\right)_{i,j,l}, & (i_r, j_r, l_r) = (i, j, l) \\ -\left(\frac{\partial C}{\partial y}\right)_{i,j,l'}, & (i_r, j_r, l_r) = (i, j, l') \\ 0, & \text{иначе} \end{cases}$$

Каждое слагаемое в (1.33) является неотрицательно определенной матрицей, так как $\Phi_{i,j,l,l'}^{R,k}$ всегда положительно и aa^T образует неотрицательно определенную матрицу для любого вектор-столбца a .

Таким образом, A_k из (1.30) – симметричная положительно определенная матрица и у линейной системы (1.29) существует единственное решение. Тогда алгоритм уточнения многозначного отображения V_t в текущем кадре состоит из следующих шагов:

1. Инициализировать $\Delta u_{x,i,j}^{0,l} = \Delta u_{y,i,j}^{0,l} = 0$ для всех (i,j,l) из области определения многозначного отображения.
2. Для каждого $k = 1 \dots N_{rfp}$ вычислить значения весов $\Phi_{i,j,l}^{D,k}, \Phi_{i,j,l,l'}^{R,k}$ на текущей итерации и коэффициенты системы (1.29). Решить систему с симметричной положительно определенной матрицей с помощью метода сопряженных градиентов с N_{CG} итерациями. Получим значения $\Delta u_{x,i,j}^{k,l}, \Delta u_{y,i,j}^{k,l}$ – результат k -ой итерации.
3. Выполнить присваивание $u_{x,i,j}^l := u_{x,i,j}^l + \Delta u_{x,i,j}^{N_{rfp},l}, u_{y,i,j}^l := u_{y,i,j}^l + \Delta u_{y,i,j}^{N_{rfp},l}$ для каждого (i,j) из области определения отображения в l -ый кадр.

В авторской реализации экспериментально были подобраны следующие значения параметров: $\lambda = 200, N_{rfp} = 1, N_{CG} = 50$.

Стоит отметить, что с помощью вышеописанного алгоритма уточнения невозможно исправить крупные ошибки модели движения фона (больше, чем

несколько пикселей), однако за счет повторного применения на каждом кадре предложенный алгоритм позволяет предотвратить накопление ошибок модели движения фона, если они достаточно малы. Ценой увеличения объема вычислений можно ввести несколько внешних итераций с повторным разложением в ряд Тейлора в рамках алгоритма уточнения, аналогично алгоритму построения оптического потока, описанному в разделе 1.3.1. Однако экспериментально было установлено, что это может приводить к ухудшению временной стабильности результата восстановления фона, если отклонение результата уточнения от начального приближения становится слишком большим.

1.3.5 Компенсация изменений условий освещенности

В предыдущих разделах был описан алгоритм построения отображения из области неизвестного фона Ω в известный фон на базе значений яркости пикселей C . В случае обработки RGB видео, где $C : I \mapsto \mathbb{R}^3$ – значения цвета пикселей, требуется сначала вычислить значения яркости и затем построить требуемое отображение согласно описанному выше алгоритму. При копировании фона согласно построенному отображению необходимо учесть уже все три канала входной видеопоследовательности. Однако, как уже было упомянуто ранее, изменение условий освещенности от кадра к кадру может приводить к появлению видимых швов в результате работы алгоритма, если восстановление проводить простым копированием. Также к появлению видимых швов могут приводить оставшиеся ошибки выравнивания между фрагментами фона, восстановленными путем копирования из различных исходных кадров.

Для снижения заметности швов предлагается модификация метода бесшовного монтажа, известного как Poisson Blending [20]. Существующие обобщения этого метода для задачи восстановления фона в видео [9] используют дополнительное слагаемое, накладывающее ограничение временной стабильности с уче-

том движения в восстановленной области. Отличительной особенностью предложенной модификации является использование адаптивных весов в слагаемом, отвечающим за поддержание временной стабильности, что позволяет сохранять возможность равномерного изменения цвета всей восстанавливаемой области от кадра к кадру и, в то же время, усиливать ограничения временной стабильности в областях локально неконсистентного изменения цвета. А именно, предложенный алгоритм финального восстановления фона по вычисленному отображению V_t включает вычисление минимума следующего функционала:

$$\begin{aligned}
E_{PB}[X] = & \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t^{(K)} \cap \bar{\Omega}_t^{(1)} \\ V_t(p) \in I \setminus \Omega}} \|\nabla_2 X(p) - G_t(p)\|^2 + \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t^{(K)} \cap (\bar{\Omega}_t^{(1)} \setminus \Omega_t) \\ V_t(p) \in I \setminus \Omega}} w_p^{PB} \|X(p) - C(p)\|^2 + \\
& \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t^{(K)} \cap \Omega_t \\ V_t(p) \in I \setminus \Omega}} (1 - w_p^{PB}) \|X(p) - C(O^{t \rightarrow t \pm 1}(p))\|^2, \text{ где} \tag{1.34} \\
w_p^{PB} = & \frac{1}{1 + \sigma \|\nabla_2 X^{\text{init}}(p) - G_t(p)\|^2}
\end{aligned}$$

В качестве неизвестных здесь выступают значения цвета пикселей $X(p)$ в области $\Omega_t^{(K)} \cap \bar{\Omega}_t^{(1)}$, где $\Omega_t^{(K)} \subset \bar{\Omega}_t$ – область определения отображения V_t , а $\bar{\Omega}_t^{(1)}$ – результат морфологического расширения области неизвестного фона Ω_t с структурным элементом в виде квадрата размером 3×3 пиксела. Также стоит отметить, что рассматриваются только такие точки p , которые отображаются V_t в область известного фона (что в общем случае верно не для всех p из области определения, как описано в разделе 1.3.4). Первое слагаемое (1.34) штрафует отклонение векторного поля пространственных градиентов результата $\nabla_2 X$ от эталонного векторного поля градиентов в текущем кадре G_t , которое получено путем копирования полей градиентов исходных кадров в текущий согласно V_t . Второе слагаемое – мягкое граничное условие, штрафующее отклонение результата от известного фона, окружающего область Ω_t в текущем кадре, в пределах области $\bar{\Omega}_t^{(1)} \setminus \Omega_t$. Третье слагаемое отвечает за поддержание временной стабильности относительно результата восстановления фона в предыдущем кадре в рамках текущего про-

хода с учетом используемой модели движения фона $O^{t \rightarrow t \pm 1}$. Компромисс между соблюдением граничных условий и поддержанием временной стабильности определяется весами w_p^{PB} , где X^{init} – результат независимого применения метода Poisson Blending к текущему кадру (то есть, без учета третьего слагаемого и с $w_p^{PB} \equiv 1$), σ – параметр алгоритма (в авторской реализации $\sigma = 0.002$). Таким образом, граничные условия сменяются на ограничения временной стабильности в областях с высокой ошибкой воспроизведения эталонного поля градиентов G_t , где изменение цвета, требуемое для сокрытия швов, существенно локально непостоянно. В ходе экспериментов было установлено, что данная модификация существенно повышает устойчивость метода к локальным выбросам на отдельных кадрах, обеспечивая высокую временную стабильность. Однако, в отличие от покадрового применения алгоритма Poisson Blending, резко возникающие локальные швы могут сохранять заметность в результате восстановления фона на протяжении нескольких кадров.

Задача минимизации E_{PB} из (1.34) является задачей наименьших квадратов и сводится к решению линейной системы уравнений с разреженной пятидиагональной матрицей для каждого из каналов входной видеопоследовательности. На практике решение таких систем занимает продолжительное время, поэтому для снижения объема вычислений вместо минимизации E_{PB} напрямую предлагается использовать эвристику, которая была изначально предложена в работе [33]. А именно, используя простое копирование согласно V_t для получения первого приближения восстановленного фона X_0 , чередуется независимое построчное решение с игнорированием всех вертикальных зависимостей и независимое решение по столбцам:

$$\begin{aligned} X'_k &= \arg \min_X \left(\varepsilon_k \left(\sum_p \|X(p) - X_{k-1}(p)\|^2 \right) + E_H[X] \right), \\ X_k &= \arg \min_X \left(\varepsilon_k \left(\sum_p \|X(p) - X'_k(p)\|^2 \right) + E_V[X] \right), k = 1, \dots, K \end{aligned} \quad (1.35)$$

Здесь E_H и E_V – это модификации исходного функционала E_{PB} , игнорирующие все вертикальные и горизонтальные зависимости между пикселями реше-

ния, соответственно. Задачи наименьших квадратов (1.35) сводятся к решению линейных систем с трехдиагональной матрицей, для чего может быть использован метод прогонки, обладающий значительно сниженной вычислительной сложностью. $\varepsilon_k = \varepsilon * a^{k-1}$ – экспоненциально возрастающие с номером итерации веса сохранения решения предыдущей итерации (в авторской реализации $\varepsilon = 10^{-4}$, $a = 8$, количество итераций $K = 5$).

Последним шагом, завершающим обработку текущего кадра t , является выполнение присваивания $C(p) := X(p)$ для всех пикселей p с вычисленным значением цвета восстановленного фона $X(p)$ в пределах области Ω_t .

1.4 Анализ вычислительной сложности алгоритма

Предложенный алгоритм восстановления фона в видео состоит из трех проходов по входной последовательности кадров, где обработка каждого кадра в каждом из проходов состоит из:

1. Вычисления отображения в предыдущий кадр $O^{t \rightarrow t \pm 1}$ согласно модели движения фона (раздел 1.3.2). Такое отображение строится путем решения задачи оптимизации (1.13), которая сводится к решению $N_{pyr} \times N_{OF} \times N_{fp}$ СЛАУ, причем количество неизвестных в каждой из них не превосходит $2 \lceil W/s \rceil \lceil H/s \rceil$. Матрица в левой части каждой из этих СЛАУ является разреженной с максимальным числом ненулевых элементов в каждой строке, не зависящим от входных данных (то есть, число ненулевых элементов в каждой строке можно оценить как $O(1)$). Алгоритм вычисления коэффициентов данных СЛАУ имеет временную сложность $O(WH)$. Для решения СЛАУ используется итеративный метод релаксации с фиксированным числом итераций N_{SOR} . Таким образом, временную сложность данного шага алгоритма можно оценить как $O(WH)$.

2. Вычисления начального приближения отображения V_t и проведения его уточнения (раздел 1.3.4). Как уже отмечалось ранее, по построению количество отображений в различные исходные кадры из каждой точки $N(p)$ можно ограничить сверху константой, не зависящей от количества кадров во входной видеопоследовательности и ее ширины/высоты, а именно $N(p) \leq (2d+1)^2$. Поэтому временную сложность алгоритма вычисления начального приближения V_t можно ограничить сверху как $O(WH)$. Затем, для уточнения V_t решается задача оптимизации (1.26), сводящаяся к решению N_{rfp} СЛАУ с $2N_{ref}$ неизвестными, где $N_{ref} \leq (2d+1)^2|\bar{\Omega}_t|$, $|\bar{\Omega}_t|$ – количество пикселей в рамках расширенной области неизвестного фона в текущем кадре $\bar{\Omega}_t$. Так как получаемые СЛАУ снова имеют разреженную матрицу в левой части и для их решения используется итеративный метод сопряженных градиентов с фиксированным числом итераций N_{CG} , то временную сложность алгоритма уточнения V_t также можно ограничить сверху как $O(WH)$.
3. Проведения восстановления фона в текущем кадре и компенсации изменений освещенности (раздел 1.3.5). Результат восстановления фона получается путем решения набора СЛАУ с трехдиагональной матрицей методом прогонки (отдельно для каждой строки и каждого столбца текущего кадра, содержащего восстановленный фон), что обеспечивает оценку временной сложности этого шага в $O(WH)$.

Алгоритм пространственного восстановления фона, применяемый к первому и последнему кадрам, также имеет временную сложность $O(WH)$. Таким образом, временная сложность обработки каждого кадра в каждом из проходов – $O(WH)$, а временную сложность всего алгоритма восстановления фона можно оценить как $O(WHT)$.

Входные и выходные данные алгоритма можно представить в виде четырехканальных видеопоследовательностей (каналы R,G,B и бинарная маска области неизвестного фона). Тогда предложенный алгоритм допускает реализацию, где объем потребляемой памяти не будет зависеть от количества кадров во входном

Алгоритм	Временная сложность	Пространственная сложность
Предложенный	$O(WHT)$	$O(WH)$
Huang et al. [11]	$O(WHT)$	$O(WHT)$
Newson et al. [10]	$O(WHT)$	$O(WHT)$

Таблица 1 — Сравнение вычислительной сложности предложенного алгоритма восстановления фона с аналогами. Приведенные оценки пространственной сложности не включают память, занимаемую входными и выходными данными алгоритма.

видео, если исключить из рассмотрения память, занимаемую входными и выходными данными. То есть, пространственная сложность алгоритма – $O(WH)$. Обработка каждого кадра в каждом из проходов тогда состоит в копировании фрагментов фона из кадров входной видеопоследовательности в кадр выходной последовательности согласно V_t (с компенсацией изменений освещенности) и соответствующей модификации выходной маски области неизвестного фона в текущем кадре. Для вычисления V_t достаточно знать $O^{t \rightarrow t \pm 1}$ и $V_{t \pm 1}$, то есть не требуется хранить в памяти их значения для всех t , а объем памяти, занимаемый V_t , можно оценить как $O(WH)$ благодаря оценке $N(p) \leq (2d + 1)^2$. Число ненулевых коэффициентов в матрицах СЛАУ, возникающих при вычислении $O^{t \rightarrow t \pm 1}$ и уточнении V_t , можно оценить как $O(WH)$, поэтому объем памяти, требуемый для решения задач оптимизации (1.13) и (1.26), также можно оценить как $O(WH)$.

Низкая пространственная сложность является отличительной особенностью предложенного алгоритма по сравнению с другими алгоритмами со схожими характеристиками [10; 11], которые базируются на формулировке задачи восстановления фона в виде единой задачи оптимизации для всего видео с числом неизвестных пропорциональным количеству пикселей в области неизвестного фона во всех кадрах входной видеопоследовательности. Предложенные в данных работах алгоритмы обладают пространственной сложностью $O(WHT)$, то есть объем потребляемой памяти растет прямо пропорционально количеству кадров во входной видеопоследовательности даже если исключить из рассмотрения память, занима-

емую входными и выходными данными. Сравнение вычислительной сложности предложенного алгоритма с аналогами приведено в таблице 1.

1.5 Экспериментальная оценка

Для экспериментальной оценки предложенного алгоритма использовалась методика, описанная во второй главе. А именно, использовался набор из 7 тестовых видеопоследовательностей с эталонными изображениями восстановленного фона, на которых результаты работы различных алгоритмов сравнивались с помощью четырех различных функционалов качества. Сравнение проводилось с алгоритмами восстановления фона, для которых на момент проведения исследования была доступна реализация, в том числе с коммерческими инструментами для восстановления фона и алгоритмами восстановления фона в изображениях:

- Алгоритм восстановления фона в видео, предложенный в работе [10].
- Алгоритм восстановления фона в видео из работы [27], предназначенный для использования в задаче конвертации видео в стереоскопический формат.
- Инструмент F_RigRemoval из программного пакета Nuke [34].
- Инструмент Remove Rig из программного пакета PFClean [35].
- Простой алгоритм восстановления фона в изображениях Telea [30], который применялся независимо к каждому кадру.
- Более сложный алгоритм восстановления фона в изображениях Planar Guidance [36], который также применялся независимо к каждому кадру.

Некоторые алгоритмы не способны полностью восстановить фон во всей области Ω в некоторых тестовых последовательностях; в таких случаях перед проведением сравнения применялся алгоритм [30] независимо к каждому кадру для получения окончательного результата восстановления фона. По результатам экспериментальной оценки предложенный алгоритм превзошел все вышеперечис-

ленные по каждому из функционалов качества восстановленного фона. Результаты данного сравнения приведены в таблицах 2, 3, 4, 5.

