# ЯРОСЛАВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ им. П.Г. ДЕМИДОВА

На правах рукописи

Матвеев Дмитрий Вячеславович

# РАЗРАБОТКА АЛГОРИТМОВ АНАЛИЗА АУДИТОРИИ ДЛЯ СИСТЕМ ПРИКЛАДНОГО ТЕЛЕВИДЕНИЯ

#### **ДИССЕРТАЦИЯ**

на соискание ученой степени кандидата технических наук по специальности 05.12.04 – Радиотехника, в том числе системы и устройства телевидения

Научный руководитель – д.т.н., доцент Приоров Андрей Леонидович

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
ГЛАВА 1. СОВРЕМЕННЫЕ МЕТОДЫ, ЗАДАЧИ И АЛГОРИТМЫ В	
ОБЛАСТИ СИСТЕМ АНАЛИЗА АУДИТОРИИ	. 12
1.1. Вводные замечания	
1.2. Признаковое описание изображений	. 14
1.3. Методы и алгоритмы на основе машинного обучения	
1.3.1. Решающие деревья	
1.3.2. Бустинг	
1.3.3. Машина опорных векторов	
1.3.4. Нейронные сети	
1.4. Особенности построения систем анализа аудитории в задачах	
спортивной видеоаналитики	. 30
1.5. Краткие выводы	
ГЛАВА 2. РАЗРАБОТКА И ИССЛЕДОВАНИЕ АЛГОРИТМА	
ДЕТЕКТИРОВАНИЯ ЛИЦ НА ОСНОВЕ КАСКАДА АНСАМБЛЕЙ	
РЕШАЮЩИХ ДЕРЕВЬЕВ	
2.1. Вводные замечания	. 35
2.2. Алгоритм детектирования на основе каскада ансамблей решающих	
деревьев	
2.3. Обучение детектора	
2.4. Тестирование модификаций детектора	. 44
2.5. Тестирование алгоритма детектирования в условиях присутствия	
искажений и помех на изображениях	
2.6. Краткие выводы	. 54
ГЛАВА 3. РАЗРАБОТКА И ИССЛЕДОВАНИЕ АЛГОРИТМА	
ДЕТЕКТИРОВАНИЯ ФИГУРЫ ЧЕЛОВЕКА	
3.1. Вводные замечания	
3.2. Реализация алгоритма вычисления НОG-признаков	
3.2.1. Вычисление градиента	
3.2.2. Группировка направлений	
3.2.3. Блоки дескрипторов	
3.2.4. Нормализация блоков	
3.2.5. Метод бегущего окна и НОG-детектор	
3.3. Алгоритм слежения за пешеходами на основе метода Лукаса-Канаде	
3.3.1. Точечные особенности	
3.3.2. Алгоритм Лукаса-Канаде	
3.4. Алгоритм Далала - Триггса	
3.5. Разработка модифицированного алгоритма	. 08

3.6. Оценка качества работы детектора фигуры человека	70
3.7. Результаты тестирования	73
3.8. Краткие выводы	
ГЛАВА 4. РАЗРАБОТКА АЛГОРИТМОВ ДЕТЕКТИРОВАНИЯ	
ЧЕЛОВЕКА НА ВИДЕОИЗОБРАЖЕНИЯХ С КУПОЛЬНЫХ КАМЕР	77
4.1. Вводные замечания	77
4.2. Разработка алгоритма на основе детектирования движения	80
4.3. Разработка алгоритма на основе контекста формы	83
4.4. Разработка алгоритма на основе HOG – признаков	85
4.5. Сравнительный анализ работы алгоритмов	86
4.6. Разработка алгоритма на основе бустинга	87
4.7. Разработка алгоритма на основе локальных бинарных шаблонов	88
4.8. Разработка алгоритм постклассификации ложноположительных	
результатов	91
4.9. Сравнение алгоритмов бустинга и ЛБШ с применением и без	
применения алгоритма постклассификации	92
4.10. Тестирование алгоритма на основе бустинга для спортивной	
видеоаналитики	93
4.11. Краткие выводы	96
ЗАКЛЮЧЕНИЕ	97
СПИСОК ЛИТЕРАТУРЫ	100
ПРИЛОЖЕНИЕ 1. АКТЫ ВНЕДРЕНИЯ РЕЗУЛЬТАТОВ РАБОТЫ	114

#### **ВВЕДЕНИЕ**

Актуальность темы. На современном этапе развития науки и техники задачи анализа видеоданных представляют важный теоретический и практический интерес, как один из важнейших механизмов обеспечения эффективного взаимодействия электронной техники с человеком [1–13]. С каждым годом появляется все больше камер видеонаблюдения и, соответственно, возрастает значимость решения задачи автоматического анализа видеопотока ДЛЯ систем охранного телевидения [14–22]. Одновременно с этим растет спрос на интеллектуальные системы, способные использовать подобный анализ для решения самых различных прикладных задач [18, 19]. К таким задачам относится, например, задача анализа аудитории, под которой понимается количественный и, в случае технической возможности, качественный (по полу, возрасту, расе, эмоциям и др.) анализ потока людей по одному или нескольким видеоизображениям, производимый в режиме реального времени [4-6].

Ключевой проблемой систем анализа аудитории является решение задачи обнаружения людей на видеоизображениях при различных ракурсах видеокамер. Целью решения такой задачи является определение наличия человека на видеопоследовательности и нахождение его положения. Алгоритмы, решающие задачу обнаружения людей, лежат в основе современных интерфейсов взаимодействия систем прикладного телевидения с человеком, находящих применение в следующих областях: охранное видеонаблюдение, робототехника, следящие системы, системы помощи водителю, системы спортивной видеоаналитики и т. д. [2, зависимости от ракурса камеры объектом интереса систем анализа аудитории может служить лицо человека (задача детектирования лиц), фигура человека (задача детектирования пешеходов) или верхняя часть головы (задача анализа видеоизображений с купольных (потолочных) камер). Разработке и исследованию алгоритмов для эффективного решения этих задач и посвящена настоящая диссертационная работа.

Проблема детектирования и локализации лиц на изображениях систем прикладного телевидения может быть сформулирована следующим образом: в данном кадре видеопоследовательности определить наличие или отсутствие лиц людей и при положительном ответе найти границы прямоугольных рамок, целиком включающих лица. Решение этой задачи осложняется рядом факторов [1–11]:

- изображение лица меняется в зависимости от текущих эмоций, возраста, наличия или отсутствия макияжа и волос на лице, условий освещения, угла поворота к камере, перекрытия части лица другими объектами;
- низкое разрешение видеоизображения в практических приложениях может привести к большому количеству ошибок как первого, так и второго рода;
- число лиц на изображении обычно неизвестно;
- детектирование лиц должно осуществляться в режиме реального времени, что накладывает жесткие условия на вычислительную сложность алгоритмов.

Среди существующих на сегодняшний момент алгоритмов детектирования лиц классическим считается подход на основе метода Виолы-Джонса [4, 11],ставший де-факто стандартом ДЛЯ МНОГИХ современных исследований. В настоящее время для решения практических задач исследуется множество других подходов к обнаружению лиц, среди которых можно выделить алгоритмы на основе гистограмм направленных градиентов, модели деформируемых частей, сверточных нейронных сетей, глубокого обучения и др. [4, 5]. Основной проблемой таких алгоритмов обычно является ИХ высокая вычислительная сложность, поэтому исследования в этой области остаются актуальными.

Задача детектирования фигуры человека также является практически востребованной в широком спектре приложений. Одной из наиболее актуальных и, в то же время, наиболее требовательных к качеству и скорости

работы подобных систем сфер применения является детектирование пешеходов на видеопотоке с камеры, установленной на движущемся автомобиле, в режиме реального времени [6]. При этом критичной является устойчивость к ложным срабатываниям системы, т. к. ее вмешательство в процесс управления транспортным средством должно происходить только в обоснованной необходимости. Существенные сложности решении данной задачи связаны с многообразием внешнего вида людей, окружающих предметов и их взаимного расположения, с тем, что детектирование должно производиться на видеоизображении с движущейся Следует отметить, что создание алгоритма детектирования пешеходов, обеспечивающего требуемый уровень качества, в настоящий момент остается открытой проблемой [4, 6].

Особый интерес представляют системы автоматического распознавания движений спортсменов для игровых видов спорта, таких как футбол, баскетбол, волейбол получения статистической информации, ДЛЯ предназначенной для анализа поведения отдельных игроков, команд, проведенных матчей [52-63]. Отличительной чертой таких систем является TOT факт, получения корректных результатов необходимо что ДЛЯ использовать видеоданные, исключающие сильные взаимные перекрытия объектов. Под данное условие отлично подходят видеоданные, полученные с помощью купольных камер. Задача детектирования головы человека при таком ракурсе остается малоизученной и представляет широкий научный и практический интерес [45].

В разработку алгоритмов цифровой обработки изображений внесли вклад как отечественные ученые — Ю.Б. Зубарев, М.И. Кривошеев, В.П. Дворкович, А.В. Дворкович, М.К. Чобану, А.С. Крылов, М.Н. Рычагов, Ю.С. Бехтин, Ю.С. Радченко, А.К. Бернюков, Д.С. Ватолин, С.В. Умняшкин, Е.П. Петров, И.С. Трубин, В.Ю. Волков, А.Л. Приоров, В.В. Хрящев, так и зарубежные — R. Chellappa, Z. Wang, A. Bovik, Y. Neuvo, J. Astola, T. Ojala, K. Egiazaryan, M. Nikolova, R. Szeliski, R. Lukac, T. Kanade [1-13].

Первые работы по распознаванию динамических объектов опубликованы еще в конце 1980-х гг. Следует отметить значительный вклад советских и российских ученых, занимающихся в данной области, таких как Ю.И. Журавлев, В.А. Сойфер, Е.В. Медведева, А.И. Галушкин, В.Н. Вапник, А.Я. Червоненкис, Б.А. Алпатов, М.Н. Фаворская, Ю.В. Визильтер, А.С. Конушин [7, 9, 10, 42].

Для решения практических задач детектирования человека при разных ракурсах камеры актуальным вопросом остается поиск алгоритмов, работающих в режиме реального времени и позволяющих идентифицировать человека в условиях наличия искажений и помех на телевизионных изображениях. Поэтому, на современном этапе развития науки и техники разработка и анализ алгоритмов детектирования человека для систем прикладного телевидения представляет собой актуальную научнотехническую задачу.

**Основной целью работы** является разработка и исследование новых алгоритмов детектирования человека на видеоизображениях в условиях наличия искажений и помех, обусловленных практическими требованиями для функционирования систем прикладного телевидения.

Объектом исследования являются алгоритмы детектирования человека на видеоизображениях, применяемые в радиотехнических системах обработки и анализа многомерных цифровых сигналов и в современных системах прикладного телевидения.

**Предметом исследования** являются разработка и модификация алгоритмов детектирования человека с целью повышения эффективности функционирования систем прикладного телевидения.

#### Задачи диссертационной работы

 разработка и исследование алгоритма детектирования лиц на видеоизображениях, характерных для систем прикладного телевидения, с

- дополнительными требованиями на скорость работы и робастность к основным видам искажений и помех;
- разработка и исследование алгоритма детектирования фигуры человека
   при различном качестве входного видеоизображения;
- разработка и исследование алгоритма детектирования головы человека на видеоизображениях, полученных с помощью купольных камер.

**Методы исследования.** При решении поставленных задач использовались современные методы цифровой обработки изображений, технического зрения, распознавания образов, машинного обучения, теории вероятностей и математической статистики. Для практической реализации алгоритмов применялись современные численные методы и методы программирования на языках С#, Python, MatLab.

#### Научная новизна полученных результатов

В рамках диссертационной работы получены следующие новые научные результаты:

- разработан алгоритм детектирования лиц на видеоизображениях с использованием ансамбля решающих деревьев;
- разработана модификация алгоритма Далала-Триггса с использованием классификатора на базе бустинга и переобучения на сложных примерах для детектирования пешеходов на видеоизображениях;
- разработан алгоритм детектирования головы человека на видеоизображениях, полученных с помощью купольных камер, с дополнительным классификатором на основе гистограмм направленных градиентов.

#### Практическая значимость полученных результатов

1. Разработана методика и программное обеспечение для исследования алгоритмов детектирования человека на телевизионных изображениях при различных ракурсах видеокамер.

- 2. Проведен анализ работы алгоритмов детектирования ЛИЦ телевизионных изображениях И предложен новый ансамбля использованием решающих деревьев, позволяющий улучшить комплексные характеристики системы в среднем на 13%.
- 3. Предложены, реализованы на языках высокого уровня И протестированы робастные алгоритмы детектирования головы человека вертикального ДЛЯ диагонального И ракурса камер, позволяющие повысить точность обнаружения человека по сравнению с известными подходами.

Разработанные алгоритмы требуют для своей практической реализации относительно небольших вычислительных ресурсов, что позволяет использовать их для обработки телевизионных изображений в системах реального или близкого к реальному режимах времени.

Результаты работы внедрены в соответствующие разработки ООО «Пиклаб» г. Ярославль и ООО «БАЙТ-ПРО» г. Ярославль. Отдельные результаты диссертационной работы внедрены в учебный процесс ЯрГУ им. П.Г. Демидова в рамках дисциплин «Цифровая обработка изображений», «Компьютерное зрение». Получено свидетельство о государственной регистрации программы для ЭВМ.

Достоверность полученных научных результатов обусловлена применением адекватного математического аппарата, подтверждается их согласованностью с результатами проведенного компьютерного моделирования и сопоставлением ряда полученных результатов с научными данными, известными из российской и зарубежной литературы.

**Апробация работы.** Результаты работы докладывались и обсуждались на следующих научно-технических конференциях:

- 13–17-я международные конференции «Цифровая обработка сигналов и ее применение» (DSPA), Москва, 2011–2015;

- 2-я всероссийская научная конференция молодых ученых с международным участием «Теория и практика системного анализа»,
   Рыбинск, 2012;
- 4–5-я всероссийские конференции «Радиоэлектронные средства получения, обработки и визуализации информации», Нижний Новгород, 2014; Москва, 2015;
- Международная конференция «Радиоэлектронные устройства и системы для инфокоммуникационных технологий РЭУС-2015»,
   Москва, 2015;
- 12-я международная научно-техническая конференция «Распознавание
   2015», Курск, 2015;
- 11-я международная научно-техническая конференция «Перспективные технологии в средствах передачи информации ПТСПИ-2015», Суздаль, 2015;
- 21-я международная научно-техническая конференция «Радиолокация, навигация, связь», Воронеж, 2015.

**Публикации.** По теме диссертации опубликовано 17 научных работ, из них 4 статьи в рецензируемых журналах из перечня ВАК и 13 докладов на научных конференциях различного уровня.

Структура и объем работы. Диссертация состоит из введения, четырех глав, заключения, списка литературы, содержащего 115 наименований, и одного приложения. Она изложена на 114 странице машинописного текста, содержит 29 рисунков и 18 таблиц.

#### Основные научные положения и результаты, выносимые на защиту

 Алгоритм детектирования лиц на видеоизображениях с использованием ансамбля решающих деревьев, увеличивающий площадь под ROC-кривой в среднем на 13% по сравнению со стандартным алгоритмом Виолы-Джонса.

- Модифицированный алгоритм Далала-Триггса с использованием классификатора на базе бустинга и переобучения на сложных примерах, позволяющий на 10–25% повысить точность детектирования фигуры человека на видеоизображениях.
- Алгоритм детектирования головы человека на видеоизображениях,
   полученных с помощью купольных камер, на основе бустинга с дополнительным классификатором на основе гистограмм направленных градиентов.

Благодарности. Автор выражает глубокую благодарность своему научному руководителю доктору технических наук Андрею Леонидовичу Приорову за помощь на всех этапах выполнения данной работы. Отдельная ЯрГУ В.А. Бондаренко, М.Л. Мячину, благодарность преподавателям А.Н. Морозову, А.А. Короткину, А.Ю. Левину, Г.М. Бродскому, работы которых оказали значительное влияние на формирование взглядов автора в данном научном направлении. Эти взгляды формировались также в совместной работе с коллегами по лаборатории «Цифровые цепи и сигналы» кафедры динамики электронных систем ЯрГУ, среди которых особенно хочется отметить профессора Юрия Александровича Брюханова и доцента Владимира Вячеславовича Хрящева, а также Ю. Лукашевича, В. Волохова, А. Шемякова, О. Степанову (Кисельникову), В. Павлова, Л. Шмаглита, А. Лебедева.

# ГЛАВА 1. СОВРЕМЕННЫЕ МЕТОДЫ, ЗАДАЧИ И АЛГОРИТМЫ В ОБЛАСТИ СИСТЕМ АНАЛИЗА АУДИТОРИИ

#### 1.1. Вводные замечания

В настоящее время в области систем видеоанализа достигнуты определённые успехи и наблюдается большое многообразие подходов к решению различных проблем. Одними из самых популярных задач являются задачи обнаружения человека и человеческого лица на видеоизображении. Такие задачи формируют различные требования к условиям получения видеоданных для анализа. Фронтальное изображение человека (рис. 1.1а) подходит для решения задач детектирования и распознавания лиц [42-44], но дает неудовлетворительные результаты в задаче подсчета числа людей в определенной области. Для подобных систем необходимо использовать видеоданные, исключающие сильные взаимные перекрытия объектов [45-47]. Этому условию удовлетворяют видеоданные полученные, с помощью купольных камер (рис. 1.16, в). При этом потенциально максимальная точность достигается при вертикальном расположении камеры (рис. 1.1в), что позволяет эффективно использовать видеопоток для решения ряда задач:

- обнаружение людей в стерильной зоне;
- обнаружение пересечения сигнальной линии;
- обнаружение оставленных объектов;
- обнаружение нетипичного поведения людей в определенной зоне;
- сбор статистических данных о различных объектах в сцене (подсчет количества объектов в определенной зоне) [45].



Рис. 1.1. Видеоданные, получаемые с камер, расположенных: (а) фронтально; (б) под углом; (в) вертикально

Однако для того чтобы охватить большую площадь сцены, камеры следует располагать на большей высоте либо использовать широкоугольные камеры. При этом имеется большая вариативность изображений людей, полученных с помощью купольных камер. В разных частях изображения люди могут выглядеть как снятые с вертикально расположенных камер (рис. 1.2в), а так же как снятые с камеры, расположенной под углом (рис. 1.2б). Данных факт сильно осложняет формирование универсального классификатора, способного обнаруживать людей в любой области видеоизображения.



Рис. 1.2. Иллюстрация вариативности изображений людей в кадре, полученном с помощью купольной камеры: (а) исходный кадр; (б) фрагмент кадра диагонального типа; (в) фрагмент кадра вертикального типа

Задача детектирования (обнаружения) человека на видеоизображениях заключается в поиске координат прямоугольников, окаймляющих положения всех объектов заданного класса. В связи с актуальностью и сложностью такой задачи в настоящий момент существует множество разнообразных подходов к ее решению [11]. Так, предпринимались попытки осуществить детектирование путем сравнения предобработанных изображений с некоторыми эталонами (шаблонами) с помощью специальной метрики [11]; использования детекторов и дескрипторов особых точек [23]; применения сегментации изображения [24-25] и т. д.

Одним из наиболее популярных и перспективных на настоящий момент подходов к детектированию объектов на изображении является, так называемый, метод бегущего окна. Он основан на осуществлении объектов определенного размера в пикселях: экстенсивного поиска последовательно рассматриваются области изображения заданного размера, им ставится в соответствие признаковое описание, на основе которого с помощью алгоритма классификации принимается решение, содержит ли данная область объект или нет. Более подробно данный подход и особенности его реализации рассматриваются ниже, здесь же, отметим, что ключевыми компонентами системы детектирования, основанной на таком подходе, являются алгоритмы получения признакового описания изображения и его классификации.

### 1.2. Признаковое описание изображений

Одним из наиболее популярных признаковых описаний является SURF (Speeded up Robust Features, высокопроизводительные робастные признаки) [27], который одновременно выполняет поиск особых точек и строит их описание, инвариантное к изменению масштаба и вращению. Кроме того, сам поиск ключевых точек обладает инвариантностью в том смысле, что повернутый объект сцены имеет тот же набор особых точек, что и образец. Ключевая точка изображения — точка, которая имеет некие признаки,

существенно отличающие ее от основной массы точек (резкие перепады освещенности, углы и т. д.). Ключевые точки изображения ищутся путем анализа определителя матрицы  $\Gamma$ ессе H для каждого пикселя изображения:

$$\det(H) = \frac{\partial^2 f}{\partial x^2} \cdot \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x \partial y}\right)^2. \tag{1.1}$$

Использование гессиана обеспечивает инвариантность относительно преобразования типа «поворот», но не инвариантность относительно изменения масштаба, поэтому SURF применяет фильтры разного масштаба для вычисления Гессиана. Предположим, что исходное изображение задается матрицей интенсивностей I, текущий рассматриваемый пиксель обозначим через X = (x, y), а  $\sigma$  – масштаб фильтра. Тогда матрица Гессе имеет вид:

$$H(X,\sigma) = \begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix},$$
 (1.2)

где  $L_{xx}(X,\sigma), L_{xy}(X,\sigma), L_{yy}(X,\sigma)$  — свертки аппроксимации второй производной Гауссова ядра с изображением I. Детерминант матрицы Гессе достигает экстремума в точках максимального изменения градиента яркости. Поэтому SURF проходит фильтром с Гауссовым ядром по всему изображению и находит точки, в которых достигается максимальное значение детерминанта матрицы Гессе. Отметим, что такой проход выделяет как темные пятна на белом фоне, так и светлые пятна на темном.