Алгоритм	Ср. место	Тестовая последовательность						
		№1	№2	№3	№4	№5	№6	№7
Предложенный	1.9	0.267 ²	0.222 ⁴	0.091 ³	0.038 ¹	0.062 ¹	0.070 ¹	0.039 ¹
Зачесов и др. [27]	2.6	0.221 ¹	0.192 ²	0.217 ⁵	0.070 ²	0.090 ³	0.108 ³	0.083 ²
F_RigRemoval [34]	2.7	0.291 ³	0.211 ³	0.078 ²	0.120 ⁴	0.068 ²	0.091 ²	0.104 ³
Remove Rig [35]	3.4	0.307 ⁵	0.187 ¹	0.077 ¹	0.094 ³	0.143 ⁵	0.163 ⁵	0.106 ⁴
Newson et al. [10]	5.3	0.307 ⁴	0.252 ⁵	0.116 ⁴	0.162 ⁵	0.195 ⁷	0.355 ⁷	0.237 ⁵
Telea [30]	5.7	0.333 ⁷	0.623 ⁷	0.614 ⁶	0.206 ⁶	0.141 ⁴	0.133 ⁴	0.367 ⁶
Planar Guidance [36]	6.4	0.318 ⁶	0.603 ⁶	0.682 ⁷	0.240 ⁷	0.177 ⁶	0.302 ⁶	0.438 ⁷

Таблица 2 — Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества MS-DSSIM, оценивающим близость результата восстановления фона к эталону (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений MS-DSSIM и затем для каждого алгоритма было вычислено среднее место в рейтинге.

Алгоритм	Ср. место	Тестовая последовательность						
		№1	№2	№3	№4	№5	№6	№7
Предложенный	2.0	0.023 ⁶	0.005 ²	0.003 ²	0.003 ¹	0.002 ¹	0.002 ¹	0.003 ¹
Зачесов и др. [27]	2.4	0.013 ¹	0.005 ¹	0.036 ⁵	0.007 ²	0.007 ³	0.009 ³	0.004 ²
Remove Rig [35]	3.6	0.018 ³	0.007 ³	0.003 ¹	0.011 ⁴	0.013 ⁶	0.016 ⁵	0.009 ³
F_RigRemoval [34]	3.7	0.020 ⁵	0.012 ⁵	0.004 ³	0.012 ⁵	0.003 ²	0.006 ²	0.009 ⁴
Newson et al. [10]	4.0	0.018 ⁴	0.009 ⁴	0.008 ⁴	0.011 ³	0.009 ⁴	0.016 ⁴	0.021 ⁵
Telea [30]	5.3	0.013 ²	0.092 ⁶	0.197 ⁶	0.016 ⁶	0.010 ⁵	0.019 ⁶	0.046 ⁶
Planar Guidance [36]	7.0	0.156 ⁷	0.301 ⁷	0.458 ⁷	0.092 ⁷	0.067 ⁷	0.145 ⁷	0.186 ⁷

Таблица 3 — Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества MS-DSSIMdt, оценивающим стабильность результата восстановления фона во времени (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений MS-DSSIMdt и затем для каждого алгоритма было вычислено среднее место в рейтинге.

Алгоритм	Ср. место	Тестовая последовательность						
		№1	№2	№3	№4	№5	№6	№7
Предложенный	1.6	0.127 ³	0.046 ²	0.021 ²	0.020 ¹	0.040 ¹	0.052 ¹	0.020 ¹
Зачесов и др. [27]	2.9	0.118 ¹	0.056 ⁴	0.119 ⁵	0.040 ²	0.055 ³	0.081 ³	0.043 ²
Remove Rig [35]	3.1	0.141 ⁵	0.045 ¹	0.020 ¹	0.056 ³	0.088 ⁴	0.122 ⁵	0.046 ³
F_RigRemoval [34]	3.4	0.140 ⁴	0.075 ⁵	0.027 ³	0.072 ⁴	0.041 ²	0.069 ²	0.050 ⁴
Newson et al. [10]	4.6	0.119 ²	0.053 ³	0.028 ⁴	0.090 ⁵	0.117 ⁶	0.278 ⁷	0.099 ⁵
Telea [30]	5.6	0.183 ⁶	0.286 ⁶	0.313 ⁶	0.130 ⁶	0.092 ⁵	0.102 ⁴	0.212 ⁶
Planar Guidance [36]	6.9	0.200 ⁷	0.293 ⁷	0.409 ⁷	0.158 ⁷	0.118 ⁷	0.227 ⁶	0.288 ⁷

Таблица 4 — Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества $MS-C_{DSSIM}$, оценивающим локальное сходство результата восстановления фона с эталоном (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений $MS-C_{DSSIM}$ и затем для каждого алгоритма было вычислено среднее место в рейтинге.

Алгоритм	Ср. место	Тестовая последовательность						
		№1	№2	№3	№4	№5	№6	№7
Предложенный	1.1	0.015 ²	0.007 ¹	0.004 ¹	0.005 ¹	0.007 ¹	0.005 ¹	0.004 ¹
Зачесов и др. [27]	2.7	0.015 ¹	0.007 ³	0.024 ⁵	0.009 ²	0.009 ³	0.010 ³	0.007 ²
Remove Rig [35]	3.0	0.017 ³	0.007 ²	0.004 ²	0.011 ³	0.015 ⁴	0.013 ⁴	0.009 ³
F_RigRemoval [34]	3.6	0.020 ⁵	0.015 ⁵	0.006 ³	0.014 ⁴	0.008 ²	0.009 ²	0.011 ⁴
Newson et al. [10]	4.7	0.018 ⁴	0.011 ⁴	0.008 ⁴	0.018 ⁵	0.017 ⁵	0.023 ⁶	0.021 ⁵
Telea [30]	5.9	0.023 ⁶	0.065 ⁶	0.096 ⁶	0.027 ⁶	0.019 ⁶	0.016 ⁵	0.049 ⁶
Planar Guidance [36]	7.0	0.071 ⁷	0.105 ⁷	0.128 ⁷	0.060 ⁷	0.039 ⁷	0.081 ⁷	0.091 ⁷

Таблица 5 — Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества $MS-C_{DSSIMdt}$, оценивающим стабильность во времени локального сходства результата восстановления фона с эталоном (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений $MS-C_{DSSIMdt}$ и затем для каждого алгоритма было вычислено среднее место в рейтинге. Стоит отметить, что данный функционал качества продемонстрировал наивысшую корреляцию с экспертной оценкой среди всех функционалов, исследованных во второй главе.

Алгоритм	Предложенный	Huang et al. [11]	Newson et al. [10]
Предложенный	—	52%	91%
Huang et al. [11]	48%	—	90%
Newson et al. [10]	9%	10%	—

Таблица 6 — Результаты экспертного сравнения предложенного алгоритма с аналогами на тестовой выборке из 26 видео. Здесь в каждой ячейке таблицы приводится частота предпочтения результата алгоритма из соответствующей строки результату алгоритма из соответствующего столбца. Например, респонденты указали, что результат предложенного алгоритма превосходит результат алгоритма из работы 2016 года [11] по визуальному качеству в 52% случаев.

1.6 Экспертная оценка

Также было проведено сравнение предложенного алгоритма с аналогами на тестовом наборе из 26 видео, на котором авторы работы 2016 года [11] демонстрировали результаты своего алгоритма. В связи с тем, что данный тестовый набор не содержит эталонные изображения восстановленного фона, сравнение осуществлялось путем экспертной оценки. В данном сравнении приняло участие 63 респондента, которым попарно демонстрировались результаты работы трех различных алгоритмов. Каждому респонденту требовалось сравнить 18 пар результатов, включая 3 скрытых контрольных вопроса, выбрав в каждой паре результат с более высоким визуальным качеством или отметив, что оба результата выглядят приблизительно одинаково. В контрольных вопросах требовалось сравнить пары результатов, где один из них обладает очевидно более высоким качеством; результаты респондентов, которые ошиблись хотя бы в одном контрольном вопросе, исключались из сравнения. Сравнение проводилось в сети Интернет с использованием онлайн-платформы Subjectify.us. Результаты приведены в таблице 6.

Результаты работы предложенного алгоритма на вышеупомянутой тестовой выборке из 26 видео, тестовом наборе из 7 последовательностей, который использовался в предыдущем разделе, и других видеопоследовательностях (всего 40 примеров работы) публично доступны в сети Интернет по адресу http://videocompletion.org/fast_video_completion/.

1.7 Программная реализация

Предложенный алгоритм был реализован в виде консольного приложения на языке C++ с использованием библиотеки OpenCV для чтения, хранения и обработки изображений. Для параллельного решения СЛАУ, возникающих при вычислении оптического потока, был реализован метод релаксации с красно-черным упорядочиванием. Также использовались SSE оптимизации для эффективного использования SIMD инструкций процессора. Для решения СЛАУ, возникающих в задаче уточнения отображения в исходные кадры, использовалась параллельная реализация метода сопряженных градиентов из библиотеки Eigen. Реализация предложенного алгоритма обладает сравнительно высокой скоростью работы. Для тестовой последовательности Camel в разрешении 854×480 из выборки, использованной в экспертной оценке, средняя скорость составила:

- Предложенный алгоритм: **1,2 кадра в секунду** (реализован на языке C++ с использованием библиотек OpenCV и Eigen)
- Huang et al. [11]: **0,008 кадров в секунду** (реализован авторами в среде разработки MATLAB с использованием подключаемых модулей, реализованных на языке C++)
- Newson et al. [10]: **0,007 кадров в секунду** (реализован авторами в среде разработки MATLAB с использованием подключаемых модулей, реализованных на языке C++)

Время работы реализаций алгоритмов замерялось на стационарном компьютере с четырехядерным процессором серии Intel Core i7.

1.8 Выводы по главе 1

В главе был предложен алгоритм восстановления фона в видео, основанный на непараметрической модели движения фона и покадровом уточнении результата восстановления. К основным особенностям можно отнести:

- Модификацию базового алгоритма построения оптического потока, которая позволяет совместно находить оптический поток в области известного фона и интерполировать его внутрь области неизвестного фона путем решения единой задачи оптимизации.
- Покадровое восстановление фона, что позволяет достичь более низкой вычислительной сложности по сравнению с формулировкой задачи как единой задачи оптимизации по всему видео, что является преобладающим подходом среди современных аналогов.
- Алгоритм покадрового восстановления фона с уточнением, позволяющий уменьшить накопление ошибки используемой модели движения фона.

Предложенный алгоритм не уступил аналогам по результатам проведенной экспериментальной и экспертной оценки. В то же время, предложенный алгоритм обладает более низкой пространственной сложностью и авторская реализация алгоритма показывает значительно более высокую скорость обработки видео по сравнению с известными реализациями аналогов.

Результаты главы опубликованы автором в [1; 5]

Глава 2. Методика сравнения алгоритмов восстановления фона в видео

2.1 Введение

Во множестве работ по восстановлению фона в видео [8—11] авторами не проводится какая-либо объективная оценка качества или количественное сравнение результатов различных алгоритмов, что можно связать с отсутствием общепризнанной методики оценки качества в данной области. Большинство авторов ограничиваются публикацией результатов работы предложенного ими алгоритма на нескольких видео, иногда вместе с результатами работы аналогов на тех же видеопоследовательностях. Применение объективных методов оценки качества затруднено даже при наличии эталонных результатов восстановления, так как алгоритмы восстановления фона, как правило, оцениваются по визуальному качеству восстановления с точки зрения наблюдателя, а не по близости к некоторому эталонному результату. Эта проблема приобретает особую актуальность в случаях, когда область неизвестного фона имеет большой размер как в смысле пространственных размеров, так и временной продолжительности (например, в случае бесследного удаления объекта из видео).

В данной главе проводится исследование различных способов объективной оценки качества восстановленного фона и сравнения результатов работы различных алгоритмов. По результатам проведенного исследования предлагается набор функционалов качества восстановленного фона, позволяющих проводить объективное сравнение результатов различных алгоритмов в предположении, что доступен эталонный фон. Эффективность предложенной методики оценки качества была подтверждена следующим образом:

- Построены 7 тестовых видеопоследовательностей, включающих эталонный фон, который является идеальным результатом восстановления (раздел 2.3);

- Проведена автоматическая оценка качества результатов работы 6 алгоритмов восстановления фона на построенных видеопоследовательностях с помощью предложенных функционалов качества (раздел 2.4);
- Проведена экспертная оценка качества путем попарного сравнения результатов различных алгоритмов восстановления фона с участием более 300 респондентов (раздел 2.5);
- Вычислена корреляция между результатами экспертной и автоматической оценок (раздел 2.6)

2.2 Обзор существующих методов сравнения алгоритмов восстановления фона

В данном разделе представлен обзор существующих подходов к оценке качества и получения эталонных результатов восстановления фона в видеопоследовательностях. Рассматриваются методы и наборы данных, которые использовались авторами алгоритмов восстановления фона для проведения объективной оценки качества и сравнения с аналогами.

Количественная оценка качества восстановления часто используется в случаях, когда область неизвестного фона имеет относительно малый размер либо в смысле пространственных размеров, либо временной продолжительности. Это верно, например, для задачи скрытия ошибок, вызванных потерей пакетов в процессе передачи видео по сети [37—39]. Как правило, используются традиционные методы оценки качества видео, такие как PSNR и SSIM [40]. Эталонный результат восстановления здесь оказывается доступен вследствие того, что ошибки, эмулирующие эффект потери пакетов разной интенсивности, вносятся в исходный видеопоток искусственным образом. Авторы алгоритма, предложенного в работе [21], проводят его апробацию в ряде различных сценариев применения, однако количественная оценка качества работы проводится только для сценария скрытия

ошибок, вызванных потерей пакетов при передаче видеопотока. Аналогичным образом оценка качества производится для задач автоматического устранения логотипов [41; 42] и удаления текста [43] из видеопоследовательности. Авторы работы [22] используют меру RMSE (Root-Mean-Square Error – корень из среднеквадратичной ошибки) для оценки качества восстановления достаточно крупных областей, но имеющих продолжительность не более 5 кадров. В некоторых тестовых последовательностях целые кадры включались в область неизвестного фона, что, фактически, сводит задачу к межкадровой интерполяции. Предложенный алгоритм восстановления был сравнен с одним альтернативным подходом на 10 последовательностях низкого разрешения (порядка 352×240) продолжительностью от 35 до 100 кадров.

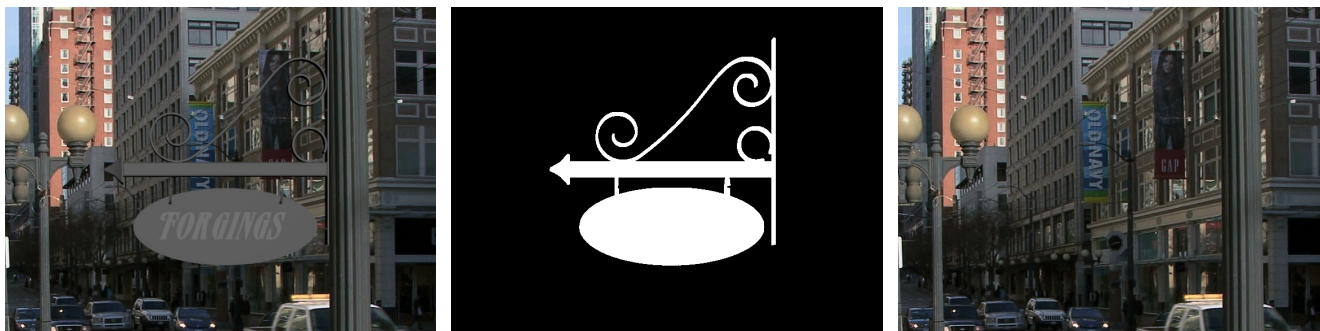
В работе [44] для построения тестовых последовательностей на каждом кадре исходной видеопоследовательности вырезается случайным образом выбранный блок. Предложенный алгоритм сравнивается с пятью аналогами. Авторы работ [45] и [46] проводят сравнение на единственной последовательности, где небольшой блок вырезается из одной и той же позиции на каждом кадре. Для оценки качества используются PSNR и среднеквадратичная ошибка. В работе [47] используется простая сумма модулей разности (SAD) между результатом восстановления и оригинальной видеопоследовательностью для демонстрации высокой точности результата предложенного алгоритма. Приводится сравнение с двумя аналогами на одной тестовой последовательности и трех отдельных областях неизвестного фона, которые характеризуются различным характером движения (быстрое движение, медленное движение и его полное отсутствие). Однако стоит отметить, что рассматриваемые авторами области неизвестного фона являются сравнительно крупными как в пространственном, так и во временном направлениях, но никакого обоснования используемой методики оценки качества не приводится. Авторы работы [48] проводят подсчет индекса сходства SSIM на одной из тестовых последовательностей, однако приведенное сравнение с аналогами имеет качественный, а не количественный характер. В обзоре алгоритмов восстановления фона 2014 года [49] отмечается, что авторам неизвестны работы, которые бы

рассматривали проблему количественной оценки качества восстановления фона в видео в общей постановке.

По результатам проведенного обзора можно сделать вывод, что проблема количественной оценки качества восстановления крупных пространственно-временных областей в видеопоследовательностях явным образом не рассматривалась в предшествующих работах. Классические методы оценки качества, такие как PSNR и SSIM, хорошо подходят для оценки качества восстановления областей, которые либо являются небольшими в смысле пространственных размеров, либо имеют малую продолжительность, не превышающую нескольких кадров. Однако эти методы становятся значительно менее надежными для областей неизвестного фона, которые являются крупными как в смысле пространственных размеров, так и временной продолжительности, так как соответствие результата восстановления некоторому единому эталону уже не обязательно для достижения высокой экспертной оценки качества.

2.3 Построение тестового набора данных

Для количественного сравнения различных алгоритмов, в первую очередь, требуется определить набор данных, на котором оно будет проводиться. При построении тестовых видеопоследовательностей принимался во внимание ряд принципов. Во-первых, каждая тестовая видеопоследовательность должна представлять сложность для существующих алгоритмов, иначе невозможно надежно ранжировать алгоритмы по качеству работы, так как их результаты будут близки к одинаковым. Также предпринимается попытка охватить как можно больше практических случаев восстановления фона, включая последовательности со статичной и свободно перемещающейся камерой, статический и динамический фон, динамические текстуры в видео (например, поверхность воды, огонь). Области неизвестного фона в некоторых из построенных видеопоследовательностей вклю-



(а) Входной кадр (б) Входная маска (в) Эталонный фон

Рисунок 2.1 — Пример из составленной тестовой выборки, включающий фрагмент входного кадра видео (а), соответствующую маску области неизвестного фона (б), и эталонное изображение фона (в).

чают фрагменты фона, не видимые ни в одном из кадров входного видео, что представляет дополнительную сложность для существующих алгоритмов восстановления фона. И, наконец, все предлагаемые тестовые видеопоследовательности имеют разрешение 1920×1080 и продолжительность от 150 до 200 кадров. Это важно для выделения более практичных подходов, так как многие алгоритмы восстановления фона в видео имеют серьезные ограничения в плане обработки продолжительных видеопоследовательностей высокого разрешения из-за низкой скорости работы и/или слишком высокого потребления памяти.

Все построенные тестовые видеопоследовательности относятся к одному приложению алгоритмов восстановления фона, а именно бесследному удалению объектов из видео. Для получения видеопоследовательностей с эталонным восстановленным фоном проводится искусственное наложение различных объектов переднего плана на фоновые видеопоследовательности средствами компьютерной графики. Для максимально правдоподобной интеграции объектов переднего плана в фоновое видео с учетом движения камеры используется программный пакет Blender [50]. Пример построенного видео с маской области неизвестного фона и эталонным результатом восстановления приведен на рисунке 2.1.

Таким образом, каждый алгоритм восстановления фона получает на вход видеопоследовательность с наложенным объектом переднего плана и соответствующую маску области неизвестного фона. Затем проводится оценка качества

результата восстановления с учетом исходной видеопоследовательности, содержащей эталонные изображения фона за наложенным объектом. Всего было построено 7 тестовых видеопоследовательностей для сравнения качества различных алгоритмов восстановления фона. Построенная тестовая выборка доступна в сети Интернет по адресу <http://videocompletion.org/>.

2.4 Функционалы качества восстановленного фона

Предположим, что известны эталонные значения яркости фона $C_{GT} : I \mapsto R$ (в проведенных экспериментах учет цветовых компонент оказал малое влияние на результат, поэтому рассматривается только яркость). Как и в первой главе, $I = [1, W] \times [1, H] \times \{1, \dots, T\}$ – область определения видео, $C : I \mapsto R$ – результат восстановления фона тем или иным алгоритмом. Требуется построить функционал качества $\mathcal{F}[C, C_{GT}] = q \in \mathbb{R}$, согласованный с экспертной оценкой. Для определенности положим, что чем меньше значение такого функционала, тем выше должно быть визуальное качество результата.

Адаптируем традиционные методы оценки качества видео для задачи восстановления фона с учетом области неизвестного фона $\Omega \subset I$:

$$\begin{aligned} \text{MSE}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} \text{MSE}[C(P), C_{GT}(P)], \\ \text{DSSIM}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} 1 - \text{SSIM}[C(P), C_{GT}(P)]. \end{aligned} \tag{2.1}$$

Здесь P обозначает квадратный блок размера $s \times s \times 1$ (то есть, двумерный блок с пространственным размером $s \times s$ пикселей и продолжительностью в 1 кадр). Во всех определенных в данном разделе функционалах, если явно не указано обратное, используются двумерные блоки с параметром $s = 9$. MSE обозначает среднеквадратичное отклонение результата восстановления от эталона в рамках соответствующего блока. Индекс структурного сходства SSIM между блоками опре-

деляется согласно работе [40]. Отметим, что по построению для любых блоков $P, Q \subset I$ верно: $-1 \leq \text{SSIM}[C(P), C_{GT}(Q)] \leq 1$. Значение индекса 1 достигается лишь в случае $C(P) \equiv C_{GT}(Q)$, а чем меньше его значение, тем больше заметна разница между C и C_{GT} , определенных на P и Q соответственно, с точки зрения наблюдателя. Наконец, N – количество всевозможных (возможно перекрывающихся друг с другом) блоков P фиксированного размера, пересекающиеся хотя бы в одном пикселе с Ω , то есть, количество слагаемых в последующей сумме.

Традиционные методы оценки качества видео (2.1) в контексте задачи оценки качества восстановленного фона обладают рядом недостатков. Далее эти недостатки будут рассмотрены и будут предложены возможные способы их устранения. Во-первых, определенные выше функционалы MSE, DSSIM имеют высокую чувствительность к пространственным сдвигам результата восстановления фона относительно эталона, хотя на визуальное качество подобные сдвиги оказывают малое влияние. Одним из простых способов частичного устранения данного недостатка является проведение анализа сразу на нескольких масштабах:

$$\begin{aligned} \text{MS-MSE}[C, C_{GT}] &= \sum_{i=0}^{M-1} w_i^{\text{MSE}} \text{MSE}[C^i, C_{GT}^i], \\ \text{MS-DSSIM}[C, C_{GT}] &= \sum_{i=0}^{M-1} w_i^{\text{DSSIM}} \text{DSSIM}[C^i, C_{GT}^i]. \end{aligned} \quad (2.2)$$

Верхний индекс в C^i и C_{GT}^i обозначает текущий уровень Гауссовой пирамиды (каждый кадр уменьшается по обоим пространственным размерностям в 2^i раз на i -ом уровне пирамиды). M – количество уровней пирамиды, w_i – веса соответствующих уровней. Присвоение больших весов более высоким уровням пирамиды увеличивает устойчивость метода к небольшим сдвигам между результатом восстановления и эталоном. Значения весов w_i вычисляются на основе проведенной экспертной оценки различных алгоритмов восстановления фона (см. раздел 2.6).

Зрительная система человека имеет повышенную чувствительность к нестабильным во времени искажениям, однако рассмотренные до сих пор методы не пригодны для измерения такого вида искажений. Для устранения этой

проблемы предлагаются следующие функционалы, явным образом учитывающие стабильность во времени:

$$\begin{aligned} \text{MSEdt}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{PCI \\ P \cap \Omega \neq \emptyset}} \max (\text{MSE}[C(P), C(P_{\text{prev}})] - \\ &\quad \text{MSE}[C_{GT}(P), C_{GT}(P_{\text{prev}})], 0), \\ \text{DSSIMdt}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{PCI \\ P \cap \Omega \neq \emptyset}} \max (\text{DSSIM}[C_{GT}(P), C_{GT}(P_{\text{prev}})] - \\ &\quad \text{DSSIM}[C(P), C(P_{\text{prev}})], 0), \end{aligned} \quad (2.3)$$

Здесь P_{prev} обозначает результат смещения блока P согласно вектору движения от текущего к предыдущему кадру в центре P в рамках эталонной видеопоследовательности. Было опробовано несколько алгоритмов построения оптического потока для вычисления векторов движения в эталонной видеопоследовательности, однако наилучший результат был получен путем использования алгоритма PatchMatch [15] для нахождения наиболее схожего блока P_{prev} на предыдущем кадре с радиусом поиска равным $1/20$ ширины кадра; сходство здесь оценивается с помощью MSE. Определенные в (2.3) функционалы измеряют нестабильность результата восстановления фона вдоль векторов движения эталона. Однако это неявно предполагает полную выравненность результата восстановления относительно эталона. Для ослабления этого требования вводится одновременный учет нескольких масштабов аналогично MS-MSE и MS-DSSIM:

$$\begin{aligned} \text{MS-MSEdt}[C, C_{GT}] &= \sum_{i=0}^{M-1} w_i^{\text{MSEdt}} \text{MSEdt}[C^i, C_{GT}^i], \\ \text{MS-DSSIMdt}[C, C_{GT}] &= \sum_{i=0}^{M-1} w_i^{\text{DSSIMdt}} \text{DSSIMdt}[C^i, C_{GT}^i]. \end{aligned} \quad (2.4)$$

В целом, устойчивость рассмотренных выше функционалов качества к сдвигам между результатом восстановления и эталоном ограничивается лишь пространственными сдвигами небольшой величины. Альтернативным подходом является оценка локального сходства результата восстановления фона с эталоном. То есть, путем нахождения для каждого блока восстановленного фона P соответствующего блока Q эталона (возможно, на другом кадре), который максимизиру-

ет сходство $C(P)$ и $C_{GT}(Q)$, можно значительно повысить устойчивость соответствующих функционалов качества к сдвигам:

$$\begin{aligned} C_{\text{MSE}}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} \min_{Q \subset I} (\text{MSE}[C(P), C_{GT}(Q)]), \\ C_{\text{DSSIM}}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} \min_{Q \subset I} (1 - \text{SSIM}[C(P), C_{GT}(Q)]). \end{aligned} \quad (2.5)$$

Как упоминалось в разделе 1.2, существует класс алгоритмов восстановления фона [8; 10; 16], основанных максимизации локального сходства результата восстановления в области Ω с известным фоном в $I \setminus \Omega$ с использованием трехмерных пространственно-временных блоков пикселей. В отличие от алгоритмов этого класса, функционалы качества из (2.5) оценивают локальное сходство с эталоном в пределах всей области определения I и используют двумерные блоки. Согласно [10] определим также аналоги (2.5) с использованием трехмерных блоков P^{3D} , Q^{3D} размера $5 \times 5 \times 5$ пикселей:

$$\begin{aligned} C_{\text{MSE}}^{3D}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P^{3D} \subset I \\ P^{3D} \cap \Omega \neq \emptyset}} \min_{Q^{3D} \subset I} (\text{MSE}[C(P^{3D}), C_{GT}(Q^{3D})]), \\ C_{\text{T-MSE}}^{3D}[C, C_{GT}] &= \frac{1}{N} \sum_{\substack{P^{3D} \subset I \\ P^{3D} \cap \Omega \neq \emptyset}} \min_{Q^{3D} \subset I} (\text{T-MSE}[C(P^{3D}), C_{GT}(Q^{3D})]). \end{aligned} \quad (2.6)$$

Использование трехмерных пространственно-временных блоков также позволяет косвенно оценивать сходство характера движения сравниваемых областей видеопоследовательностей. T-MSE – предложенная авторами [10] модификация MSE для измерения сходства пространственно-временных блоков, которая позволила улучшить визуальное качество восстановленного фона в рамках предложенного ими алгоритма:

$$\text{T-MSE}[C(P^{3D}), C_{GT}(Q^{3D})] = \frac{1}{n} (\|C(P^{3D}) - C_{GT}(Q^{3D})\|_2^2 + \lambda \|T(P^{3D}) - T_{GT}(Q^{3D})\|_2^2). \quad (2.7)$$

Здесь $T(P^{3D})$, $T_{GT}(Q^{3D})$ – значения дополнительных текстурных признаков, предварительно вычисленных в каждом пикселе результата восстановления фона и эталона.

По аналогии с (2.2) и (2.4) определим также функционалы $MS-C_{MSE}$, $MS-C_{DSSIM}$, $MS-C_{MSE}^{3D}$, $MS-C_{T-MSE}^{3D}$. Совмещение результатов на разных масштабах в функционалах, основанных на измерении локального сходства, позволяет отдельно учитывать искажения разного масштаба, к которым зрительная система человека имеет различную чувствительность [51].

В то время как функционалы (2.6) позволяют учитывать сходство движения между результатом восстановления и эталоном за счет использования трехмерных блоков, функционалы (2.5) учитывают лишь пространственные искажения в отдельных кадрах. Оценим также временную стабильность локального сходства результата восстановления с эталоном путем измерения того, насколько расстояние до наиболее схожего двумерного блока Q в эталонном видео изменяется от кадра к кадру. Точнее, для каждого блока в текущем кадре результата восстановления фона найдем наиболее схожий блок на предыдущем кадре в рамках окна с ограниченным радиусом поиска, и затем сравним расстояния до наиболее схожего блока эталона между ними:

$$C_{MSEdt}[C, C_{GT}] = \frac{1}{N} \left| \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} \min_{Q \subset I} (\text{MSE}[C(P), C_{GT}(Q)]) - \min_{Q \subset I} (\text{MSE}[C(P_{\text{prev}}^{\text{MSE}}), C_{GT}(Q)]) \right|, \quad (2.8)$$

$$P_{\text{prev}}^{\text{MSE}} = \arg \min_{Q \subset \Omega_{\text{prev}}^{w \times w}(P)} \text{MSE}[C(P), C(Q)].$$

$$C_{DSSIMdt}[C, C_{GT}] = \frac{1}{N} \left| \sum_{\substack{P \subset I \\ P \cap \Omega \neq \emptyset}} \min_{Q \subset I} (1 - \text{SSIM}[C(P), C_{GT}(Q)]) - \min_{Q \subset I} (1 - \text{SSIM}[C(P_{\text{prev}}^{\text{SSIM}}), C_{GT}(Q)]) \right|, \quad (2.9)$$

$$P_{\text{prev}}^{\text{SSIM}} = \arg \max_{Q \subset \Omega_{\text{prev}}^{w \times w}(P)} \text{SSIM}[C(P), C(Q)].$$

Здесь $\Omega_{\text{prev}}^{w \times w}(P)$ обозначает окрестность размера $w \times w$ пикселей вокруг P , но на предыдущем относительно него кадре. В авторской реализации используется w равное $1/10$ ширины кадра. Точно так же, как и ранее, определим функционалы $MS-C_{MSEdt}$, $MS-C_{DSSIMdt}$, совмещающие результаты с разных масштабов. В отличие от функционала $MS-MSEdt$, который позволяет оценивать стабильность

результата восстановления вдоль векторов движения эталонного видео (это предполагает необходимость полного соответствия движения в результате восстановления и движения в эталоне, что во многих случаях может быть слишком сильным предположением), функционалы (2.8) и (2.9) измеряют то, насколько искажения блоков результата восстановления варьируются от кадра к кадру.