Далее для каждой найденной ключевой точки в ее окрестности ищется направление вектора максимального перепада функции интенсивности изображения вблизи данного пикселя с использованием фильтров Хаара. Вокруг ключевой точки описывается прямоугольная область размером 20S, где S — масштаб, на котором получено максимальное значение детерминанта матрицы Гесса. Эта область разбивается на 16 квадрантов, одинаковых размеров. Прямоугольная область затем поворачивается в соответствии с ориентацией ключевой точки.

На следующем шаге считаются оценки для каждого из 16-ти квадрантов области помощью фильтров Xaapa:  $\sum dx, \sum |dx|, \sum dy, \sum |dy|$  – суммарные градиенты по квадранту и сумма модулей точечных градиентов. Размеры прямоугольной области, а также размеры фильтров Хаара зависят от размера области взятия вторых производных. Дескриптор формируется в результате склеивания взвешенных описаний градиента для 16 квадрантов вокруг особой точки. Элементы дескриптора взвешиваются на коэффициенты Гауссова ядра. необходимы для большей устойчивости к шумам в удаленных точках. Дополнительно к дескриптору заносится след матрицы Гессе. Эти компоненты необходимы, чтобы различать темные и светлые пятна. Для светлых точек на темном фоне след отрицателен, для темных точек на светлом фоне – положителен. В результате получается вектор из 64-х чисел.

Отметим, что алгоритм SURF используется для поиска объектов. Тем не менее, дескриптор никак не использует информацию об объектах. Такой подход рассматривает изображение как единое целое и выделяет особенности всего изображения, поэтому он плохо работает с объектами простой формы.

Другим популярным способом признакового описания изображений является НОG-дескриптор (Histogram of Oriented Gradients, гистограмма ориентированных градиентов), изначально разработанный для решения задачи детектирования пешеходов [12] и в настоящий момент в том или ином виде использующийся в большинстве современных детекторов.

Основная идея, лежащая в основе НОG, заключается в том, что внешний вид и форма части объекта могут быть достаточно хорошо описаны распределением градиентов интенсивности пикселей, соответствующих данной части, без точной информации о градиентах в каждой точке. Под градиентом здесь понимается аппроксимация градиента функции интенсивности (яркости), которая предполагается дифференцируемой, но известной лишь в узлах равномерной сетки — пикселях, в заданной точке с

помощью некоторой разностной схемы. Алгоритм вычисления НОG-признаков за исключением некоторых моментов идентичен алгоритму вычисления SIFT-дескриптора [4, 15, 16]. Основным отличием НОG от SIFT является то, что SIFT-описание составляется для окрестности ключевой точки [4, 42], в то время как НОG-признаки покрывают все изображение (более того, с перекрытием).

Базовой единицей HOG-дескриптора является блок – прямоугольная область пикселей изображения заданных размеров. Блок состоит из ячеек, в свою очередь состоящих из пикселей. Каждой ячейке ставится в соответствие ориентаций (углов наклона относительно горизонтали) гистограмма градиентов из заданного количества полос, при этом направление считается «беззнаковым», т. е. наклон в  $\alpha$  и  $(2\pi-\alpha)$  считаются эквивалентными. Подобно SIFT-дескриптору амплитуда градиента в некотором пикселе дает вклад в полосы гистограммы ячейки, которой принадлежит данный пиксель, а также При ЭТОМ гистограммы соседних ячеек. используется интерполяция по углу наклона (полосам одной гистограммы), и билинейная по пространственному расположению (по гистограммам соседних ячеек). Также возможно взвешивание амплитуд градиентов с помощью гауссиана с центром, совпадающим с центром блока. После вычисления гистограмм в каждой ячейке блока, они конкатенируются, тем самым образуя вектор признаков блока. Полученный вектор подвергается нормализации. Такие признаковые описания вычисляются для всех блоков, не выходящих за пределы изображения, с координатами левого верхнего пикселя, кратными заданным шагам по вертикали и горизонтали. Причем данные шаги, как правило, задаются так, что блоки перекрываются, т. е. градиент пикселя учитывается при вычислении признаковых описаний нескольких блоков. HOG-описание изображения получается путем конкатенации векторов признаков всех блоков.

Еще одним популярным способом признакового описания являются локальные бинарные шаблоны (ЛБШ) [29] — простой оператор,

используемый для классификации текстур в компьютерном зрении. Впервые ЛБШ были описаны финскими учеными в 1996 году. Они представляют собой описание окрестности пикселя изображения в двоичной форме. Оператор ЛБШ, который применяется к пикселю изображения, использует восемь пикселей окрестности, принимая центральный пиксель в качестве порога. Пиксели, которые имеют значения больше, чем центральный пиксель (или равное ему), принимают значения «1», те же, которые меньше центрального, принимают значения «0». Таким образом, получается восьмиразрядный бинарный код, описывающий окрестность пикселя.

Локальный бинарный шаблон представляет собой фильтр, обозначаемый как LBP $_{P,R}(x,y)$ , где P — число точек, а R — радиус окрестности. Точки окрестности обозначим как  $g_i$ , где  $i=\overline{0,P-1}$ . При этом координаты точки рассчитываются как  $\left(R\cdot\cos\left(\frac{i}{2\pi}\right);R\cdot\sin\left(\frac{i}{2\pi}\right)\right)$ . Обозначим изображение как f(x,y) или f(g), если g точка. Пусть  $s_i(x,y)=\begin{cases} 1, & f(g_i)>f(g_c)\\ 0, & \text{иначе} \end{cases}$ , где  $g_c$  — точка с координатами (x,y). Тогда:

LBP<sub>P,R</sub>
$$(x,y) = \sum_{i=0}^{P-1} 2^i \cdot s_i(x,y).$$
 (1.3)

Таким образом, последовательность  $s_i(x,y)$ , где  $i=\overline{0,P-1}$ , представляет собой двоичную последовательность кода локального бинарного шаблона. Следовательно, LBP $_{P,R}(x,y)\in [0,2^P-1]$ . Для сравнения двух изображений в качестве векторов признаков используются гистограммы кодов LBP. В общем случае для каждого изображения строится гистограмма H(l) для значений LBP $_{P,R}(x,y)$ , где  $l\in \overline{0,P-1}$ . Существует несколько методов расчета расстояния между гистограммами. Например, расстояние хи-квадрат:

$$\chi^{2}(H_{1}, H_{2}) = \sum_{i=0}^{B-1} \frac{(H_{1}(i) - H_{2}(i))^{2}}{(H_{1}(i) + H_{2}(i))},$$
(1.4)

где B — число кодов. Существуют также модификации фильтра. Некоторые коды несут в себе больше информации, чем другие. Коды, в двоичной циклической записи которых число переходов между последовательностями «1» и «0» не превышает двух, считаются равномерными. Для заданного P существует  $P \cdot (P-1) + 2$  равномерных значений. Модифицированный фильтр  $LBP_{P,R}^{u2}$  в этом случае возвращает коды равномерных значений, добавляя только один код для неравномерных значений:

$$LBP_{P,R}^{u2} = \begin{cases} \text{индекс кода, если он равномерный} \\ P(P-1) + 2, \text{ иначе} \end{cases}$$
 (1.5)

Так как окрестность представляет собой круг, то можно найти группы кодов, инвариантных к повороту. Для каждого кода LBP существует P кодов, инвариантных к повороту, получаемых путем циклического сдвига P-битового числа. Для каждой такой группы в фильтр попадает минимальное значение кодов данной группы. Задача определения количества кодов, инвариантных к повороту, является нетривиальной. Фильтр обозначается как  $LBP_{P,R}^{ri}$ . С учетом предыдущих двух свойств определяется также равномерный фильтр, инвариантный к повороту. Кодов LBP, обладающих одновременно двумя свойствами, всего P+2, которые отличаются друг от друга числом бит, равных «1». В этом случае фильтр  $LBP_{P,R}^{ri2}$  задается следующим образом:

$$LBP_{P,R}^{ri2} = \begin{cases} \text{число единиц, если код равномерный} \\ P+1, \text{иначе} \end{cases}$$
 (1.6)

Для сравнения гистограмм объекта используются различные методы. Изображение объекта разбивается на  $k \times k$  участков, для каждого из которых рассчитывается гистограмма. Итоговая гистограмма изображения объекта определяется как конкатенация гистограмм участков изображения. Второй способ основан на взвешенной матрице: задается матрица весов  $k \times k$ , каждый элемент которой соответствует участку изображения. Обозначим

гистограмму j—го участка изображения как  $H^j$ , j = 0,  $k^2 - 1$ . Для каждого j-го участка задается вес  $w_j$ . Тогда можно определить модифицированное (взвешенное) расстояние хи-квадрат следующим образом:

$$\chi_w^2(H_1, H_2) = \sum_{j=0}^{k^2 - 1} w_j \cdot \sum_{i=0}^{B-1} \frac{\left(H_1^j(i) - H_2^j(i)\right)^2}{\left(H_1^j(i) + H_2^j(i)\right)},\tag{1.7}$$

где B — число кодов локального бинарного шаблона.

# 1.3. Методы и алгоритмы на основе машинного обучения 1.3.1. Решающие деревья

В сложных задачах распознавания образов, в которых требуется учитывать большое количество различных возможных признаков, полное сравнение неизвестного вектора признаков со многими векторами признаков эталонных образцов может занимать слишком много времени. Использование решающих деревьев решений позволяет чередовать этапы выделения признаков и классификации [30].

Решающее дерево [30, 51] используется для распознавания объектов, описываемых набором признаков. Каждой вершине дерева ставится в соответствие предикат, подразумевающий несколько вариантов ответов, соответствующих выходящим рёбрам. В зависимости от выбранного варианта ответа осуществляется переход к вершине следующего уровня. Концевым вершинам поставлены в соответствие метки, указывающие на отнесение распознаваемого объекта к одному из классов. Решающее дерево называется бинарным, если каждая внутренняя или корневая вершина инцидентна только двум выходящим рёбрам.

Для распознаваемого объекта проводится конечная последовательность сравнений значений его признаков с константами на равенство или неравенство, причём от результатов каждого сравнения зависит, что делать дальше: продолжать сравнивать или давать какой-либо ответ распознавания.

Если процедура обхода дерева решений достигает терминального узла (листа), то неизвестный вектор признаков классифицируется как элемент соответствующего класса образцов [51].

Процесс обучения дерева состоит в выборе его структуры, операций сравнения, сравниваемых признаков и порогов в каждом узле ветвления, и ответов в каждом терминальном узле. Обучение дерева решений относится к классу задач обучения с учителем, то есть обучающая и тестовая выборки содержат классифицированный набор примеров. В решающих деревьях разделение по категориальному признаку порождает столько дочерних узлов, сколько может принимать признак. С другой стороны, порядковые и количественные признаки определяют бинарное разделение: выбирается некоторое значение  $v_0$ , такое, что все объекты, для которых значение признака меньше или равно  $v_0$ , попадают в левый дочерний узел, а все Отсюда остальные правый. следует, ЧТО древовидные модели нечувствительны к монотонным преобразованиям шкалы признака, то есть они игнорируют шкалу количественных (непрерывных) признаков, трактуя их как порядковые. То же самое справедливо для моделей на основе правил. Для оценки признака с точки зрения разделения примеров на положительные и отрицательные используются следующие меры [49, 50]:

– энтропийный индекс неоднородности

$$\gamma_e(S) = -\sum_{i=1}^{L} P_i \ln P_i ,$$
 (1.8)

- индекс Джини

$$\gamma_g(S) = 1 - \sum_{i=1}^{L} P_i^2$$
, (1.9)

индекс ошибочной классификации

$$\gamma_m(S) = 1 - \max(P_i),$$
 $1,...,L$ 
(1.10)

где S — произвольная выборка, содержащая объекты из классов  $K_1, \dots, K_L, P_i$  — доля объектов класса  $K_i$  в выборке S. Данные индексы достигают минимального значения при принадлежности всех объектов обучающей выборки одному классу.

#### 1.3.2. Бустинг

Описанный ниже подход к детектированию лиц был предложен П. Виолой и М. Джонсом в работе [11]. Он использует процедуру обучения, основанную на бустинге [4]. Такой алгоритм состоит из трех этапов: переход к интегральному изображению, построение классификатора на основе бустинга, комбинирование классификаторов в каскадную структуру. Предлагаемый алгоритм использует три вида простых признаков. Значение двухпрямоугольного признака вычисляется как разность между суммами значений пикселей, принадлежащих двум прямоугольным областям. Области имеют одинаковую длину и ширину и ориентированы вертикально или горизонтально, как показано на рис. 1.3.

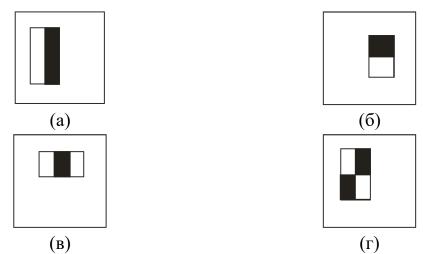


Рис. 1.3. Примеры прямоугольных признаков и их расположения внутри окна детектора: (а), (б) двухпрямоугольные признаки; (в) трехпрямоугольный признак; (г) четырехпрямоугольный признак

**Переход к интегральному изображению.** Интегральное изображение позволяет быстро вычислять признаки изображения, которые используются классификатором. Прямоугольные признаки могут быть посчитаны из

интегрального изображения (x, y), содержащего сумму пикселей, расположенных слева и сверху над этой точкой:

$$ii(x,y) = \sum_{x' \le x, y' \le y} i(x',y'),$$
 (1.11)

где ii(x,y) – интегральное изображение, i(x,y) – исходное изображение. Для вычисления используется следующая пара рекурсий:

$$s(x,y) = s(x,y-1) + i(x,y),$$
 (1.12)

$$ii(x, y) = ii(x - 1, y) + s(x, y),$$
 (1.13)

где s(x,y) – кумулятивная строка суммы, s(x,-1)=0, ii(-1,y)=0.

**Метод построения классификатора на основе алгоритма бустинга** Сложный классификатор можно рассматривать как суперпозицию простых классификаторов порогового типа. Обозначим простой классификатор как:

$$h_{j}(x) = \begin{cases} 1, \text{если } p_{j} f_{j}(x) < p_{j} \theta_{j}, & j = 1...N, \\ 0, \text{в остальных случаях} \end{cases}$$
 (1.14)

где  $p_j$  — показывает направление знака неравенства,  $\theta_j$  — значение порога,  $f_{i(x)}$  — вычисленное значение признака, x — входное изображение, N — общее количество признаков.

Рассмотрим алгоритм бустинга, с помощью которого осуществляется отбор простых классификаторов и формирование на их основе сложного классификатора. Зададим обучающий набор  $X_p = \{x_{p_1}...x_{p_L}\}$ , содержащий L изображений лиц, разрешением пикселя и обучающий набор  $X_p = \{x_{n_1}...x_{n_M}\}$ , содержащий M изображений «нелиц» того же разрешения. Пусть  $w_{l,i} = \frac{1}{2M}$ ,  $v_{l,i} = \frac{1}{2L}$ , где  $w_{l,i}$  и  $v_{l,i}$  — начальные веса для изображений обучающего набора «лиц» и «нелиц» соответственно. Тогда в цикле t = 1...T, где T — число отбираемых простых классификаторов, произведем следующие операции:

#### а) нормализация весов:

$$w_{t,i} \leftarrow \frac{w_{t,i}}{\sum_{j=1}^{M} w_{t,j} + \sum_{j=1}^{L} v_{t,j}}, \quad i = 1...M;$$
 (1.15)

$$v_{t,i} \leftarrow \frac{v_{t,i}}{\sum_{j=1}^{M} w_{t,j} + \sum_{j=1}^{L} v_{t,j}}, \quad i = 1...L;$$
 (1.16)

б) вычисление ошибки классификации:

$$\sigma_{t}(f, p, \theta) = \sum_{i=1}^{M} w_{t,i} h(x_{n_{i}}, f, p, \theta) + \sum_{i=1}^{L} v_{t,i} [1 - h(x_{pi}, f, p, \theta)]; \quad (1.17)$$

в) выбор классификатора с минимальной ошибкой:

$$h_t(x) = h(x, f_t, p_t, \theta_t),$$
 (1.18)

где  $f_t, p_t, \theta_t$  — параметры, при которых функция ошибки  $\sigma_t(f, p, \theta)$  принимает минимальное значение;

г) адаптивное обновление весов:

$$w_{t+1,i} \leftarrow w_{t,i} \beta_t^{1-h_t(x_{n_i}, f_t, p_t, \theta_t)}, \quad i = 1...M; \quad (1.19)$$

$$v_{t+1,i} \leftarrow v_{t,i} \beta_t^{1-h_t(x_{n_i}, f_t, p_t, \theta_t)}, \quad i = 1...L;$$
 (1.20)

где коэффициент обновления весов  $\beta_t$  определяется по следующей формуле:

$$\beta_t = \frac{\sigma_t(f_t, p_t, \theta_t)}{1 - \sigma_t(f_t, p_t, \theta_t)}.$$
(1.21)

В результате работы алгоритма бустинга на каждой итерации формируется простой классификатор, который имеет минимальную ошибку по отношению к текущим значениям весов, задействованных в процедуре обучения для определения ошибки. После процедуры обновления весов подчеркиваются те образцы, которые были неправильно классифицированы. Тем самым, на текущей итерации обучения следующий простой классификатор будет сформирован таким образом, чтобы не допустить ошибку на этих элементах обучающей выборки.

Отобранные простые классификаторы формируют сложный классификатор по следующему правилу:

$$C(x) = \begin{cases} 1, \text{если } \sum_{t=1}^{T} \alpha_t h_t(x) \ge Q \\ 0, \text{ в остальных случаях} \end{cases}$$
 (1.22)

где  $\alpha_t$  – коэффициенты линейной комбинации, а Q – оптимальный порог классификатора, определяемые по формулам:

$$\alpha_t = \ln \frac{1}{\beta_t}, Q = \frac{1}{2} \sum_{t=1}^{T} \alpha_t.$$
 (1.23)

#### Метод комбинирования классификаторов в каскадную структуру

Структура каскадного детектора приведена на рис. 1.4. Каскад состоит из слоев, которые представляют собой классификаторы, обученные с помощью процедуры бустинга [11].

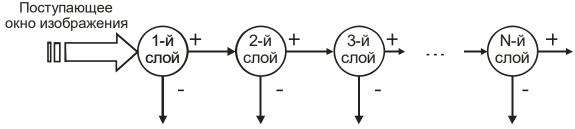


Рис. 1.4. Структура каскадного детектора

При конструировании каскада его структура определяется исходя из того, какие целевые требования к нему предъявляются. В результате обучения определяются следующие параметры каскада:

- количество слоев;
- количество простых классификаторов в каждом слое;
- пороговое значение каждого слоя.

Задаются требуемые значения целевых параметров для каждого слоя. Каждый слой формируется с помощью процедуры бустинга, в процессе которой количество простых классификаторов увеличивается до тех пор, пока не будут достигнуты требуемые значения коэффициентов. Во время

процедуры обучения данные разделяются на две выборки: обучающую и контрольную. Формирование простых классификаторов происходит с использованием обучающей выборки, а определение требуемых параметров – с использованием контрольной выборки.

#### 1.3.3. Машина опорных векторов

Метод МОВ (машина опорных векторов) [4] может рассматриваться как парадигма обучения классификаторов на основе полиномиальных функций, нейронных сетей или радиальных базисных функций. В то время как большинство методов обучения классификаторов (таких как байесовский классификатор или нейронные сети) основаны на минимизации ошибки обучения, т. е. эмпирического риска, МОВ работает на другом принципе, называемом минимизацией структурного риска, целью которого является минимизация верхней границы ожидаемой ошибки обобщения. МОВклассификатор – это линейный классификатор, для которого разделяющая гиперплоскость выбирается, исходя из минимизации ожидаемой ошибки классификации для тестовых изображений. Эта оптимальная гиперплоскость определяется как взвешенная комбинация небольшого поднабора обучающих называемых опорными векторами. Оценка оптимальной векторов, гиперплоскости эквивалентна решению линейно ограниченной задачи квадратичного программирования. Однако вычисления требуют больших затрат времени и ресурсов. В [32] Осуна и др. разработали эффективный метод обучения МОВ для широкого спектра задач и применили его для выделения лиц. Основываясь на двух тестовых наборах из 10 000 000 изображений размером 19×19 пикселей, их система имела достаточно низкий уровень ошибок и работала примерно в 30 раз быстрее системы Сена и Поджио [33]. Метод МОВ также был использован для выделения лиц и пешеходов в вейвлет-области [34-36].