Точное вычисление функционалов, основанных на оценке локального сходства, затруднено чрезмерно высокой вычислительной сложностью полного перебора всех блоков эталонной видеопоследовательности. Значительно более эффективным образом можно вычислить приблизительные значения функционалов с помощью алгоритма PatchMatch [15]. В случае использования трехмерных блоков данный алгоритм применяется в полной аналогии с работой [10]. Для нахождения наиболее схожих двумерных блоков алгоритм PatchMatch применяется в покадровом режиме, результат предыдущего кадра используется как начальное приближение в текущем для ускорения сходимости. Также, для сравнения блоков всегда используется метрика SSD (сумма квадратов разностей) в цветовом пространстве RGB. Явная максимизация индекса сходства SSIM в проведенных экспериментах не привела к существенному повышению корреляции с результатами экспертной оценки, в то время как ее использование многократно увеличивает вычислительную сложность алгоритма вычисления соответствующих функционалов качества.

2.5 Экспертная оценка

На момент проведения исследования авторы лишь нескольких работ по восстановлению фона в видео предоставили реализации предложенных алгоритмов. Поэтому для увеличения количества данных для проведения экспертной оценки качества восстановленного фона и измерения корреляции различных функционалов качества с экспертной оценкой также была проведена оценка коммерческих

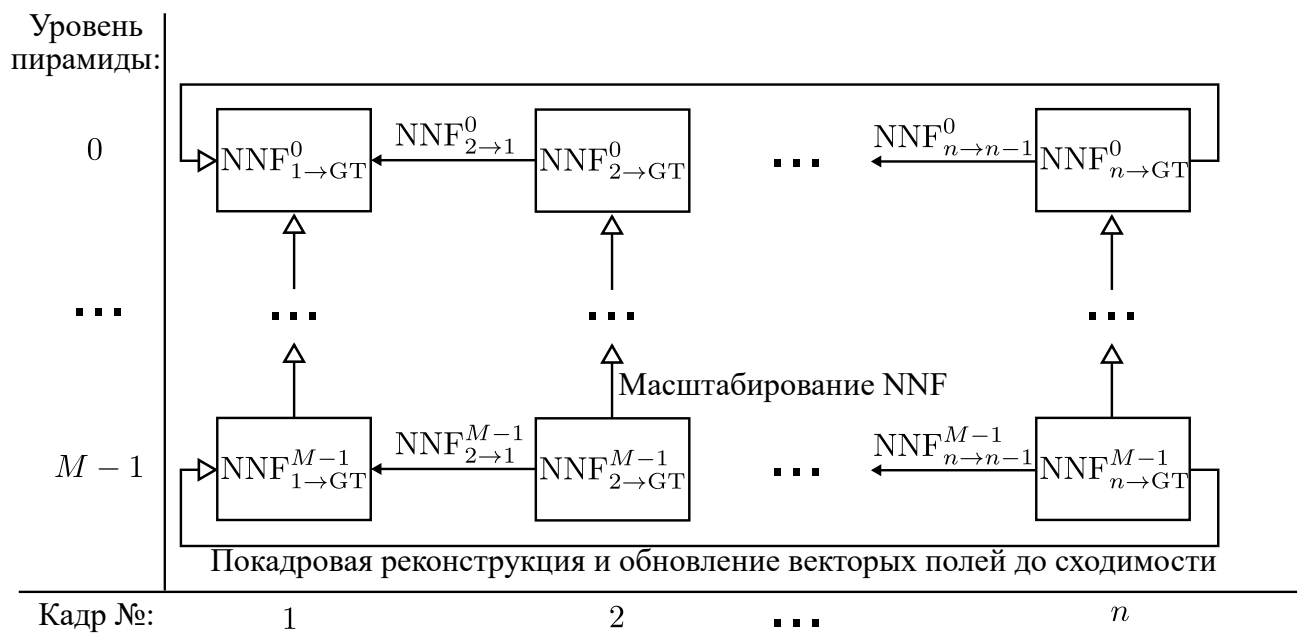


Рисунок 2.2 — Общая схема простого жадного алгоритма, используемого для совместной минимизации функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ на базе предоставленного начального приближения. В ходе алгоритма итеративно вычисляются векторные поля $NNF_{t \rightarrow t-1}$, сопоставляющие блоки соседних кадров, и векторные поля $NNF_{t \rightarrow GT}$, сопоставляющие каждому блоку текущего кадра наиболее схожий блок в эталонном видео, с помощью алгоритма PatchMatch [15]. На основе вычисленных векторных полей проводится покадровая реконструкция, то есть обновление значений в области неизвестного фона с помощью (2.10). После достижения сходимости происходит переход на следующий уровень Гауссовой пирамиды, где процесс повторяется.

инструментов восстановления фона в видео и нескольких алгоритмов восстановления фона в изображениях. Итого в сравнении приняло участие 6 алгоритмов:

- Video Inpainting of Complex Scenes [10];
- Алгоритм BGR, описанный в работе [27];
- Инструмент F_RigRemoval из программного пакета Nuke [34].
- Инструмент Remove Rig из программного пакета PFClean [35].
- Простой алгоритм восстановления фона в изображениях Telea Inpainting [30].
- Более сложный алгоритм восстановления фона в изображениях Image Completion using Planar Structure Guidance [36].

Для дополнительного расширения используемого набора данных также был добавлен ряд синтетических результатов, полученных путем прямой совместной минимизации функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ с использованием эталонного фона. Это является дополнительным способом проверки того, что более низкие значения $MS-C_{MSE}$ и $MS-C_{MSEdt}$ соответствуют более высокому визуальному качеству. Это не является очевидным, так как подход, на котором основаны эти функционалы качества, явным образом не оценивает сходство движения между результатом восстановления и эталоном. Для минимизации используется модификация алгоритма реконструкции, предложенного в работе [10]. Общая схема алгоритма представлена на рисунке 2.2. Основная модификация заключается в использовании двух типов векторных полей в процессе реконструкции – поля $NNF_{t \rightarrow t-1}$, которое сопоставляет каждому блоку текущего кадра наиболее схожий блок в предыдущем кадре в пределах результата реконструкции, и поля $NNF_{t \rightarrow GT}$, которое сопоставляет каждому блоку текущего кадра результата реконструкции наиболее схожий блок во всем эталонном видео. Тогда одна итерация реконструкции пиксела p на текущем кадре результата восстановления C имеет следующий вид:

$$C(p) := \frac{(1 - \tau) \sum_{q \in P(p)} w^s(q) C_{GT}(p + NNF_{t \rightarrow GT}(q)) + \tau \sum_{q \in P(p)} w^t(q) C(p + NNF_{t \rightarrow t-1}(q))}{(1 - \tau) \sum_{q \in P(p)} w^s(q) + \tau \sum_{q \in P(p)} w^t(q)}. \quad (2.10)$$

Здесь τ – это константа, обозначающая вес темпоральной компоненты, которая использует результат реконструкции предыдущего кадра для реконструкции текущего (в авторской реализации $\tau = 0.4$). $P(p)$ – двумерный блок пикселей размера 9×9 с центром в p . $w^s(q)$ и $w^t(q)$ обозначают весовые функции, которые присваивают большие веса тем векторам полей $NNF_{t \rightarrow GT}$ и $NNF_{t \rightarrow t-1}$, соответственно, которые соединяют более похожие блоки (с меньшим значением метрики SSD между блоками). Используются экспоненциальные весовые функции по аналогии с работой [10].

Такой подход позволяет находить локальные оптимумы, которые зависят от начального приближения. Результаты всех 6 алгоритмов восстановления фона

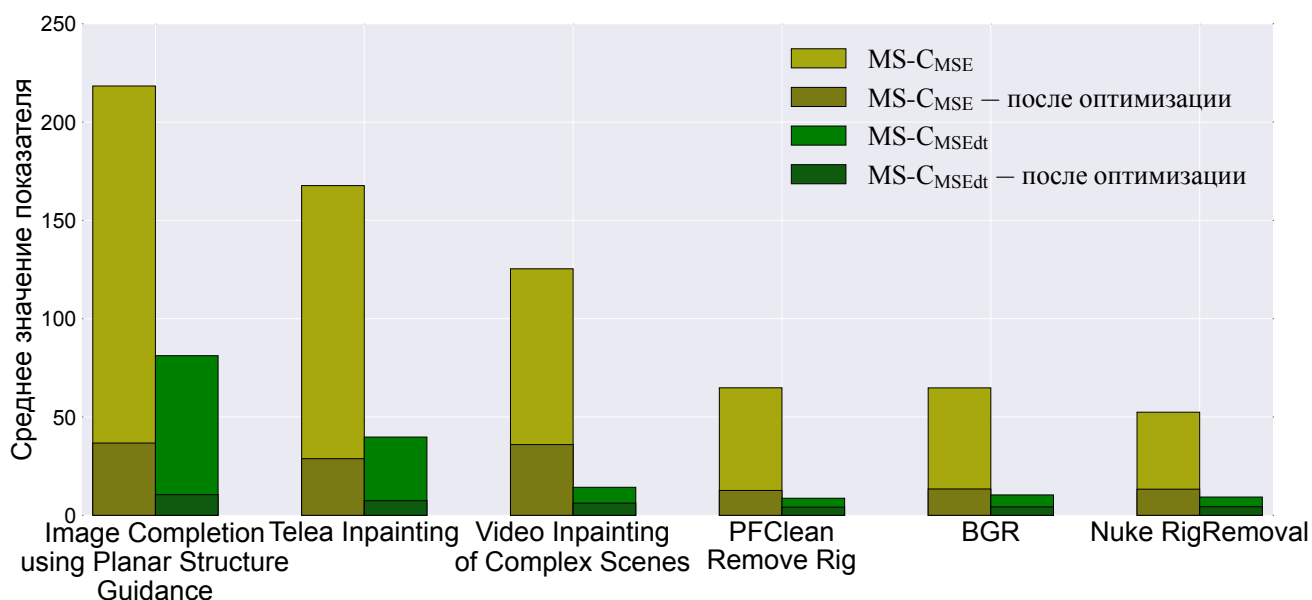


Рисунок 2.3 — Значения функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ до и после применения алгоритма их минимизации к результатам различных алгоритмов, усредненные по всем тестовым видеопоследовательностям. Весовые коэффициенты для разных уровней Гауссовой пирамиды в функционалах были выбраны в соответствии с описанием в разделе 2.6.

использовались в качестве начальных приближений: во всех случаях предложенный подход позволил существенно уменьшить значения обоих функционалов (см. рисунок 2.3). Добавление таким образом полученных синтетических результатов удваивает количество данных для проведения экспертной оценки.

Для количественной оценки визуального качества было проведено исследование, в рамках которого участники попарно сравнивали результаты различных алгоритмов и выбирали результат с наилучшим, по их мнению, качеством. Каждый участник сравнил 28 пар результатов восстановления фона, которые включали 3 скрытых контрольных вопроса, где требовалось сравнить эталонный фон с результатом заведомо низкого качества. Для успешного прохождения требовалось правильно ответить на все контрольные вопросы. Общее количество пар, требующих сравнения, составило 2964. Это число составлено из 6 оригинальных алгоритмов, 6 соответствующих синтетических алгоритмов, состоящих в применении вышеописанной процедуры совместной минимизации функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ к результатам оригинальных алгоритмов, и 19 видеопоследователь-



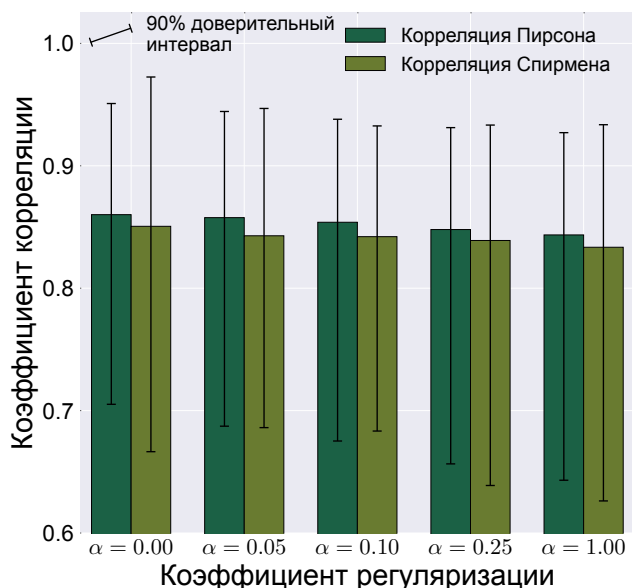
Рисунок 2.4 — Общее ранжирование алгоритмов восстановления фона в видео по визуальному качеству, полученное в результате экспертной оценки.

ностей, на которых проводилось их сравнение (использовались различные фрагменты фиксированного размера из исходных 7 видео). Итого было собрано 8533 попарных сравнений от 341 респондента, которые затем были преобразованы в субъективные ранги с помощью модели Тёрстоуна [52] как для каждой последовательности по отдельности, так и для всех последовательностей вместе (см. рисунок 2.4).

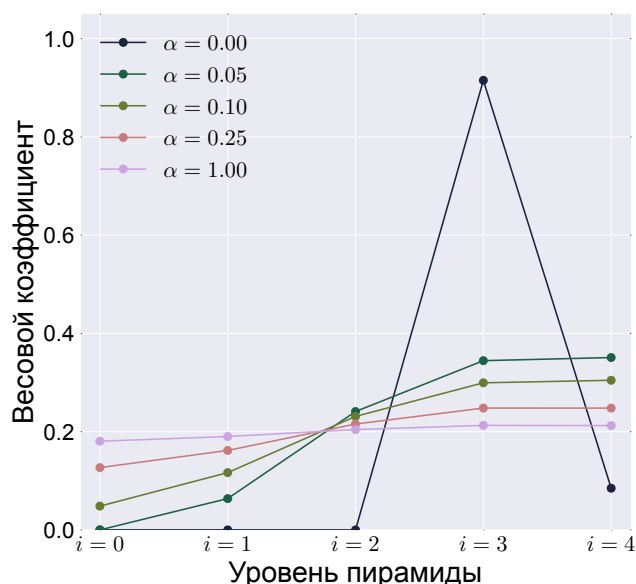
2.6 Выбор функционала качества

Для получения оптимальных значений весов в функционалах, включающих анализ на нескольких уровнях Гауссовой пирамиды, проводится максимизация корреляции с субъективными рангами, полученными в результате экспертной оценки. Сумма весов фиксируется равной единице, и используется L_2 регуляризация:

$$\mathbf{w} = \arg \max_{\substack{\mathbf{w}=[w_0, \dots, w_4] \\ \sum w_i=1}} \sum_{s=1}^S \text{corr}(-\log(\mathbf{w}\mathbf{M}_s), \mathbf{r}_s) - \alpha \|\mathbf{w}\|_2^2. \quad (2.11)$$



(а) Корреляция функционала MS-DSSIM с экспертной оценкой



(б) Распределение весов между разными уровнями Гауссовой пирамиды (масштабами) для функционала MS-DSSIM

Рисунок 2.5 — Иллюстрация эффекта регуляризации при выборе оптимальных весов в ходе максимизации корреляции с экспертной оценкой.

Здесь $\mathbf{M}_s = \{m_{ij}^s\}$, m_{ij}^s — значение функционала для i -ого уровня пирамиды, j -ого алгоритма и s -ой видеопоследовательности (S — общее количество последовательностей), r_s — соответствующий набор субъективных рангов различных алгоритмов на s -ой последовательности. Эффект регуляризации проиллюстрирован на рисунке 2.5. Она повышает надежность соответствующих функционалов за счет покрытия большего числа масштабов, но ценой небольшой потери в значениях корреляции с субъективными рангами. Более корректным способом выбора весов является явное использование синтетических примеров с искажениями на различных масштабах, как было предложено авторами метода MS-SSIM [51]. Однако такой подход является слишком трудозатратным в рамках рассматриваемой задачи, поэтому итоговые веса вычисляются путем решения задачи (2.11) с коэффициентом регуляризации $\alpha = 0.1$. Итоговое сравнение различных функционалов качества восстановления фона по корреляции с экспертной оценкой приведено на рисунке 2.6.

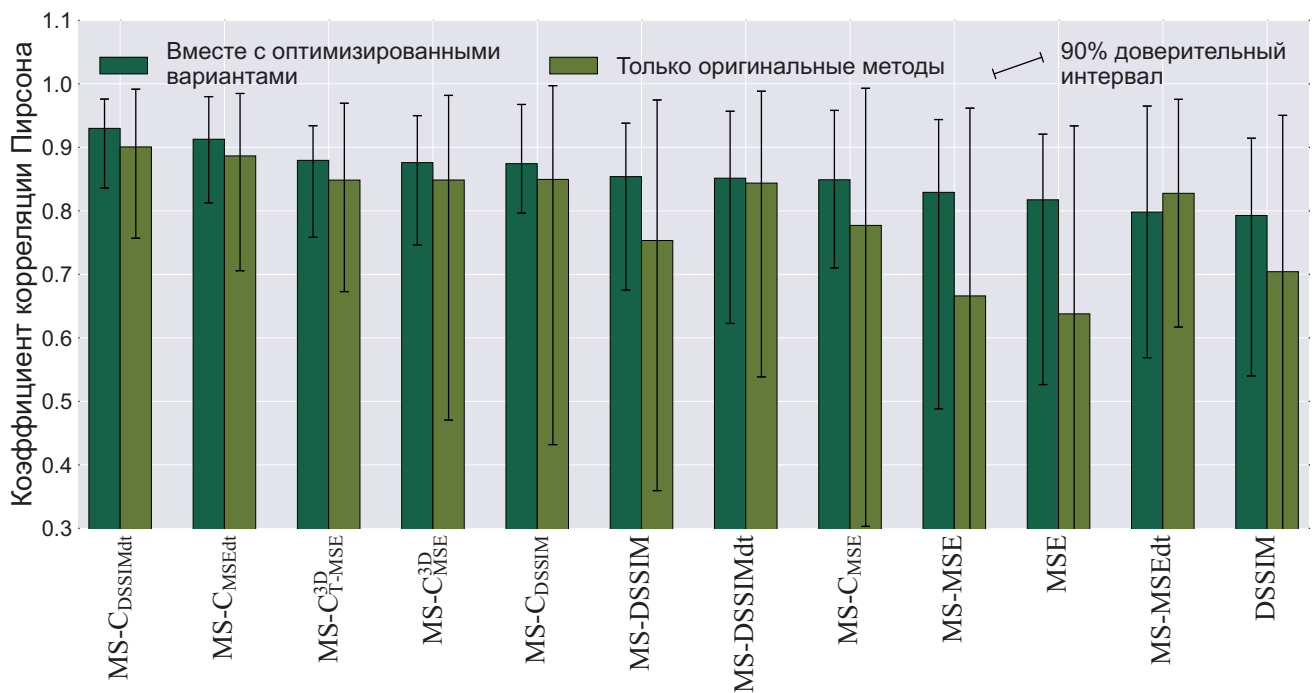


Рисунок 2.6 — Итоговые значения корреляции рассмотренных функционалов качества восстановленного фона с экспертной оценкой. Корреляция вычислялась на двух различных наборах данных: полном наборе, включающем синтетические результаты, полученные путем процедуры оптимизации (см. раздел 2.5), и наборе, включающем только результаты б оригинальных алгоритмов восстановления фона.

В первую очередь, стоит отметить, что практически все функционалы качества показывают худший результат на наборе данных, включающем только б оригинальных алгоритмов, особенно функционалы, которые базируются на прямом сравнении с эталоном (MSE, DSSIM, MS-MSE, MS-DSSIM). Это можно объяснить тем, что используемая процедура оптимизации уменьшает различия между результатом восстановления и эталоном (что естественно, так как эталонный результат в процессе оптимизации напрямую используется) и в то же время увеличивает экспертную оценку (см. рисунок 2.4). Сочетание этих двух факторов приводит к повышению корреляции таких простых функционалов качества как MSE и DSSIM на полном наборе данных. Независимо от используемого набора данных подходы, которые базируются на индексе структурного сходства (SSIM), стабильно показывают лучший результат по сравнению с подходами, использующими среднеквадратичную ошибку (MSE). Также, оценки временной стабильно-

сти, как правило, показывают более высокую корреляцию с экспертной оценкой по сравнению с оценками пространственной ошибки, что и следовало ожидать. С другой стороны, функционалы $MS-C_{MSEdt}$ и $MS-C_{DSSIMdt}$ показывают на удивление высокую корреляцию, учитывая, что явным образом сходство движения между результатом восстановления и эталоном в них не используется. Оценки, использующие трехмерные блоки, также показывают неплохую корреляцию с экспертной оценкой, но алгоритм их вычисления обладает более высокой вычислительной сложностью по сравнению с оценками, основанными на использовании двумерных блоков. Таким образом, следующие функционалы были выбраны для объективной оценки качества восстановления фона в видео в рамках предложенной методики:

- $MS-C_{DSSIMdt}, \mathbf{w}^{CDSSIMdt} = [0.00, 0.08, 0.25, 0.30, 0.37]$
- $MS-C_{DSSIM}, \mathbf{w}^{CDSSIM} = [0.04, 0.11, 0.21, 0.29, 0.35]$
- $MS-DSSIMdt, \mathbf{w}^{DSSIMdt} = [0.00, 0.00, 0.30, 0.32, 0.38]$
- $MS-DSSIM, \mathbf{w}^{DSSIM} = [0.05, 0.12, 0.23, 0.30, 0.30]$

2.7 Выводы по главе 2

В главе была предложена методика сравнения алгоритмов восстановления фона в видео, которая состоит из:

- Тестовой выборки, включающей 7 видеопоследовательностей с доступными эталонными изображениями фона.
- Набора из 4 функционалов качества восстановленного фона, включающего различные оценки пространственных искажений и временной стабильности результата того или иного алгоритма относительно эталонного фона.

В рамках данной методики было предложено два новых функционала качества восстановленного фона, оценивающих локальное сходство с эталоном и

временную стабильность локального сходства, которые превзошли ранее известные функционалы по корреляции с экспертной оценкой. Разработанная методика послужила основой для открытого для новых участников сравнения алгоритмов восстановления фона, доступного в сети Интернет по адресу <http://videocompletion.org/>.

Результаты главы опубликованы автором в [2; 3].

Глава 3. Алгоритм построения стереоскопических ракурсов с помощью многослойного восстановления фона

3.1 Постановка задачи

Для преобразования видео в стереоскопический формат требуется построить левый и правый ракурс выходной стереоскопической видеопоследовательности на базе одного входного ракурса. Рассмотрим задачу построения нового стереоскопического ракурса по одному входному. Трансформация исходного ракурса в целевой может описана с помощью отображения $S(p) = p + (\mathcal{D}(p), 0, 0)$, где $p = (x, y, t) \in I$, а функция горизонтального смещения $\mathcal{D} : I \mapsto \mathbb{R}$ называется диспаратностью соответствующих точек исходного ракурса. Диспаратность может быть однозначным образом вычислена из значения глубины точки p (расстояния от плоскости линзы камеры), фокусного расстояния камеры и значения стереобазы создаваемого стереоскопического видео (расстояния между оптическими осями виртуальных камер, снимающих левый и правый ракурс). Задача построения карты глубины, ставящей в соответствие каждой точке кадра ее значение глубины, является отдельной задачей, рассмотрение которой выходит за рамки данной работы. Для части точек целевого ракурса q можно найти p такое, что $S(p) = q$; в таком случае цвет целевого ракурса в q может быть определен как цвет точки исходного ракурса с наименьшим значением глубины среди тех, которые отображаются в q отображением S . На практике для получения высококачественного результата часто также требуется моделировать эффект полупрозрачности; в таком случае цвет целевого ракурса в q можно выразить как смесь цветов точек исходного ракурса, отображаемых в q . Вопрос моделирования полупрозрачности при построении стереоскопических ракурсов подробно рассмотрен в работе [53] и также выходит за рамки рассмотрения данной работы. Важно отметить, что для точек в области $\{q | \nexists p : S(p) = q\}$ невозможно вычислить цвет целевого ракурса



Рисунок 3.1 — Пример отображения S , преобразующего исходный ракурс в целевой. В области целевого ракурса, обозначенной белым цветом, неизвестны значения цвета, так как в нее не отображается ни одна из точек исходного ракурса. Такая область называется областью открытия в целевом ракурсе. Одной из важных подзадач задачи построения стереоскопических ракурсов является задача вычисления цвета в областях открытия.

путем копирования цвета из соответствующей точки исходного ракурса. Назовем такую область областью открытия, которая возникает в ходе построения целевого ракурса. Область открытия включает фрагменты фона, которые были скрыты объектами переднего плана в исходном ракурсе, но становятся видимыми после изменения ракурса просмотра (иллюстрация приведена на рисунке 3.1). Таким образом, в задаче построения целевого ракурса возникает подзадача заполнения областей открытия, то есть вычисления значений цвета в рамках области открытия в целевом ракурсе.

В главе предлагается алгоритм построения стереоскопических ракурсов, который использует результаты восстановления фона в качестве дополнительных данных для заполнения областей открытия. Для сцен вида «объект переднего плана на однородном фоне», где выполнено предположение о постоянстве диспаратности, которое утверждает, что диспаратность каждой точки фона остается неизменной от кадра к кадру на протяжении всей входной видеопоследовательности, задача полностью сводится к задаче восстановления фона, рассмотренной в первой главе. А именно, значения диспаратности в области неизвестного фона Ω , занимаемой объектом переднего плана, могут быть восстановлены простым копированием из исходных кадров, и область открытия может быть заполнена путем отображения восстановленного фона в целевой ракурс согласно восстанов-

ленным значениям диспаратности. Для расширения области применимости алгоритма также было предложено:

- Алгоритм многослойного восстановления фона, основанный на использовании в качестве входных данных карты слоев $\mathcal{L} : I \mapsto \mathbb{N}$, обобщающей бинарную маску области неизвестного фона Ω , используемую в первой главе. Данный алгоритм позволяет также заполнять области открытия около пересечения трех и более объектов, расположенных на разной глубине (раздел 3.3);
- Обобщение описанной в первой главе модели движения фона, позволяющее также моделировать межкадровое изменение значений диспаратности (раздел 3.3.1);
- Обобщение алгоритма покадрового восстановления фона с уточнением, которое позволяет также уточнять восстановленные значения диспаратности (раздел 3.3.2).

3.2 Обзор существующих алгоритмов

В научных работах было предложено множество различных подходов к решению задачи заполнения областей открытия. Их можно подразделить на следующие основные классы:

1. Алгоритмы, основанные на деформации изображений. Данные алгоритмы базируются на работах из области адаптивного изменения соотношения сторон в видео [54] и в них ставится цель максимально незаметным образом деформировать фон и объект переднего плана так, чтобы полностью предотвратить появление областей открытия. Величина требуемой деформации здесь прямо пропорциональна ширине соответствующей области открытия, что затрудняет построение стереоскопическо-

го видео с большой стереобазой с использованием алгоритмов данного класса.

2. Алгоритмы, основанные на алгоритмах восстановления фона в изображениях. Был предложен ряд модификаций традиционных алгоритмов восстановления фона в изображениях, которые используют карту глубины в качестве дополнительного источника информации. Отличительной особенностью алгоритмов данного класса является использование лишь пространственной окрестности в том же кадре для заполнения области открытия. Другими словами, данные алгоритмы не учитывают то, что искомые фрагменты фона могут быть видимы в других кадрах входной видеопоследовательности. Алгоритмы этого класса могут накладывать явные ограничения на стабильность результата заполнения во времени, однако высокое визуальное качество все равно остается труднодостижимым в видеопоследовательностях со свободно движущейся камерой.
3. Алгоритмы, основанные на алгоритмах восстановления фона в видео. Основным преимуществом этого класса алгоритмов является возможность использования для заполнения областей открытия истинного изображения фона во всех фрагментах, которые были видимы хотя бы в одном кадре входного видео. Предлагаемый алгоритм относится к данному классу.

Наиболее простым способом предотвращения появления областей открытия является применение размытия по Гауссу к карте глубины перед ее использованием для построения новых ракурсов [55]. Однако такой подход приводит к характерным искажениям возле границ объектов, которые приводят к значительному снижению визуального качества результата. Более высокое визуальное качество может быть достигнуто за счет неравномерного распределения требуемой деформации по всей области фона и/или объекта переднего плана. Например, в работе [56] предлагается метод вычисления пары отображений, преобразующих исходный ракурс в целевые левый и правый ракурсы без областей открытия, путем минимизации специальной функции энергии. Минимизируется отклоне-

ние значений диспаратности получаемой в результате отображения стереопары от целевых значений с ограничением, что в результате отображений не должно возникать областей открытия. В работе [57] было предложено добавить дополнительное слагаемое в минимизируемую функцию энергии, которое отвечает за сохранение прямых линий в результате отображения, что уменьшает заметность деформаций в результате работы алгоритма.

Авторы [58] провели экспертную оценку набора простых способов заполнения областей открытия, основанных на продолжении текстуры фона внутрь заполняемой области в построенном ракурсе. По результатам проведенного сравнения было показано, что использование простой горизонтальной экстраполяции цвета крайнего пиксела с большим значением глубины приводит к наилучшему визуальному качеству. Данный способ до сих пор нередко используется при производстве стереоскопических фильмов для заполнения областей открытия малого размера. Для улучшения качества заполнения более крупных областей открытия были предложены модификации известных алгоритмов восстановления фона в изображениях, использующие карту глубины в качестве дополнительных входных данных. Например, в работе [59] было показано, что модификация алгоритма Криминиси [60] действительно показывает более высокое визуальное качество заполнения в отдельных кадрах, однако крайне низкая временная стабильность приводит к неудовлетворительным результатам при применении данного алгоритма к видеопоследовательностям. Более современные основанные на восстановлении фона в изображениях алгоритмы как правило формулируют задачу дискретной оптимизации вместо жадной эвристики, используемой в алгоритме Криминиси. Также явным образом вводятся ограничения на временную стабильность результата. Например, в работе [61] проводится совместная минимизация заметности пространственных швов в результате заполнения и временной нестабильности результата вдоль векторов движения, экстраполированных из окружающего фона внутрь области открытия. Для уменьшения вычислительной сложности решения получаемой дискретной оптимизационной задачи используется метод Multilevel Belief Propagation. Схожий подход также использовался для построения новых

ракурсов путем экстраполяции в многоракурсном видео [62]. В работе [63] предлагается дополнительный способ уменьшения вычислительной сложности решения дискретной задачи оптимизации путем использования специализированной модификации шага передачи сообщений в алгоритме Loopy Belief Propagation.

Множество алгоритмов, основанных на алгоритмах вычитания фона, было предложено для решения задачи заполнения областей открытия в сценах со статичной камерой. Данные алгоритмы либо моделируют фон с использованием смеси Гауссиан [64; 65], либо проводят явную сегментацию на фон и передний план на базе карты глубины и затем восстанавливают статичное изображение фона [66; 67]. Некоторые алгоритмы также поддерживают простые сцены с движущейся камерой за счет моделирования движения фона с помощью проективного преобразования [68; 69]. Другие алгоритмы [70; 71] используют трехмерные пространственно-временные блоки по аналогии с классическими алгоритмами восстановления фона в видео [8]. Однако, как было показано в работе [11] у такого подхода есть существенные ограничения при обработке видео с произвольным движением камеры.