В работах [37, 38] описан алгоритм на базе МОВ, использующий каскадную структуру. Этот алгоритм удобно разбить на три этапа:

#### Построение классификатора на базе метода опорных векторов

Применение метода опорных векторов [32] к задаче выделения людей поиске оптимальной гиперплоскости в заключается В признаковом пространстве, отделяющей класс изображений людей от изображений «нелюдей». В качестве признаков служат значения пикселей изображения, представленные в виде *т*-мерного вектора. Оптимальной считается гиперплоскость, которая максимизирует ширину полосы между классами (рис. 1.5). Разделяющая гиперплоскость определяется как линейная комбинация небольшого набора тренировочных векторов, называемых опорными векторами. Обозначим набор собственных векторов как  $\{X_1,...,X_m\}$ , а соответствующие им коэффициенты линейной комбинации –  $\{\alpha_1,...,\alpha_m\}$ . МОВ – это линейный классификатор, поэтому для разделения линейно неразделимых классов применяется неявное проецирование векторов-признаков в пространство потенциально намного более высокой размерности (еще выше, чем пространство изображений), в котором классы могут оказаться линейно разделимыми (рис. 1.6).

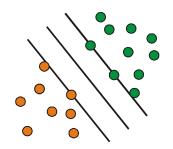


Рис. 1.5. Пример разделяющей полосы классификатора на базе МОВ

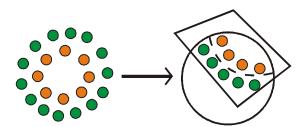


Рис. 1.6. Пример разделения линейно неразделимых классов с помощью перехода к пространству более высокой размерности

Проецирование осуществляется с помощью аппарата ядерных функций. Неявное проецирование с помощью ядерных функций не приводит к усложнению вычислений, что позволяет успешно использовать линейный классификатор для линейно неразделимых классов. Формально решающее правило классификатора на базе МОВ для входного изображения X может быть записано следующим образом:

$$f(x) = \operatorname{sgn}\left(\sum_{i=1}^{m} y_i \alpha_i k(X_i, X) + b\right), \tag{1.24}$$

где  $k(X_i,X)$  — это ядерная функция, а b — смещение. В качестве ядерной функции использовалась радиальная базисная функция Гаусса  $k(x,x') = \exp\left(-\frac{\|x-x'\|^2}{2\sigma^2}\right) \text{при } \sigma = 10.$ 

#### Вычисление аппроксимаций для классификатора MOB

Для ускорения принятия решения обученный классификатор на базе МОВ, аппроксимируется классификатором с меньшим набором опорных векторов  $\{Z_1,...,Z_{m'}\}$ , где m' < m. При этом быстродействие алгоритма увеличивается в  $\frac{m}{m'}$  раз. Решающее правило такого классификатора определяется по формуле

$$f'(X) = \operatorname{sgn}\left(\sum_{i=1}^{m'} \beta_i k(Z_i, X) + b\right). \tag{1.25}$$

Для нахождения  $\beta_i$  и  $Z_i$  необходимо решить оптимизационную задачу

$$\min \left\| \sum_{i=1}^{m} y_i \alpha_i k(X_i) - \sum_{i=1}^{m'} \beta_i k(Z_i) \right\|^2.$$
 (1.26)

На практике  $\beta_i$  и  $Z_i$  находятся при помощи градиентного метода оптимизации [19].

#### Комбинирование классификаторов в каскадную структуру

С целью увеличения быстродействия алгоритма и уменьшения ошибки классификации классификатор МОВ организуется в каскадную структуру, аналогичную каскадной структуре алгоритма на базе SNoW (рис. 1.7).

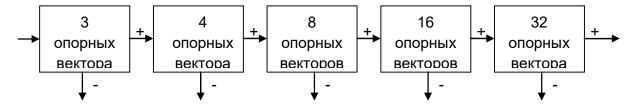


Рис. 1.7. Каскадная структура алгоритма на базе МОВ

Каскад состоит из 5 слоев, в которых классификатор МОВ аппроксимируется уменьшенным набором из 3, 4, 8, 16 и 32 опорных векторов соответственно.

#### 1.3.4. Нейронные сети

Так как выделение людей можно рассматривать как задачу распознавания шаблонов двух классов, для ее решения могут быть предложены нейронные сети [40] с различной архитектурой. Достоинством использования нейронных сетей для выделения людей является способность обученной системы охватить широкий класс условий, изменяющих внешний вид человека на изображении. Однако одним из недостатков является сложность настройки архитектуры сети (выбор числа слоев, числа узлов, целевых параметров и пр.) для получения оптимальных результатов.

Задача распознавания подразумевает умение нейронной сети быть устойчивой к небольшим сдвигам, поворотам и изменению масштаба изображения, т. е. она должна извлекать из данных некие инварианты. Решение этой задачи было найдено американским ученым французского происхождения Яном ЛеКуном [41], вдохновленным работами нобелевских лауреатов в области медицины Торстена Нильса Визеля и Дэвида Хантера Хьюбела. Принцип работы сверточных нейронных сетей заключается в чередовании сверточных слоев (C-layers, convolutional),

(S-layers, sub-sampling), субдискретизирующих слоев понижающих размерность карт признаков, и наличии полносвязных (F-layers, fullyвыходе, connected) слоев на играющих роль линейного МОВ-классификатора. Нейронные сети обучаются с помощью метода стохастического градиентного спуска c ручной настройкой гиперпараметров. Для предотвращения переобучения изображение случайно изменяется с помощью небольших преобразований (вращение, перемещение, масштабирование) перед подачей на сеть.

# 1.4. Особенности построения систем анализа аудитории в задачах спортивной видеоаналитики

Первоначально системы анализа аудитории фокусировались на оценке поведения отдельно взятого изолированного человека. Исследования по данной теме продолжаются уже много лет, и в последние годы извлечение информации для описания объекта уже не ограничивается особенностями формы и движения; используется также информация о пространственновременном контексте. Однако по-прежнему большая часть проводимых исследований фокусируется преимущественно на обработке отдельно взятого человека, а не групп людей.

В то же время, весьма часто поведение человека обусловлено влиянием окружающей действительности, например, взаимодействием с другими людьми и объектами или влиянием естественных границ сцены. Можно обобщить все эти факторы как локальный (пространственный и временной) контекст текущей сцены, оказывающий влияние на настоящее и будущее поведение человека. Структура такой зависимости хорошо проявляется для систем видеоанализа в командных видах спорта, таких как волейбол или баскетбол (необходимо учитывать амплуа игрока и его позиции на площадке). Предполагая известную геометрию игровой площадки как общую точку соприкосновения между кадрами видео, можно исследовать каждый видеокадр и описывать специфические индивидуальные действия отдельного

спортсмена с помощью применения гистограмм направленных градиентов для внешнего вида и гистограмм направленного потока (Histograms of Oriented Flow (HOF) [52]) для движения. Применение подобных гистограмм хорошо зарекомендовало себя в случаях, когда области объекта известны априори. Кроме того, целесообразно использовать информацию о местоположении наблюдаемого игрока. С помощью данных признаков, действия игрока классифицируются на основе метода опорных векторов. Результаты классификации для отдельно взятого игрока используются в качестве контекстной информации с помощью дескриптора контекста действий для описания вероятностей действий на площадке (рис. 1.8).



Рис. 1.8. Пример анализа кадра игроков на площадке

Разработаны различные виды дескрипторов, основанные на движении, форме, ключевых позах, моделях частей тела или траекториях ключевых

точек. Первые дескрипторы использовали детекторы ключевых точек для обнаружения простых признаков, таких как углы или края. Это, например, детектор углов Харриса, предложенный авторами [53], который позже был доработан до трехмерного случая в работе [54]. Позднее были разработаны дескрипторы, использующие признаки формы и движения (например, сидение, размахивание руками или специальные упражнения из аэробики). Широко используемый дескриптор на основе гистограмм направленных градиентов моделирует форму И часто используется совместно дескриптором на основе гистограмм оптического потока [52] для описания движения. Один из примеров детекторов ключевых точек – детектор на основе SIFT [55] и трехмерный SIFT-дескриптор [56]. После распознавания отдельных людей производится распознавание групп людей или даже толпы. Естественно, что отношения между этими людьми могут дать подсказку о действиях отдельно взятого человека. Это и является причиной для внедрения контекстных дескрипторов и для пространственного, и для временного измерений.

В работе [57] действия группы людей, снятых камерой видеонаблюдения, рассматриваются как совокупность действий отдельно взятого человека и поведения людей вокруг него. В работе [58] такие коллективные действия, как стояние в очереди, переход через дорогу, ожидание и общение, распознаются с помощью построения дескриптора пространственно-временного контекста, основанного на позиционировании и признаках движения около каждого человека в кадре. В работе [59] индивидуальные действия в сцене связываются для создания контекста действий. Движения на видео сегментируются (непрерывные движения разбиваются на сегменты действий), эти сегменты связываются в контексте между собой. Сегменты действий, связанные друг с другом в пространстве и времени, образуют группы действий. Сочетание пространственного и временного контекстов помогает различить действия.

Применение систем видеоанализа к хоккею на траве был представлен в [60]; матч был записан восемью камерами HD и игроки обеих команд были выделены с помощью вычитания фона и цветовых моделей. распознавания действий в баскетболе используются траектории игроков [61, 62]. Тренер изображает шаблонные схемы взаимодействия в защите и атаке для нескольких игроков на планшете, затем реальные траектории сравниваются с этими шаблонами. Подобно работе [60], авторы [63] используют специальные алгоритмы для распознавания вида спорта на Позиции спортивной арене. игроков, представленные гауссовыми распределениями, комбинируются по времени в карты интенсивности, которые коррелируют с различными видами спорта (бадминтон, баскетбол, футбол, волейбол и проч.). Все перечисленные системы распознавания действий в спорте распознают только действия команды, но не отдельных игроков. Разработка алгоритмов детектирования отдельного человека по лицу, контуру или голове сверху позволит проводить подобный анализ спортивных видеоданных более качественно.

#### 1.5. Краткие выводы

Результаты проведенного анализа существующих задач, методов и алгоритмов в области систем анализа аудитории позволяют сделать следующие основные выводы.

- 1. Большая вариативность в геометрическом расположении камер в системах прикладного телевидения не позволяет разработать универсального алгоритма анализа аудитории для детектирования и слежения за объектами.
- 2. На современном этапе разработаны и активно используются алгоритмы основе бустинга, локальных бинарных шаблонов, Их применение В решающих деревьев. системах прикладного практические телевидения открывает новые возможности ДЛЯ построения систем анализа видеоизображений.

3. Представляется необходимым проведение тестирования алгоритмов детектирования человека, построенных на базе глобальных и локальных признаков, в условиях наличия искажений и помех на изображении, характерных для систем прикладного телевидения.

Таким образом, проведенный анализ позволил сформулировать следующие основные задачи диссертации:

- разработка и исследование алгоритма детектирования лиц на видеоизображениях, характерных для систем прикладного телевидения, с дополнительными требованиями на скорость работы и робастность к основным видам искажений и помех;
- разработка и исследование алгоритма детектирования фигуры человека
   при различном качестве входного видеоизображения;
- разработка и исследование алгоритма детектирования головы человека на видеоизображениях, полученных с помощью купольных камер.

# ГЛАВА 2. РАЗРАБОТКА И ИССЛЕДОВАНИЕ АЛГОРИТМА ДЕТЕКТИРОВАНИЯ ЛИЦ НА ОСНОВЕ КАСКАДА АНСАМБЛЕЙ РЕШАЮЩИХ ДЕРЕВЬЕВ

#### 2.1. Вводные замечания

График рабочей характеристики приемника или ROC-кривая (Receiver Operating Curve) [74] — это метод для визуализации, организации и выбора классификаторов на основе их эффективности. ROC-кривые долгое время теории обнаружения сигналов, чтобы использовались В компромисс между вероятностью правильного обнаружения сигнала и вероятностью ложного срабатывания классификатора. В последние годы наблюдается рост в использовании ROC-кривых в машинном обучении, отчасти из-за осознания того, что ROC-кривые обладают свойствами, которые позволяют сделать их особенно полезными в исследованиях, где обучение ошибок, чувствительно К стоимости И присутствуют несбалансированные классы.

Так как модель классификатора – это отображение примеров в предсказанные классы, TO возможны два вида выходных данных классификатора – дискретные и непрерывные. Классификаторы, выдающие непрерывные выходные данные (например, оценку вероятности в случае принадлежности примера к классу), являются непрерывными. К ним можно применить различные пороги или точки отсечения для того чтобы определить, к какому классу принадлежит пример. Варьируя данный параметр, получают то или иное разбиение на два класса и различные величины ошибок I и II рода.

Модели, выходными данными которых являются дискретные отметки классов, показывают только предсказанный класс для данного примера и называются дискретными. Обозначим фактические классы как  $\{p;n\}$ , а предсказанные классификатором –  $\{Y;N\}$ , и построим матрицу

сопряженности, представленную в виде табл. 2.1. Основной принцип построения матрицы сопряженности основан на сравнении предсказаний модели по тестовой выборке и истинных значений интересующей переменной. Столбцы матрицы отражают действительные классы тестовой выборки, а строки — предсказанные моделью классы. Эта матрица является основой для многих распространенных метрик.

Таблица 2.1 Матрица сопряженности

	Фактические классы		
Модель	p (+)	n (-)	
Y (+)	True Positives	False Positives	
N (-)	False Negatives	True Negatives	

При анализе чаще оперируют не абсолютными показателями, а относительными — долями (rates). В целях избегания смысловых ошибок, вызванных различными вариантами перевода терминов, будем использовать их английские аналоги. Доля истинно положительных примеров или чувствительность алгоритма классификации (true positive rate, hit rate, recall, sensitivity) классификатора вычисляется как:

TruePositiveRate(TPR) = 
$$\frac{TP}{TP + FN} = \frac{TP}{P} = Sensitivity = Recall.$$
 (2.1)

Доля ложно-положительных примеров (false positive rate, false alarm rate) вычисляется как:

FalsePositiveRate(FPR) = 
$$\frac{FP}{FP + TN} = \frac{FP}{N} = 1 - \text{Specificity}.$$
 (2.2)

Специфичность модели – доля истинно отрицательных примеров, которые были верно идентифицированы моделью:

Specificity = 
$$\frac{TN}{TN + FP} = \frac{TN}{N}$$
. (2.3)

Точность, доля верно предсказанных моделью значений (precision, positive predicted value, PPV) вычисляется как:

$$Precision = \frac{TP}{TP + FP}.$$
 (2.4)

Показатель точности (accuracy) – доля всех верно классифицированных положительных и отрицательных примеров от числа всех примеров:

Accuracy = 
$$\frac{TP + TN}{(TP + FN) + (FP + TN)} = \frac{P}{P + N}.$$
 (2.5)

F-мера (F-value) – среднее гармоническое величин Recall и Precision:

$$F-value = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} = \frac{2 \cdot \frac{TP}{TP + FN} \cdot \frac{TP}{TP + FP}}{\frac{TP}{TP + FN} + \frac{TP}{TP + FP}} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}. (2.6)$$

ROC-кривая является двумерным графиком, где по оси Y откладывается значение TPR, а по X – FPR. Она показывает относительный компромисс между выгодой (истинно-положительные значения) и затратами (ложно-положительные значения).

Для сравнения дискретных и непрерывных классификаторов, существуют различные методы преобразования их выходных данных. С помощью варьирования порога можно получить из непрерывного классификатора дискретный. Для дискретного классификатора на основе его бинарных решений можно построить такие значения, которые будут аналогичны непрерывным значениям вероятности:

- 1) уверенность в правиле принятия решения о принадлежности изображения к определенному классу;
  - 2) объединение меток классов;
- 3) использование бэггинга (bagging, bootstrap aggregation) [75] для создания ансамбля дискретных классификаторов, каждый из которых имеет свой «голос», этот набор голосов можно использовать для генерации непрерывного значения;
  - 4) совместное использование оценки вероятности и голосования.

Анализ современных алгоритмов детектирования лиц с использованием ROC-кривых показывает [74], что несмотря на значительный отрыв по уровню верных обнаружений, лидирующие алгоритмы [65] показывают низкие результаты по скорости обработки изображений, в связи с чем для дальнейших исследований был выбран алгоритм на основе ансамблей решающих деревьев (АРД), способный быстро работать на устройствах с ограниченной мощностью.

# 2.2. Алгоритм детектирования на основе каскада ансамблей решающих деревьев

Алгоритм АРД является модификацией стандартного алгоритма детектирования Виолы-Джонса, с применением сканирования изображения бинарных классификаторов. Область изображения каскадом классифицируется как объект интереса, если она успешно проходит через все ступени каскада. Каждый бинарный классификатор состоит из ансамбля решающих деревьев, которые сравнивают интенсивность пикселей в качестве бинарного теста на своих внутренних узлах. Процесс обучения состоит из процедуры построения дерева регрессии на основе жадного алгоритма. Большинство современных алгоритмов построения деревьев регрессии являются жадными. Такие алгоритмы осуществляют построение деревьев сверху вниз путем рекурсивного разделения обучающих данных и кратко могут быть описаны следующим образом:

- выбор лучшего разделения (того, которое обеспечивает экстремум некоторого критерия);
- разделение исходных данных на подмножества;
- рекурсивное применение данной процедуры для каждого из выделенных подмножеств.

Жадные алгоритмы обладают низкой трудоемкостью, хорошо масштабируемы, но имеют ряд недостатков: а) дерево регрессии строится постепенно без возврата к ранее принятым решениям; б) на каждом шаге

работы алгоритма принимается локально оптимальное решение, то есть решение, дающее максимальный эффект на текущем шаге, без учета его влияния на всё решение в целом. Поэтому они приводят, как правило, к неоптимальному разделению данных.

Для решения задачи регрессии на основе изображения будем применять оптимизированные бинарные решающие деревья, использующие сравнение интенсивности пикселей в качестве бинарных тестов в своих внутренних узлах. Этот подход был предложен Яли Амитом и Дональдом Джеманом в [66], и позднее успешно использован исследователями и инженерами (например, в [67-69]). Бинарный тест по сравнению интенсивности для изображения I определяется как

$$bintest(I; l_1, l_2) = \begin{cases} 0, I(l_1) \le I(l_2) \\ 1, \text{ иначе} \end{cases}, \tag{2.7}$$

где  $I(l_i)$  — интенсивность пикселя в точке  $l_i$ .  $l_1$  и  $l_2$  являются нормированными координатами из набора  $[-1;+1] \times [-1;+1]$ . Это означает, что бинарные тесты могут быть легко модифицированы, если потребуется. Каждый терминальный узел дерева содержит скаляр, который моделирует выходную величину.

Построение дерева контролируется. Обучающий набор данных — это набор  $\{(I_s, v_s, w_s): s=1,2,...,S\}$ , где  $v_s$ — контрольные данные, связанные с изображением  $I_s$ , а  $w_s$  — фактор важности (вес). К примеру, в случае бинарной классификации, контрольные данные имеют две метки класса: положительные примеры аннотированы +1, а отрицательные -1. Веса  $w_s$  позволяют присвоить каждому из этих примеров различную степень важности. Бинарный тест в каждом внутреннем узле дерева выбирается так, чтобы минимизировать взвешенную среднеквадратическую ошибку, полученную, когда входные данные были разделены тестом. Это достигается минимизацией следующей величины:

$$WMSE = \sum_{(I,v,w)\in C_0} w \cdot (v - \bar{v}_o)^2 + \sum_{(I,v,w)\in C_1} w \cdot (v - \bar{v}_1)^2, \tag{2.8}$$

где  $C_0$  и  $C_1$  – группы обучающих образцов, для которых результаты бинарного теста на изображении равны 0 и 1 соответственно. Скаляры  $\bar{v}_o$  и  $\bar{v}_1$  – взвешенное среднее контрольных данных в  $C_0$  и  $C_1$  соответственно. Так как количество сравнений интенсивности пикселей очень большое, то во время оптимизации на каждом внутреннем узле создается лишь небольшая часть путем многократной выборки двух координат ИЗ равномерного распределения на квадрате  $[-1, +1] \times [-1, +1]$ . Выбирается тот тест, который имеет наименьшую ошибку в соответствии с выражением (2.8). Обучающие данные рекурсивно объединяются в группы таким образом до тех пор, пока не выполняется прекращающее условие. Глубина деревьев ограничена для уменьшения времени обучения, увеличения скорости обработки и в соответствии с требованиями памяти. Выходное значение, связанное с каждым терминальным узлом есть взвешенное среднее контрольных данных, полученное в процессе обучения.