Также прямое отношение к предлагаемому алгоритму многослойного восстановления фона имеет понятие многослойных карт глубины (Layered Depth Images), предложенное в работе [72]. Авторы вводят представление сцены, где несколько пикселей с различными значениями глубины могут быть доступны вдоль каждого направления взгляда. Такое представление позволяет проводить корректное заполнение областей открытия в любых сценах путем независимого отображения каждого слоя в целевой ракурс и их композиции в целевом ракурсе в порядке приближения к наблюдателю. Целью предлагаемого алгоритма является восстановление подобного многослойного предоставления для каждого кадра входной видеопоследовательности, в отличие от известных алгоритмов, которые позволяют восстановить лишь один слой фона.

3.3 Описание предложенного алгоритма

Как уже было упомянуто ранее, предложенный алгоритм многослойного восстановления фона опирается на наличие карты слоев $\mathcal{L} : I \mapsto \mathbb{N}$. Данная карта каждой точке $p \in I$ ставит в соответствие номер слоя l , причем слои с более высоким номером могут перекрывать слои с более низким номером, в то время как обратное неверно. Пример такой карты слоев приведен на рисунке 3.2. Карта слоев \mathcal{L} может быть интерпретирована как обобщение бинарной маски области неизвестного фона Ω . Для каждого слоя l можно определить модель движения $O_l^{t \rightarrow t \pm 1}$ по аналогии с моделью движения фона, описанной в первой главе, используя $\Omega_{\mathcal{L}}^l = \{p | \mathcal{L}(p) > l\}$ в качестве области неизвестного фона. Другими словами, область, занимаемая всеми слоями, перекрывающими слой l , выступает в роли области неизвестного фона. Для всех точек $p \in \Omega_{\mathcal{L}}^{l,(K)} \subset \Omega_{\mathcal{L}}^l$, из которых слой l (то есть, область $\{p | \mathcal{L}(p) = l\}$) достижим путем применения композиции $O_l^{t \rightarrow t+1}$ или $O_l^{t \rightarrow t-1}$, цвет данного слоя $C_l(p)$ может быть вычислен простым копированием из исходных кадров по аналогии с алгоритмом восстановления фона, описанным в первой главе. Стоит отметить, что для последующего использования в ходе построения стереоскопических ракурсов достаточно восстановить цвет C_l в пределах Ω_{occ}^l :

$$\Omega_{\text{occ}}^l = \bigcup_{i=l}^{L-1} \Omega_{\mathcal{L}}^i \setminus \text{erode}(\Omega_{\mathcal{L}}^i, r^{l,i+1}) \quad (3.1)$$

Здесь L – общее количество слоев в пределах карты \mathcal{L} (то есть, все слои пронумерованы от 1 до L), $\text{erode}(\Omega, r)$ – результат морфологического сужения области Ω со структурным элементом размера $1 \times (2r + 1)$. Радиус морфологического сужения $r^{l,i}$ – это максимальная разница значений диспаратности вдоль границы между слоями l и i , а именно:

$$r^{l,i} = \max_{\substack{(p,q) \\ \mathcal{L}(p)=i, \mathcal{L}(q)=l \\ \|p-q\|_1 < 3}} |\mathcal{D}(p) - \mathcal{D}(q)| \quad (3.2)$$



(а) Исходный кадр

(б) Карта диспаратности

(в) Карта слоев

Рисунок 3.2 — Пример кадра (а) с дополнительными входными данными в виде карты диспаратности \mathcal{D} (б) и карты слоев \mathcal{L} (в).

Для фонового слоя $l = 1$ также возможно восстановить фрагменты, которые остаются скрытыми на протяжении всей входной видеопоследовательности, по аналогии с алгоритмом восстановления фона, описанным в первой главе. То есть, к подобласти области $\Omega_{\mathcal{L}}^1$, которая остается невосстановленной на последнем кадре после первого прямого прохода, и соответствующей подобласти первого кадра после обратного прохода требуется применить алгоритм пространственного восстановления фона [29].

Рассмотрим случай, когда выполнено предположение о постоянстве диспаратности, которое утверждает, что $\mathcal{D}(p) = \mathcal{D}(O^{t \rightarrow t \pm 1}(p))$ для любой точки p во входной видеопоследовательности. Другим словами, значения диспаратности остаются неизменными при отображении в соседний кадр согласно оптическому потоку $O^{t \rightarrow t \pm 1}$, что эквивалентно постоянству расстояния от каждой точки сцены до плоскости линзы камеры с течением времени. В таком случае значения диспаратности слоев $\mathcal{D}_l : \Omega_{\mathcal{L}}^{l,(K)} \mapsto \mathbb{R}$ могут быть вычислены точно так же путем копирования из исходных кадров. Зная значения цвета C_l и диспаратности \mathcal{D}_l , целевой ракурс может быть построен путем отображения исходного ракурса и каждого из восстановленных слоев согласно отображению S и их композиции в порядке уменьшения глубины (приближения к плоскости линзы камеры). Для вычисления цвета в оставшихся точках, в которые не отображается ни один из восстановленных слоев, предложенный алгоритм использует технику горизонтальной экстраполяции крайнего пиксела с большим значением глубины, описанную авторами

работы [58]. Более подробно алгоритм построения целевого ракурса на основе результатов многослойного восстановления фона описан в разделе 3.3.3.

Если же предположение о постоянстве диспаратности не выполнено, то требуется также моделировать межкадровое изменение диспаратности $\Delta \mathcal{D}_l^{t \rightarrow t \pm 1} : I \mapsto \mathbb{R}$ вдоль $O_l^{t \rightarrow t \pm 1}$ для каждого слоя l . В разделе 3.3.1 описано обобщение описанной в первой главе модели движения фона, которое позволяет также вычислять $\Delta \mathcal{D}_l^{t \rightarrow t \pm 1}$. Затем для каждого слоя необходимо построить отображение V в исходные кадры по аналогии с алгоритмом восстановления фона, описанном в первой главе, и соответствующее суммарное изменение диспаратности $V^{(\Delta \mathcal{D})}$ между исходным кадром и текущим (для упрощения обозначений номер текущего слоя l был опущен). Тогда цвет и диспаратность текущего слоя в точке $p \in \Omega_{\mathcal{L}}^{l, (K)}$ могут быть вычислены следующим образом:

$$\begin{aligned} C_l(p) &= C(V(p)), \text{ где } \mathcal{L}(V(p)) = l \\ \mathcal{D}_l(p) &= \mathcal{D}(V(p)) + V^{(\Delta \mathcal{D})}(p) \end{aligned} \tag{3.3}$$

V и $V^{(\Delta \mathcal{D})}$ строятся покадрово посредством нескольких проходов по входной последовательности кадров по аналогии с алгоритмом, описанным в первой главе. В процессе построения отображения V_t из каждого следующего кадра с номером t и соответствующего накопленного изменения диспаратности $V_t^{(\Delta \mathcal{D})}$ точно так же можно ввести перекрытия между областями определения отображений в различные исходные кадры и определить оптимизационную задачу уточнения, целью которой является компенсация накопления ошибки. Более подробно обобщение алгоритма покадрового восстановления с уточнением, которое также учитывает изменение диспаратности, описано в разделе 3.3.2.

3.3.1 Модель движения фона с учетом межкадрового изменения диспаратности

Модифицируем модель движения фона, описанную в разделе 1.3.2, путем добавления учета межкадрового изменения диспаратности и использования карты слоев \mathcal{L} вместо области неизвестного фона Ω . Для текущего слоя l как оптический поток $O_l^{t \rightarrow t+1}$, так и изменение диспаратности вдоль него $\Delta \mathcal{D}_l^{t \rightarrow t+1}$ могут быть напрямую оценены в пределах области $\{p | \mathcal{L}(p) \leq l\}$. Однако в области $\{p | \mathcal{L}(p) > l\}$, занимаемой слоями, перекрывающими слой l , как $O_l^{t \rightarrow t+1}$, так и $\Delta \mathcal{D}_l^{t \rightarrow t+1}$ моделируются путем интерполяции. А именно, для каждого слоя в текущем кадре t формулируется следующая оптимизационная задача:

$$\begin{aligned}
O_l^{t \rightarrow t+1}, \Delta \mathcal{D}_l^{t \rightarrow t+1} = \arg \min_{O, \Delta \mathcal{D}} \sum_{p \in I_t^{(s)}} \Phi \left(\sum_{\substack{q \in P_{s \times s}(p) \\ \mathcal{L}(q) \leq l \\ \mathcal{L}(q + F_O(p)) \leq l}} w_q \left(\nabla_{F_O(p)} C(q) - \frac{\sum_q w_q \nabla_{F_O(p)} C(q)}{\sum_q w_q} \right)^2 \right) \\
+ \Phi \left(\sum_{\substack{q \in P_{s \times s}(p) \\ \mathcal{L}(q) \leq l \\ \mathcal{L}(q + F_O(p)) \leq l}} (\nabla_{F_O(p)} \mathcal{D}(q) - \Delta \mathcal{D}(p))^2 \right) + \sum_{p \in I_t^{(s)}} \alpha_s \Phi \left(\left\| \frac{\partial F_O}{\partial x}(p) \right\|^2 + \left\| \frac{\partial F_O}{\partial y}(p) \right\|^2 \right) \\
+ \alpha_d \Phi \left(\left\| \frac{\partial \Delta \mathcal{D}}{\partial x}(p) \right\|^2 + \left\| \frac{\partial \Delta \mathcal{D}}{\partial y}(p) \right\|^2 \right). \tag{3.4}
\end{aligned}$$

Как и ранее модель вычисляется в узлах сетки $I_t^{(s)}$ с шагом s пикселей, значения в промежуточных точках затем оцениваются с помощью билинейной интерполяции. $\nabla_{F_O(p)} \mathcal{D}(q)$ здесь обозначает разницу между диспаратностью соответствующих точек: $\mathcal{D}(q) - \mathcal{D}(q + F_O(p))$, α_s – константа, определяющая вес слагаемого, отвечающего за гладкость поля векторов смещений F_O , α_d определяет вес слагаемого, отвечающего за гладкость функции межкадрового изменения глубины $\Delta \mathcal{D}$. Остальные обозначения идентичны используемым в (1.13).

Для решения задачи (3.4) модифицируем алгоритм, описанный в разделе 1.3.2. Как и для решения (1.13) воспользуемся пирамидальной схемой вычисления. На нижнем уровне пирамиды (то есть, наименьшем разрешении) инициализируем F_O и $\Delta \mathcal{D}$ нулевыми значениями, на всех остальных уровнях начальное

приближение определяется посредством масштабирования результата с предыдущего уровня пирамиды. На каждом уровне пирамиды для каждой итерации $k = 1 \dots N_{OF}$ сначала найдем приращение ΔF_O^k , зафиксировав переменные $\Delta \mathcal{D}$, затем обновим текущие значения $\Delta \mathcal{D}$, зафиксировав межкадровые смещения F_O . Таким образом задача (3.4) также сводится к решению набора СЛАУ с разреженной матрицей методом релаксации.

3.3.2 Алгоритм покадрового восстановления фона с учетом изменения диспаратности

Модифицируем алгоритм покадрового восстановления фона с уточнением, описанный в разделе 1.3.4, путем учета накопления межкадрового изменения диспаратности и использования входной карты слоев \mathcal{L} . В каждом из проходов для каждого кадра t и каждого слоя l помимо построения начального приближения отображения V_t из текущего в исходные кадры также вычислим начальное приближение функции суммарного изменения диспаратности $V_t^{(\Delta \mathcal{D})}$ с точно такой же областью определения:

$$V_t^{(\Delta \mathcal{D})}(p) = \Delta \mathcal{D}_l^{t \rightarrow t \pm 1}(p) + V_{t \pm 1}^{(\Delta \mathcal{D})}(O_l^{t \rightarrow t \pm 1}(p)) \quad (3.5)$$

На базе начальных приближений V_t и $V_t^{(\Delta \mathcal{D})}$ затем формулируется схожая оптимизационная задача уточнения, целью которой является предотвращение накопления ошибки по мере увеличения (уменьшения) номера кадра t :

$$V_t, V_t^{(\Delta \mathcal{D})} = \arg \min_{V_t, V_t^{(\Delta \mathcal{D})}} (E_D(V_t, V_t^{(\Delta \mathcal{D})}) + \lambda_s E_S(V_t) + \lambda_d E_d(V_t^{(\Delta \mathcal{D})})), \text{ где} \quad (3.6)$$

$$\begin{aligned}
E_D(V_t, V_t^{(\Delta \mathcal{D})}) &= \sum_{\substack{p \in I_t^{(1)} \\ \bar{\mathcal{L}}(p) > l, \mathcal{L}(p) = l}} \sum_{i=1}^{N(p)} \Phi \left(\left(C(V_t^{v_i(p)}(p)) - C(p) \right)^2 \right) + \\
&\quad \Phi \left(\left(\mathcal{D}(V_t^{v_i(p)}(p)) + V_t^{(\Delta \mathcal{D}), v_i(p)}(p) - \mathcal{D}(p) \right)^2 \right) + \\
&\quad \sum_{\substack{p \in I_t^{(1)} \\ \mathcal{L}(p) > l}} \sum_{i=1}^{N(p)-1} \Phi \left(\left(C(V_t^{v_{i+1}(p)}(p)) - C(V_t^{v_i(p)}(p)) \right)^2 \right) + \\
&\quad \Phi \left(\left(\mathcal{D}(V_t^{v_{i+1}(p)}(p)) + V_t^{(\Delta \mathcal{D}), v_{i+1}(p)}(p) - \right. \right. \\
&\quad \left. \left. \mathcal{D}(V_t^{v_i(p)}(p)) - V_t^{(\Delta \mathcal{D}), v_i(p)}(p) \right)^2 \right), \\
E_S(V_t) &= \sum_{i=1}^n \sum_{\substack{p \in I_t^{(1)} \\ p \in \Omega_t^{t_i}}} \left\| \frac{\partial F_{V_t^{t_i}}(p)}{\partial x} \right\|^2 + \left\| \frac{\partial F_{V_t^{t_i}}(p)}{\partial y} \right\|^2, \\
E_d(V_t^{(\Delta \mathcal{D})}) &= \sum_{i=1}^n \sum_{\substack{p \in I_t^{(1)} \\ i \in \Omega_t^{t_i}}} \left| \frac{\partial V_t^{(\Delta \mathcal{D}), t_i}}{\partial x}(p) \right|^2 + \left| \frac{\partial V_t^{(\Delta \mathcal{D}), t_i}}{\partial y}(p) \right|^2.
\end{aligned}$$

Здесь $\bar{\mathcal{L}}$ обозначает результат морфологического расширения карты слоев \mathcal{L} с квадратным структурным элементом размера $d \times d$. $V_t^{(\Delta \mathcal{D}), t_i}$ аналогично $V_t^{t_i}$ обозначает часть $V_t^{(\Delta \mathcal{D})}$ соответствующую отображению в исходный кадр с номером t_i . Остальные обозначения идентичны используемым в (1.26). Данная оптимизационная задача позволяет одновременно уточнять как отображения в исходные кадры $V_t^{t_i}$, так и соответствующие значения разности диспаратности $V_t^{(\Delta \mathcal{D}), t_i}$ между текущим кадром и кадром с номером t_i .

Для решения задачи (3.6) модифицируем алгоритм, описанный в разделе 1.3.4. Зафиксировав переменные $V_t^{(\Delta \mathcal{D})}$, получим задачу уточнения V_t , идентичную задаче (1.26). После ее решения зафиксируем V_t и найдем соответствующие оптимальные значения $V_t^{(\Delta \mathcal{D})}$, что сводится к решению набора СЛАУ методом сопряженных градиентов, аналогично решению задачи (1.26).