Если ограничить глубину деревьев до D и считать B бинарных тестов в каждом внутреннем узле, то время обучения это  $O(D \cdot B \cdot S)$  для обучающего набора, содержащего S образцов. Каждый обучающий образец тестируется с B сравнениями интенсивности пикселей для каждого внутреннего узла, которые он проходит на пути длиной D от корневого узла до терминального. Построенным деревьям требуется  $O(2^D)$  байт для хранения и скорость их работы пропорциональна O(D).

Одно решающее дерево, как правило, достигает умеренную точность. С другой стороны, ансамбль деревьев может достичь впечатляющих результатов. Для создания дискриминантного ансамбля с помощью последовательной подстройки решающего дерева для соответствующей задачи наименьших квадратов будем использовать алгоритм GentleBoost [70], являющийся модификацией хорошо известного AdaBoost. Для того, чтобы сгенерировать ансамбль из K деревьев из обучающего набора  $\{(I_s, c_s): s = 1, 2, ..., S\}$ , алгоритм проходит следующие шаги:

1. Задание начальных весов  $w_s$  для каждого изображения  $I_s$  и его отметки класса  $c_s \in \{-1; +1\}$  в виде:

$$w_s = \begin{cases} 1/_P, & c_s = +1\\ 1/_N, & c_s = -1 \end{cases}$$
 (2.9)

где P — общее количество положительных образцов, а N — общее количество отрицательных образцов.

- 2. Для k = 1, 2, ..., K:
  - а) установить решающее дерево  $T_k$  с помощью взвешенных наименьших квадратов  $c_s$  для изображения  $I_s$  с весом  $w_s$
  - б) обновление весов:

$$w_s = w_s \exp(-c_s T_k(I_s)), \tag{2.10}$$

где  $T_k(I_S)$  обозначает вещественный выход дерева  $T_k$  для изображения  $I_S$ .

- в) перенормировка весов так, чтобы их сумма была равна 1.
- 3. Выходной ансамбль  $\{T_k: k = 1, 2, ..., K\}$ .

Во время работы выходы всех деревьев в ансамбле суммируются, и получение значение проходит через порог для получения метки класса. Уровень детектирования каждой ступени детекторов регулируется путем изменения выходного порога ансамбля. Каждая ступень использует мягкий выход («доверительный») с предыдущей ступени, то есть использует выходную информацию в качестве дополнительной для улучшения характеристик. Таким образом, осуществляется постепенное накопление выходов ступеней классификации в каскаде.

Так как получившийся детектор устойчив к малым изменениям в позиции и масштабе, вокруг каждой области интереса может быть несколько рамок. Эти перекрывающиеся обнаружения объединяются в результате постобработки. Два обнаружения объединяются, если перекрытие между ними составляет более 30%:

$$\frac{D_1 \cap D_2}{D_1 \cup D_2} > 0.3. \tag{2.11}$$

Блок-схема описанного выше алгоритма приведена на рис. 2.1.



Рис. 2.1 Блок-схема алгоритма АРД

## 2.3. Обучение детектора

Виола Джонс сделали обнаружение объектов практически осуществимым в реальных приложениях. Это связано с тем, что системы, основанные на их алгоритме, могут обрабатывать изображения намного быстрее, предыдущие сопоставимом чем подходы, при уровне детектирования. В связи с развитием широкого спектра мобильных устройств

с ограниченной мощностью процессоров, производители заинтересованы в развитии более быстрых алгоритмов детектирования. Чтобы система работала в таких приложениях, разработчики готовы пожертвовать точностью детектирования для лучших скоростей обработки. Алгоритм АРД позволяет обрабатывать изображения и видео с высокой скоростью, сохраняя при этом сравнительную точность, предоставляя возможность переобучения алгоритма на необходимых наборах данных, а также возможностью построения на его основе каскада классификаторов для детектирования лиц, повернутых под разными углами относительно вертикальной оси. Алгоритм инвариантен к вращению изображения в плоскости экрана за счет использования при обучении множества копий оригинального изображения, повернутых на углы, равномерно выбранные из интервала  $[0; 2\pi)$ , а также небольших сдвигов.

Процесс обучения детектора подразумевает наличие двух наборов данных: набора, содержащего положительные примеры – лица, и набора, состоящего отрицательных примеров, есть изображений, ИЗ TO содержащих лиц. Для обучения фронтального детектора использовалась база GENKI, содержащая 3500 аннотированных лиц. С целью повышения обобшающей способности алгоритмов на классе ЛИЦ К исходным изображениям из базы применялась серия трансформаций. Из каждого изображения лица генерируется 15 положительных обучающих образцов с помощью небольших случайных изменений в позиции и масштабе лица, это делает детектор более устойчивым к шумам. Для обучения использовалось 300 000 отрицательных образцов, не содержащих лица.

Параметры процесса обучения устанавливаются заранее. Глубина каждого дерева фиксируется равной 6 и используется 20 ступеней классификации. Каждый этап классификации имеет предопределенное количество деревьев и уровень детектирования. Во время оптимизации для каждого внутреннего узла дерева проводится 256 бинарных тестов. Процесс оптимизации значительно улучшает производительность каскада.

Для обучения детектора, выделяющего лица, повернутые влево на угол  $30^{\circ} - 60^{\circ}$ , был составлен новый обучающий набор, состоящий из изображений базы лиц AFLW [71]. Набор содержит 2966 изображений, с аннотированием рамкой и 4 ключевыми точками (центры глаз, кончик носа, центр рта), полученным посредством работы в среде по аннотированию лиц. Отрицательные примеры для обучения аналогичны примерам для обучения фронтального детектора. К данному набору также были применены трансформации: поворот в плоскости на углы  $\pm 5^{\circ}$ ,  $\pm 10^{\circ}$ , сдвиг изображения  $\pm 2.5\%, \pm 5\%,$ изменение В масштабе  $\pm 5\%$ . В результате на трансформирования на основе одного изображения было получено 10 положительных обучающих образцов, то есть для обучения использовалось 29660 изображений.

Для обучения фронтального детектора и детектора left 30-60, использовался ПК со следующими параметрами: процессор AMD FX(tm)-8320 @ 3.5 ГГЦ, количество ядер -8, объем ОЗУ – 16Гб. Использование столь мощных персональных компьютеров позволило провести обучение примерно за 50 часов, что приемлемо для перенастройки детектора для конкретных практических приложений.

### 2.4. Тестирование модификаций детектора

Тестирование и анализ характеристик полученных детекторов (фронтальный, с поворотом влево, с поворотом вправо) проводился на базе изображений Robotics [72], содержащей 6623 изображения 90 людей, головы которых повернуты относительно вертикальной оси на углы в интервале [-90°; +90°] с шагом в 5° и на базе изображений AFW [73] состоящей из 205 изображений, содержащих 468 лиц разных масштабов и повернутых на различные углы относительно вертикальной оси. В результате тестирования алгоритмов была получена сумма верных обнаружений (ВО) в зависимости от угла поворота (рис. 2.2).

Из приведенных зависимостей видно, что максимальные значения верного детектирования для фронтального модуля АРД приходятся на углы от  $-30^{\circ}$  до  $30^{\circ}$ , а для детектора, обученного на углы  $[-30^{\circ}; -60^{\circ}]$  и [30°; 60°], максимальные значения приходятся именно на эти интервалы углов. Так же можно заметить, что детекторы срабатывают и на лица, которые находятся в противоположных интервалах углов относительно интервала углов поворота обучающей выборки. Исходя из этого было выдвинуто предположение о том, что аннотация базы AFLW имеет значительное количество ошибок. Была произведена ручная пересортировка данной базы для боковых детекторов лиц, после чего модули были переобучены И протестированы повторно. Результаты тестирования представлены на рис. 2.3.

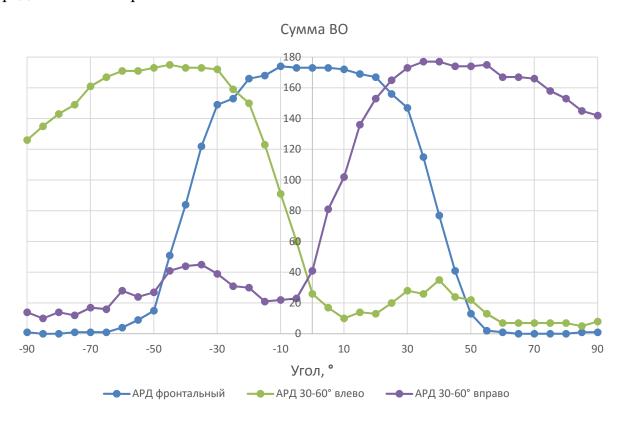


Рис. 2.2. Сумма верных обнаружений при различных углах поворота лиц для различных модулей АРД

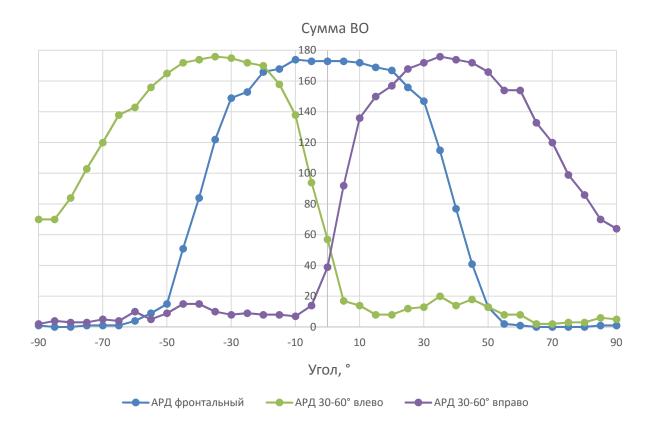


Рис. 2.3. Сумма верных обнаружений при различных углах поворота лиц для различных модулей АРД после пересортировки базы

Из рис. 2.3 видно, что после пересортировки интервалы работы модулей стали лучше соответствовать заданным интервалам при обучении, что подтверждает предположении о содержании ошибок в аннотации углов поворота лиц базы AFLW. Количество ошибочных обнаружений также снизилось.

На рис. 2.4 приводятся ROC-кривые для сравнения различных модулей детектора на основе APД. Площадь под ROC-кривой равна соответственно: APД фронтальный = 0,932, APД 30-60 влево = 0,856, APД вправо 30-60 = 0,852.

Для проведения дальнейших тестирований был построен также объединенный классификатор, включающий в себя фронтальные и повернутые лица (далее именуется просто АРД).