3.3.3 Алгоритм построения ракурсов на базе результата многослойного восстановления фона

Результатом многослойного восстановления фона являются значения цвета C_l и диспаратности \mathcal{D}_l для каждого слоя l . Алгоритм построения целевого ракурса с их использованием состоит из следующих основных шагов:

1. Отобразить исходный ракурс в целевой согласно значениям диспаратности \mathcal{D} . В результате получим C_{view} и $\mathcal{D}_{\text{view}}$ – значения цвета и диспаратности целевого ракурса, определенные на $I \setminus \Omega_{\text{occl}}$, где Ω_{occl} – область открытия (множество точек целевого ракурса, куда не отображается ни одна из точек исходного).
2. Для каждого из слоев $l = 1 \dots L$ выполнить:
 - 2.1. Отобразить слой l в целевой ракурс согласно значениям диспаратности \mathcal{D}_l . Получим C_{view}^l и $\mathcal{D}_{\text{view}}^l$, определенные на $I \setminus \Omega_{\text{occl}}^l$ в целевом ракурсе.
 - 2.2. Обновить цвет целевого ракурса путем выполнения присваивания $C_{\text{view}}(p) := W^l(p) \cdot C_{\text{view}}^l(p) + (1 - W^l(p)) \cdot C_{\text{view}}(p)$ для всех пикселей p лежащих одновременно в области определения C_{view}^l и C_{view} . Для p лежащих в области определения лишь одной из них выполняется простое копирование соответствующего значения в $C_{\text{view}}(p)$. Значения диспаратности целевого ракурса $\mathcal{D}_{\text{view}}$ обновляются полностью аналогичным образом. Веса $W^l(p)$ определяются как $(1 + e^{-a * (\mathcal{D}_{\text{view}}^l(p) - \mathcal{D}_{\text{view}}(p) - b)})^{-1}$, то есть как сигмоида, которая в мягкой форме обеспечивает то, что точки с меньшим значением глубины (диспаратности) перекрывают точки с большим ее значением.
3. Области, остающиеся за пределами области определения C_{view} и $\mathcal{D}_{\text{view}}$ после выполнения всех предыдущих шагов, заполняются с помощью ме-

тогда горизонтальной экстраполяции цвета крайнего пиксела с большим значением глубины, предложенного в работе [58].

3.4 Анализ результатов работы предложенного алгоритма

Экспериментальная оценка предложенного алгоритма была проведена на тестовой выборке, состоящей из 4 видеопоследовательностей: двух синтетических последовательностей из выборки, предложенной в работе [73], включающей эталонные карты диспаратности и карты сегментации на объекты, и двух снятых видеопоследовательностей с полуавтоматически построенными картами глубины и масками объектов переднего плана. Для синтетических последовательностей с полной сегментацией на слои используется многослойное восстановление фона в пределах областей $\Omega_{\text{осс}}^l$, определенных в (3.1). Для снятых последовательностей, где доступны лишь бинарные маски переднего плана, проводится полное восстановление фонового слоя путем использования алгоритма пространственного восстановления фона на первом и последнем кадрах.

Предложенный алгоритм путем экспертной оценки сравнивается с другим алгоритмом заполнения областей открытия с помощью восстановления фона [27] и простым алгоритмом заполнения путем горизонтальной экстраполяции цвета [58]. Так как искажения в результатах построения ракурсов являются сильно локализованными, респондентами проводится сравнение увеличенных фрагментов кадров ($1/4$ от ширины и высоты кадра). Номер кадра и пространственное положение этих фрагментов выбирались случайным образом с единственным ограничением, что получаемые фрагменты должны содержать хотя бы 1 пиксел области открытия. Данная процедура случайного выбора применялась как к построенному левому, так и правому ракурсам; в результате было получено 32 различных фрагмента. Результаты работы алгоритмов в пределах данных фрагментов затем попарно сравнивались 152 респондентами, принявшими участие в проведенном

Алгоритм	Предложенный	Зачесов и др. [27]	Vazquez et al. [58]
Предложенный	—	72.3%	73.9%
Зачесов и др. [27]	27.7%	—	52.8%
Vazquez et al. [58]	26.1%	47.2%	—

Таблица 7 — Результаты экспертного сравнения предложенного алгоритма с аналогами на тестовой выборке из 32 фрагментов кадров. Здесь в каждой ячейке таблицы приводится частота предпочтения результата алгоритма из соответствующей строки результату алгоритма из соответствующего столбца. Например, респонденты указали, что результат предложенного алгоритма превосходит результат алгоритма из работы 2015 года [27] по визуальному качеству в 72.3% случаев.

эксперименте. Для каждой пары результатов требовалось выбрать результат с более высоким визуальным качеством. Каждый участник сравнил 20 пар, включая 2 контрольных вопроса, где требовалось сравнить результаты с кардинально различающимся уровнем качества. Результаты респондентов, которые неправильно ответили хотя бы на один контрольный вопрос, исключались из сравнения. Согласно полученным результатам, которые приведены в таблице 7, предложенный алгоритм превзошел аналоги по визуальному качеству. Данное экспертное сравнение проводилось в сети Интернет с использованием онлайн-платформы *Subjectify.us*. Также на рисунке 3.3 приводится набор примеров работы различных алгоритмов. Стоит отметить, что наиболее заметным преимуществом предложенного алгоритма оказывается в областях около пересечения трех и более объектов, расположенных на разной глубине.

3.5 Выводы по главе 3

В главе был предложен алгоритм построения стереоскопических ракурсов на базе алгоритма восстановления фона в видео, предложенного в первой гла-

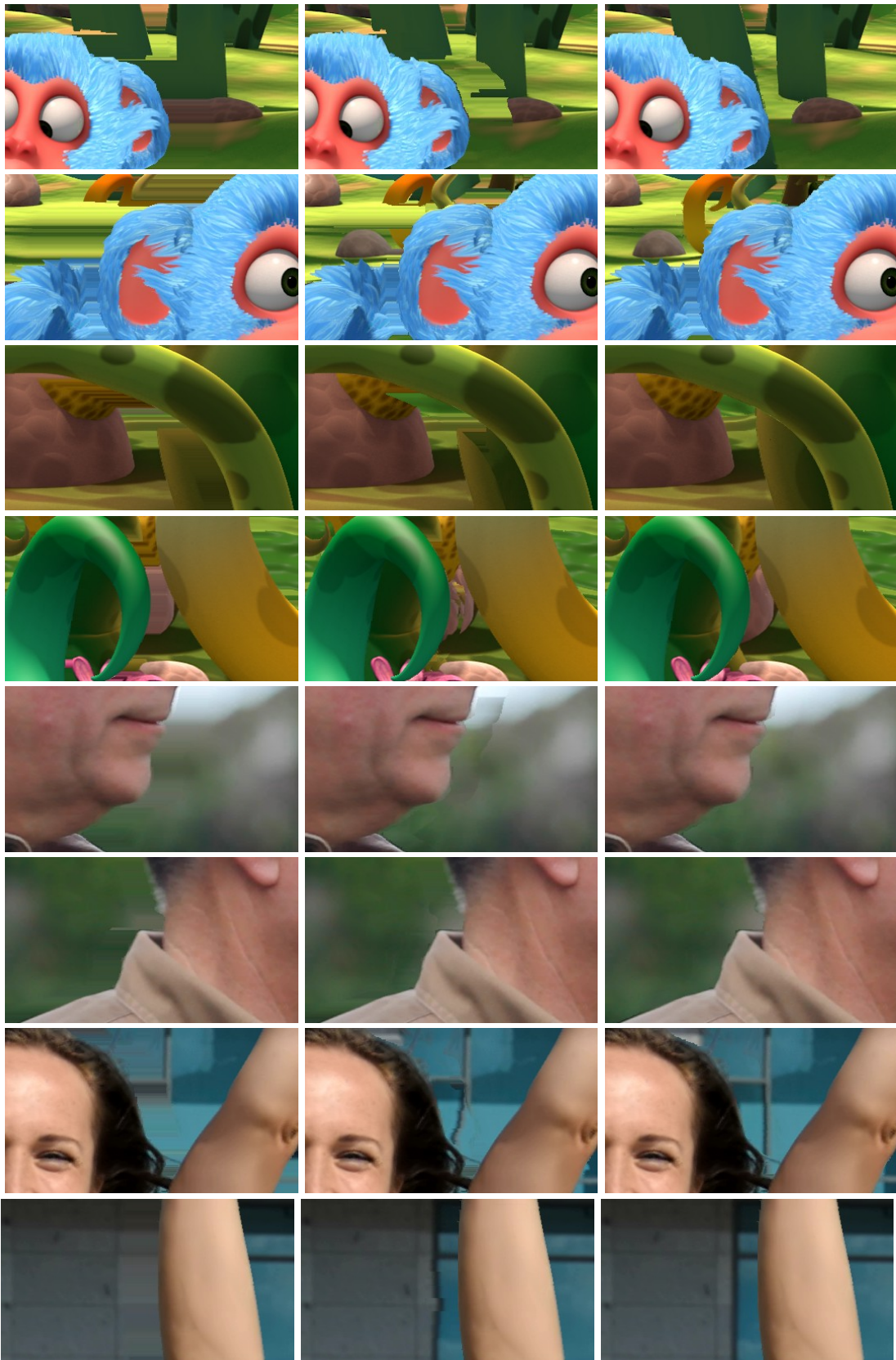


Рисунок 3.3 — Результаты построения новых ракурсов с помощью различных алгоритмов (слева направо): горизонтальная экстраполяция цвета [58], алгоритм, основанный на однослойном восстановлении фона [27] и предложенный алгоритм.

ве. Ключевой отличительной особенностью предложенного алгоритма является проведение многослойного восстановления фона, что позволяет точнее заполнять области открытия около пересечения трех и более объектов, расположенных на разной глубине. По результатам проведенной экспертной оценки предложенный алгоритм превзошел аналоги по визуальному качеству результата. Однако предложенный алгоритм требует наличия карты слоев в качестве дополнительных входных данных, что может затруднять его применение на практике.

Результаты главы опубликованы автором в [4; 6].

Заключение

В ходе диссертационного исследования были получены следующие основные результаты:

1. Предложен новый подход к решению проблемы заполнения областей открытия на основе проведения многослойного восстановления фона в видео с использованием непараметрической модели движения для каждого слоя.
2. Разработан новый алгоритм восстановления фона в видео с пространственной сложностью, не зависящей от количества кадров во входной видеопоследовательности. По результатам экспериментальной оценки предложенный алгоритм превзошел существующие аналоги по точности найденного решения и не уступил аналогам по результатам экспертной оценки.
3. Предложено два новых функционала качества работы алгоритмов восстановления фона в видео, основанных на критерии локального сходства с эталоном, которые обладают более высокой корреляцией с экспертной оценкой по сравнению с ранее использовавшимися для этой цели функционалами.
4. Все предложенные алгоритмы и функционалы качества были реализованы, а также проведены вычислительные эксперименты, подтверждающие их эффективность. Алгоритм восстановления фона был протестирован на реальных данных, предоставленных коммерческой компанией.

Предложенные алгоритмы могут быть использованы для улучшения существующих программных инструментов для полуавтоматического преобразования моноскопического видео в стереоскопический формат, а предложенная методология сравнения алгоритмов восстановления фона в видео может быть использована в ходе разработке новых алгоритмов и подборе их параметров. Дальнейшее развитие темы исследования может включать в себя:

- Исследование альтернативных моделей движения фона, в том числе предсказание векторного поля движения в области неизвестного фона средствами машинного обучения, например с помощью нейронных сетей.
- Исследование нейросетевых алгоритмов восстановления фона в изображениях для повышения качества работы для фрагментов фона, остающихся скрытыми на протяжении всей входной видеопоследовательности.

Список литературы

1. *Боков А. А., Ватолин Д. С.* Восстановление фона в видео с использованием непараметрической модели движения и покадрового уточнения // *Цифровая обработка сигналов.* — 2017. — № 3. — С. 13—21.
2. *Боков А. А., Ватолин Д. С.* Методика объективной оценки качества восстановления фона в видео // *Цифровая обработка сигналов.* — 2016. — № 3. — С. 26—33.
3. Toward an Objective Benchmark for Video Completion / A. Bokov [et al.] // *Signal, Image and Video Processing.* — 2019. — Vol. 13. — P. 601–608. — DOI: 10.1007/s11760-018-1387-5.
4. *Bokov A., Vatolin D.* Multilayer RGBD-Video Completion For Hole Filling in 3D-View Synthesis // *International Conference on 3D Immersion (IC3D).* — 2018. — P. 1–8. — DOI: 10.1109/IC3D.2018.8657873.
5. *Bokov A., Vatolin D.* 100+ Times Faster Video Completion by Optical-Flow-Guided Variational Refinement // *IEEE International Conference on Image Processing (ICIP).* — 2018. — P. 2122–2126. — DOI: 10.1109/ICIP.2018.8451683.
6. *Bokov A., Vatolin D.* Toward efficient background reconstruction for 3D-view synthesis in dynamic scenes // *IEEE International Conference on Multimedia Expo Workshops (ICMEW).* — 2017. — P. 37–42. — DOI: 10.1109/ICMEW.2017.8026297.
7. DeepFlow: Large displacement optical flow with deep matching / P. Weinzaepfel [и др.] // *International Conference on Computer Vision (ICCV).* — 2013. — С. 1385—1392. — DOI: 10.1109/ICCV.2013.175.
8. *Wexler Y., Shechtman E., Irani M.* Space-time video completion // *IEEE Conference on Computer Vision and Pattern Recognition.* Т. 1. — 2004. — С. I-120—I-127.

9. Background inpainting for videos with dynamic objects and a free-moving camera / M. Granados [и др.] // European Conference on Computer Vision. — 2012. — С. 682—695.
10. Video inpainting of complex scenes / A. Newson [и др.] // SIAM Journal on Imaging Sciences. — 2014. — Т. 7, № 4. — С. 1993—2019.
11. Temporally coherent completion of dynamic video / J.-B. Huang [и др.] // ACM Transactions on Graphics (TOG). — 2016. — Т. 35, № 6. — 196:1—196:11. — DOI: 10.1145/2980179.2982398.
12. *Sobral A., Vacavant A.* A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos // Computer Vision and Image Understanding. — 2014. — Т. 122. — С. 4—21.
13. Video completion for perspective camera under constrained motion / Y. Shen [и др.] // International Conference on Pattern Recognition. Т. 3. — 2006. — С. 63—66.
14. *Roxas M., Shiratori T., Ikeuchi K.* Video completion via spatio-temporally consistent motion inpainting // IPSJ Transactions on Computer Vision and Applications. — 2014. — Т. 6. — С. 98—102.
15. PatchMatch: A randomized correspondence algorithm for structural image editing / C. Barnes [и др.] // ACM Transactions on Graphics. — 2009. — Т. 28, № 3. — 24:1—24:11.
16. Motion-consistent video inpainting / T. Le [и др.] // IEEE International Conference on Image Processing. — 2017. — С. 2094—2098.
17. *Patwardhan K. A., Sapiro G., Bertalmio M.* Video inpainting under constrained camera motion // IEEE Transactions on Image Processing. — 2007. — Т. 16, № 2. — С. 545—553.
18. *Cheung S.-C. S., Zhao J., Venkatesh M. V.* Efficient object-based video inpainting // IEEE International Conference on Image Processing (ICIP). — 2006. — С. 705—708.

19. *Ebdelli M., Guillemot C., Le Meur O.* Exemplar-based video inpainting with motion-compensated neighbor embedding // IEEE International Conference on Image Processing (ICIP). — 2012. — С. 1737—1740.
20. *Pérez P., Gangnet M., Blake A.* Poisson image editing. — 2003. — DOI: 10.1145/1201775.882269.
21. *Ebdelli M., Le Meur O., Guillemot C.* Video inpainting with short-term windows: application to object removal and error concealment // IEEE Transactions on Image Processing. — 2015. — Т. 24, № 10. — С. 3034—3047.
22. Video completion by motion field transfer / T. Shiratori [и др.] // IEEE Conference on Computer Vision and Pattern Recognition. Т. 1. — 2006. — С. 411—418.
23. Video completion via motion guided spatial-temporal global optimization / M. Liu [и др.] // ACM International Conference on Multimedia. — 2009. — С. 537—540.
24. Video inpainting on digitized vintage films via maintaining spatiotemporal continuity / N.-C. Tang [и др.] // IEEE Transactions on Multimedia. — 2011. — Т. 13, № 4. — С. 602—614.
25. Video Background Completion Using Motion-Guided Pixel Assignment Optimization / Z. Xu [и др.] // IEEE Transactions on Circuits and Systems for Video Technology. — 2016. — Т. 26, № 8. — С. 1393—1406.
26. *Strobel M., Diebold J., Cremers D.* Flow and color inpainting for video completion // German Conference on Pattern Recognition. — 2014. — С. 293—304.
27. *Зачесов А., Ерофеев М., Ватолин Д.* Использование карт глубины при восстановлении фона в видеопоследовательностях // Новые информационные технологии в автоматизированных системах: материалы научно-практического семинара. — 2015. — № 18.

28. *Herling J., Broll W.* High-quality real-time video inpainting with PixMix // IEEE Transactions on Visualization and Computer Graphics. — 2014. — Т. 20, № 6. — С. 866—879.
29. *He K., Sun J.* Statistics of patch offsets for image completion // European Conference on Computer Vision (ECCV). — Springer, 2012. — С. 16—29. — DOI: 10.1007/978-3-642-33709-3_2.
30. *Telea A.* An image inpainting technique based on the fast marching method // Journal of Graphics Tools. — 2004. — Т. 9, № 1. — С. 23—34.
31. Full-frame video stabilization with motion inpainting / Y. Matsushita [и др.] // IEEE Transactions on Pattern Analysis and Machine Intelligence. — 2006. — Т. 28, № 7. — С. 1150—1163.
32. Steadyflow: Spatially smooth optical flow for video stabilization / S. Liu [и др.] // IEEE Conference on Computer Vision and Pattern Recognition. — 2014. — С. 4209—4216.
33. Fast global image smoothing based on weighted least squares / D. Min [и др.] // IEEE Transactions on Image Processing. — 2014. — Т. 23, № 12. — С. 5638—5653.
34. The Foundry Nuke. — <https://www.thefoundry.co.uk/products/nuke/>.
35. Pixel Farm PFClean. — <http://www.thepixelfarm.co.uk/pfclean/>.
36. Image completion using planar structure guidance / J.-B. Huang [и др.] // ACM Transactions on Graphics (TOG). — 2014. — Т. 33, № 4. — С. 129.
37. Video error concealment using spatio-temporal boundary matching and partial differential equation / Y. Chen [и др.] // IEEE Transactions on Multimedia. — 2008. — Т. 10, № 1. — С. 2—15.
38. Sequential error concealment for video/images by weighted template matching / J. Koloda [и др.] // Data Compression Conference (DCC). — 2012. — С. 159—168.

39. Sequential error concealment for video/images by sparse linear prediction / J. Koloda [и др.] // *IEEE Transactions on Multimedia*. — 2013. — Т. 15, № 4. — С. 957—969.
40. Image quality assessment: from error visibility to structural similarity / Z. Wang [и др.] // *IEEE Transactions on Image Processing (TIP)*. — 2004. — Т. 13, № 4. — С. 600—612. — DOI: 10.1109/TIP.2003.819861.
41. *Yan W. Q., Wang J., Kankanhalli M. S.* Automatic video logo detection and removal // *Multimedia Systems*. — 2005. — Т. 10, № 5. — С. 379—391.
42. *Erofeev M., Vatolin D.* Automatic logo removal for semitransparent and animated logos // *Proceedings of GraphiCon 2011*. — 2011. — С. 26—30.
43. *Mosleh A., Bouguila N., Hamza A. B.* Automatic inpainting scheme for video text detection and removal // *IEEE Transactions on Image Processing (TIP)*. — 2013. — Т. 22, № 11. — С. 4460—4472.
44. A New Low-Rank Tensor Model for Video Completion / W. Hu [и др.] // *arXiv preprint arXiv:1509.02027*. — 2015.
45. *Mosleh A., Bouguila N., Hamza A. B.* Video completion using bandlet transform // *IEEE Transactions on Multimedia*. — 2012. — Т. 14, № 6. — С. 1591—1601.
46. *Mosleh A., Bouguila N., Hamza A. B.* Bandlet-based sparsity regularization in video inpainting // *Journal of Visual Communication and Image Representation*. — 2014. — Т. 25, № 5. — С. 855—863.
47. Robust and Fast Motion Estimation for Video Completion / S. You [и др.] // *International Conference on Machine Vision Applications (MVA)*. — 2013. — С. 181—184.
48. *Benoit J., Paquette E.* Localized search for high definition video completion // *Journal of WSCG*. — 2015.
49. *Ilan S., Shamir A.* A Survey on Data-Driven Video Completion // *Computer Graphics Forum*. Т. 34. — 2015. — С. 60—85.

50. Blender. — <https://www.blender.org/>.
51. *Wang Z., Simoncelli E. P., Bovik A. C.* Multiscale structural similarity for image quality assessment // Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers. T. 2. — 2003. — C. 1398—1402.
52. *Thurstone L. L.* A law of comparative judgment // Psychological review. — 1927.
53. *Erofeev M., Vatolin D.* Multilayer semitransparent-edge processing for depth-image-based rendering // International Conference on 3D Imaging (IC3D). — 2016. — C. 1—7.
54. Content-aware video retargeting using object-preserving warping / S.-S. Lin [и др.] // IEEE Transactions on Visualization and Computer Graphics. — 2013. — Т. 19, № 10. — С. 1677—1686.
55. *Zhang L., Tam W. J.* Stereoscopic image generation based on depth images for 3D TV // IEEE Transactions on Broadcasting. — 2005. — Т. 51, № 2. — С. 191—199.
56. StereoBrush: Interactive 2D to 3D Conversion Using Discontinuous Warps / O. Wang [и др.] // Proceedings of the Eighth Eurographics Symposium on Sketch-Based Interfaces and Modeling. — ACM, 2011. — С. 47—54. — DOI: 10.1145/2021164.2021173.
57. Line-preserving hole-filling for 2D-to-3D conversion / N. Plath [и др.] // Proceedings of the 11th European Conference on Visual Media Production. — ACM. 2014. — С. 8.
58. *Vázquez C., Tam W. J., Speranza F.* Stereoscopic imaging: filling disoccluded areas in depth image-based rendering // Three-Dimensional TV, Video, and Display V. T. 6392. — 2006. — C. 63920D.
59. *Azzari L., Battisti F., Gotchev A.* Comparative analysis of occlusion-filling techniques in depth image-based rendering for 3D videos // Proceedings of the 3rd Workshop on Mobile Video Delivery. — ACM. 2010. — С. 57—62.

60. *Criminisi A., Pérez P., Toyama K.* Region filling and object removal by exemplar-based image inpainting // *IEEE Transactions on Image Processing.* — 2004. — T. 13, № 9. — С. 1200—1212.
61. *Hsu H.-A., Chiang C.-K., Lai S.-H.* Spatio-temporally consistent view synthesis from video-plus-depth data with global optimization // *IEEE Transactions on Circuits and Systems for Video Technology.* — 2014. — T. 24, № 1. — С. 74—84.
62. *Kim H. G., Ro Y. M.* Multi-view stereoscopic video hole filling considering spatio-temporal consistency and binocular symmetry for synthesized 3D video // *IEEE Transactions on Circuits and Systems for Video Technology.* — 2016.
63. *Luo G., Zhu Y., Guo B.* Fast MRF-Based Hole Filling for View Synthesis // *IEEE Signal Processing Letters.* — 2018. — T. 25, № 1. — С. 75—79.
64. Novel temporal domain hole filling based on background modeling for view synthesis / W. Sun [и др.] // *IEEE International Conference On Image Processing (ICIP).* — 2012. — С. 2721—2724.
65. *Rahaman D. M., Paul M.* Hole-filling for single-view plus-depth based rendering with temporal texture synthesis // *IEEE International Conference on Multimedia & Expo Workshops (ICMEW).* — 2016. — С. 1—6.
66. Depth image-based rendering with advanced texture synthesis for 3-D video / P. Ndjiki-Nya [и др.] // *IEEE Transactions on Multimedia.* — 2011. — T. 13, № 3. — С. 453—465.
67. *Muddala S., Olsson R., Sjöström M.* Spatio-temporal consistent depth-image-based rendering using layered depth image and inpainting // *EURASIP Journal on Image and Video Processing.* — 2016. — T. 2016, № 1.
68. A Hole Filling Approach Based on Background Reconstruction for View Synthesis in 3D Video / G. Luo [и др.] // *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.* — 2016. — С. 1781—1789.

69. *Lie W.-N., Chiu S.-T., Chiang J.-C.* Semi-automatic 2D-to-3D video conversion based on background sprite generation // Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). — 2016. — C. 1—5.
70. *Choi S., Ham B., Sohn K.* Space-time hole filling with random walks in view extrapolation for 3D video // IEEE Transactions on Image Processing. — 2013. — T. 22, № 6. — C. 2429—2441.
71. Coherent spatial and temporal occlusion generation / R. K. Gunnewiek [и др.] // IS&T/SPIE Electronic Imaging. — 2009. — C. 723713-1—723713-10.
72. Layered depth images / J. Shade [и др.] // Proceedings of the 25th annual conference on Computer graphics and interactive techniques. — 1998. — C. 231—242.
73. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation / N. Mayer [и др.] // IEEE International Conference on Computer Vision and Pattern Recognition (CVPR). — 2016. — URL: <http://lmb.informatik.uni-freiburg.de/Publications/2016/MIFDB16> ; arXiv:1512.02134.

Список рисунков

- 1.1 Пример кадра входной видеопоследовательности (а) с определенной областью неизвестного фона (б) и результатом работы алгоритма восстановления фона в рамках данной области. 12
- 1.2 Иллюстрация используемого подхода к восстановлению фона. Точка $p \in \Omega^{(K)}$ может быть отображена в область известного фона композицией межкадровых отображений $O^{t \rightarrow t-1}$, что позволяет восстановить фон простым копированием. Точка $q \in \Omega^{(U)}$ не имеет прямого отображения в область известного фона, но цвет в ней может быть восстановлен путем копирования из области $\Omega_T^{(U)} \subset \Omega^{(U)}$ на последнем кадре, к которой был применен алгоритм восстановления фона в изображениях. 20
- 1.3 Иллюстрация разницы между традиционным оптическим потоком (в) и искомым отображением $O^{t \rightarrow t \pm 1}$ (г), моделирующим движение фона как внутри области неизвестного фона, так и вне ее. В визуализации цвет определяется направлением вектора межкадрового смещения соответствующей точки, а яркость определяется длиной соответствующего вектора. 21
- 1.4 Иллюстрация разложения области Ω_t на набор непересекающихся областей определения отображений в различные кадры $\Omega_t^{t_i}$ и область $\Omega_t^{(U)}$, где отображение еще не определено, в рамках базового алгоритма покадрового восстановления фона. 37

- 1.5 Иллюстрация многозначного отображения V_t , используемого в алгоритме покадрового восстановления с уточнением, которое допускает наличие нескольких отображений в различные исходные кадры в каждой точке p . Его можно разложить на набор отображений в различные исходные кадры $V_t^{t_i}$ с возможно пересекающимися областями определения $\Omega_t^{t_i}$, вложенными в расширенную область неизвестного фона $\bar{\Omega}_t$ 39
- 2.1 Пример из составленной тестовой выборки, включающий фрагмент входного кадра видео (а), соответствующую маску области неизвестного фона (б), и эталонное изображение фона (в). 64
- 2.2 Общая схема простого жадного алгоритма, используемого для совместной минимизации функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ на базе предоставленного начального приближения. В ходе алгоритма итеративно вычисляются векторные поля $NNF_{t \rightarrow t-1}$, сопоставляющие блоки соседних кадров, и векторные поля $NNF_{t \rightarrow GT}$, сопоставляющие каждому блоку текущего кадра наиболее схожий блок в эталонном видео, с помощью алгоритма PatchMatch [15]. На основе вычисленных векторных полей проводится покадровая реконструкция, то есть обновление значений в области неизвестного фона с помощью (2.10). После достижения сходимости происходит переход на следующий уровень Гауссовой пирамиды, где процесс повторяется. 71
- 2.3 Значения функционалов $MS-C_{MSE}$ и $MS-C_{MSEdt}$ до и после применения алгоритма их минимизации к результатам различных алгоритмов, усредненные по всем тестовым видеопоследовательностям. Весовые коэффициенты для разных уровней Гауссовой пирамиды в функционалах были выбраны в соответствии с описанием в разделе 2.6. 73

- 2.4 Общее ранжирование алгоритмов восстановления фона в видео по визуальному качеству, полученное в результате экспертной оценки. . . . 74
- 2.5 Иллюстрация эффекта регуляризации при выборе оптимальных весов в ходе максимизации корреляции с экспертной оценкой. 75
- 2.6 Итоговые значения корреляции рассмотренных функционалов качества восстановленного фона с экспертной оценкой. Корреляция вычислялась на двух различных наборах данных: полном наборе, включающем синтетические результаты, полученные путем процедуры оптимизации (см. раздел 2.5), и наборе, включающем только результаты 6 оригинальных алгоритмов восстановления фона. . . 76
- 3.1 Пример отображения S , преобразующего исходный ракурс в целевой. В области целевого ракурса, обозначенной белым цветом, неизвестны значения цвета, так как в нее не отображается ни одна из точек исходного ракурса. Такая область называется областью открытия в целевом ракурсе. Одной из важных подзадач задачи построения стереоскопических ракурсов является задача вычисления цвета в областях открытия. 80
- 3.2 Пример кадра (а) с дополнительными входными данными в виде карты диспаратности D (б) и карты слоев \mathcal{L} (в). 86
- 3.3 Результаты построения новых ракурсов с помощью различных алгоритмов (слева направо): горизонтальная экстраполяция цвета [58], алгоритм, основанный на однослойном восстановлении фона [27] и предложенный алгоритм. 94

Список таблиц

- 1 Сравнение вычислительной сложности предложенного алгоритма восстановления фона с аналогами. Приведенные оценки пространственной сложности не включают память, занимаемую входными и выходными данными алгоритма. 53
- 2 Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества MS-DSSIM, оценивающим близость результата восстановления фона к эталону (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений MS-DSSIM и затем для каждого алгоритма было вычислено среднее место в рейтинге. 55
- 3 Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества MS-DSSIMdt, оценивающим стабильность результата восстановления фона во времени (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений MS-DSSIMdt и затем для каждого алгоритма было вычислено среднее место в рейтинге. 55
- 4 Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества MS-C_{DSSIM}, оценивающим локальное сходство результата восстановления фона с эталоном (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений MS-C_{DSSIM} и затем для каждого алгоритма было вычислено среднее место в рейтинге. 56

- 5 Результаты сравнения предложенного алгоритма с 6 аналогами согласно функционалу качества $MS-C_{DSSIMdt}$, оценивающим стабильность во времени локального сходства результата восстановления фона с эталоном (меньшие значения соответствуют более высокому качеству). В каждой тестовой последовательности алгоритмы были упорядочены по возрастанию значений $MS-C_{DSSIMdt}$ и затем для каждого алгоритма было вычислено среднее место в рейтинге. Стоит отметить, что данный функционал качества продемонстрировал наивысшую корреляцию с экспертной оценкой среди всех функционалов, исследованных во второй главе. 56
- 6 Результаты экспертного сравнения предложенного алгоритма с аналогами на тестовой выборке из 26 видео. Здесь в каждой ячейке таблицы приводится частота предпочтения результата алгоритма из соответствующей строки результату алгоритма из соответствующего столбца. Например, респонденты указали, что результат предложенного алгоритма превосходит результат алгоритма из работы 2016 года [11] по визуальному качеству в 52% случаев. 57
- 7 Результаты экспертного сравнения предложенного алгоритма с аналогами на тестовой выборке из 32 фрагментов кадров. Здесь в каждой ячейке таблицы приводится частота предпочтения результата алгоритма из соответствующей строки результату алгоритма из соответствующего столбца. Например, респонденты указали, что результат предложенного алгоритма превосходит результат алгоритма из работы 2015 года [27] по визуальному качеству в 72.3% случаев. . . 93