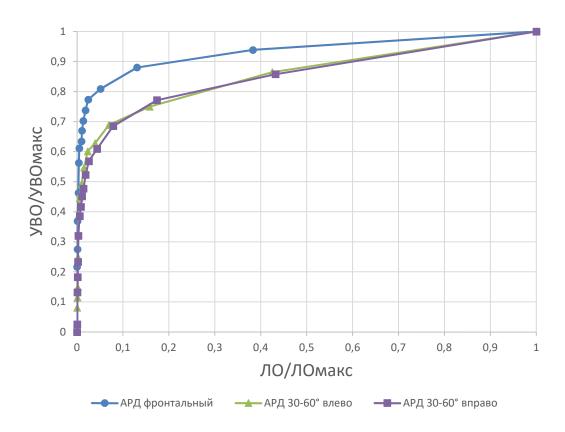


Рис. 2.4. ROC-кривые для различных модулей АРД

На рис. 2.5 приводятся ROC-кривые для сравнения различных алгоритмов детектирования лиц, полученные на тестовой базе AFW. Площадь под ROC-кривой равна соответственно: APД фронтальный = 0,932, Бустинг = 0,83, APД (фронтальный+поворот)= 0,956. Таким образом, показано, что предложенный объединенный алгоритм APД увеличивает площадь под ROC-кривой на 13% по сравнению со стандартным подходом на основе бустинга (алгоритм Виолы-Джонса). Проверим этот алгоритм на устойчивость к основным моделям помех и искажений, встречающихся в системах прикладного телевидения.

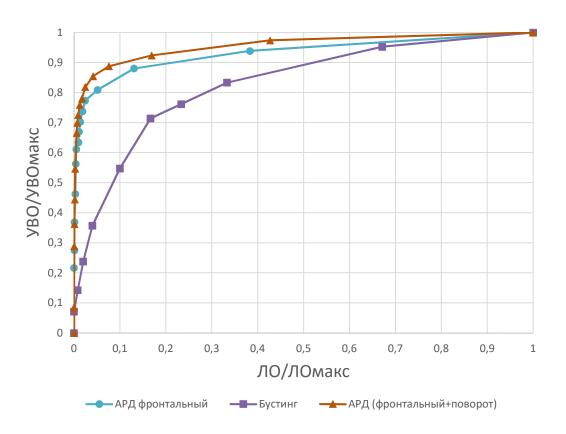


Рис. 2.5. ROC-кривые для различных алгоритмов детектирования лиц

# 2.5. Тестирование алгоритма детектирования в условиях присутствия искажений и помех на изображениях

Для оценки робастности алгоритма детектирования лиц на основе АРД были смоделированы ситуации искажений изображения, характерных для систем прикладного телевидения: размытие, присутствие аддитивного белого гауссовского шума, импульсного шума, мультипликативного шума, сжатие алгоритмами JPEG и JPEG2000. Результаты исследования приведены на рис. 2.6-2.11.

Размытие (рис. 2.6) моделировалось обработкой изображения линейным низкочастотным усредняющим фильтром с масками 20, 30 и 40. Уровень размытия определялся средним значением стандартной метрики PSNR по всей используемой базе изображений. Для маски 20 значение метрики составило PSNR=24,33 дБ, для маски 30 - PSNR=22,68 дБ, для маски 40 - PSNR=21,58 дБ. Площадь под ROC-кривой по результатам исследований

составила: без шума - 0,942, маска 20 - 0,903, маска 30 - 0,877, маска 40 - 0,872. Таким образом, можно сделать вывод, что внесение размытия в тестовый набор изображений уменьшает площадь под ROC-кривой для алгоритма APД на 3,5% при снижении метрики PSNR до 21,58дБ.

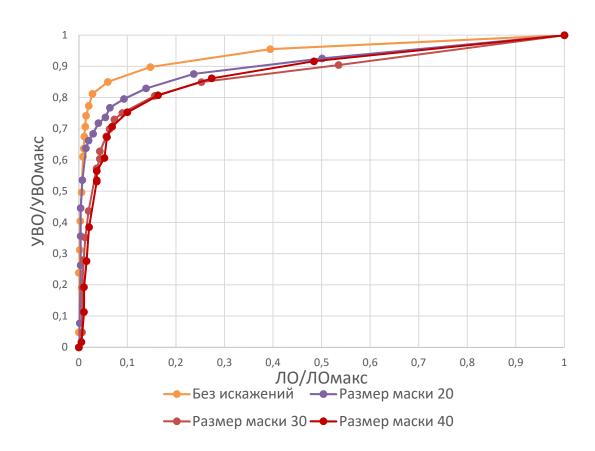


Рис. 2.6. ROC-кривые для искажения размытием

АБГШ вносился в набор тестовых изображений с заданным уровнем СКО – сигма (15, 25, 35). При этом среднее значение PSNR для тестовой базы зашумленных изображений составило: PSNR=24,94 дБ, PSNR=20,71 дБ и PSNR=17,99 дБ, соответственно (рис.2.7). Площадь под ROC-кривой по результатам исследований составила: без шума - 0,942, сигма 15 - 0,894, сигма 25 - 0,794, сигма 35 - 0,752. Таким образом, можно сделать вывод, что внесение АБГШ в тестовый набор изображений оказывает более существенное воздействие на детектор лиц и уменьшает площадь под ROC-кривой для алгоритма АРД на 15,8% при снижении метрики PSNR до 20,71 дБ и на 21,2% при снижении метрики PSNR до 17,99 дБ.

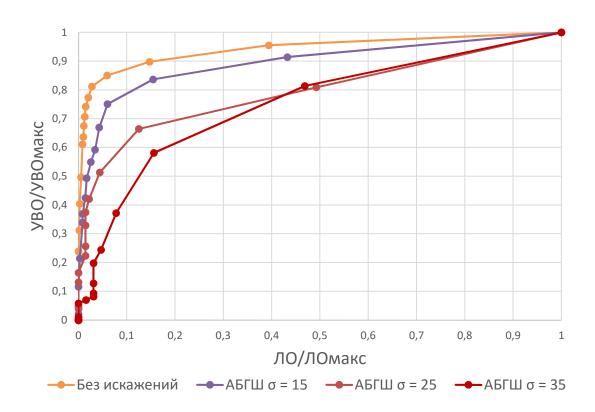


Рис. 2.7. ROC-кривые для искажения АБГШ

Импульсный шум вносился в набор тестовых изображений с заданной вероятностью появления импульсов типа salt-and-pepper — 0,02, 0,04, 0,08. При этом среднее значение PSNR для тестовой базы зашумленных изображений составило: PSNR=21,73 дБ, PSNR=18,73 дБ и PSNR=15,71 дБ, соответственно (рис.2.8). Площадь под ROC-кривой по результатам исследований составила: без шума - 0,942, вероятность 0,02 - 0,92, вероятность 0,04 - 0,88, вероятность 0,08 - 0,86. Таким образом, можно сделать вывод, что внесение импульсного шума в тестовый набор изображений оказывает существенное воздействие на детектор лиц и уменьшает площадь под ROC-кривой для алгоритма АРД на 8,8% при снижении метрики PSNR до 15,71 дБ.

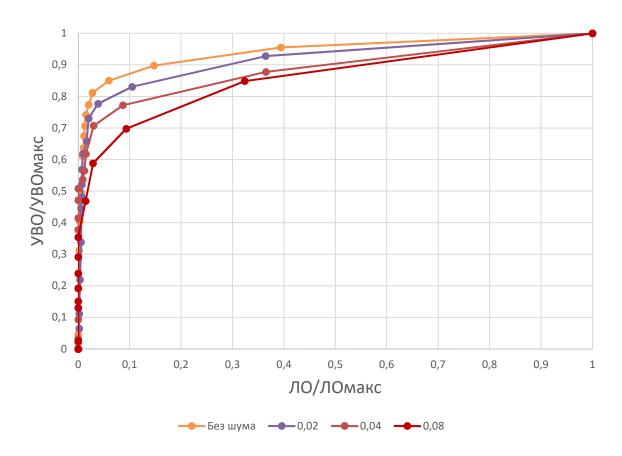


Рис. 2.8. ROC-кривые для искажения импульсным шумом

Мультипликативный шум вносился в набор тестовых изображений с параметром — 0,02, 0,04, 0,08. При этом среднее значение PSNR для искаженной тестовой базы изображений составило: PSNR=24,49 дБ, PSNR=21,59 дБ и PSNR=18,75 дБ, соответственно (рис.2.9). Площадь под ROC-кривой по результатам исследований составила: без шума - 0,942, вероятность 0,02 — 0,905, вероятность 0,04 - 0,813, вероятность 0,08 - 0,769. Таким образом, можно сделать вывод, что внесение мультипликативного шума в тестовый набор изображений оказывает относительно слабое воздействие на детектор лиц и уменьшает площадь под ROC-кривой для алгоритма АРД на 18,4% при снижении метрики PSNR до 18,75 дБ.

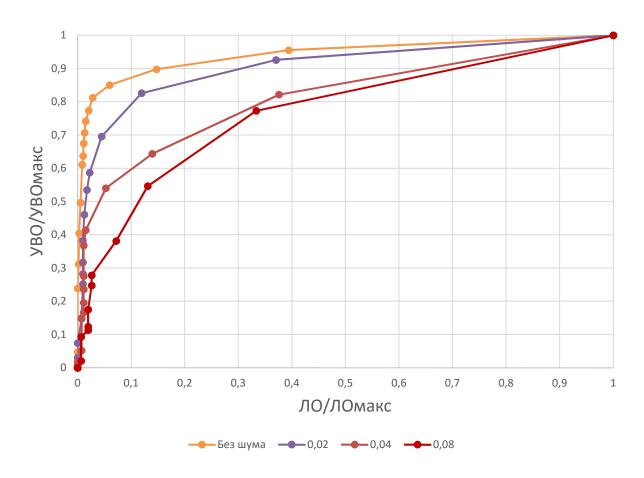


Рис. 2.9. ROC-кривые для искажения мультипликативным шумом

Проводилось сжатие тестового набора изображений алгоритмом Baseline JPEG (рис. 2.10). Уровень артефактов сжатия определялся внутренним параметром качества алгоритма, а также оценивался с помощью усреднённой метрики PSNR. Для получившихся наборов сжатых изображений показатели этой метрики составили: PSNR=35,21 дБ для качества 20, PSNR=32,17 дБ для качества 10, PSNR=24,73 дБ для качества 5. Площадь под ROC-кривой по результатам исследований составила: без шума = 0,942, качество 20 = 0,941, качество 10 = 0,930, качество 5 = 0,872. Таким образом, слабое сжатие алгоритмом JPEG оказывает незначительное влияние на работы детектора. Увеличение степени сжатия до PSNR=24,73 дБ понижает площадь под ROC-кривой на 7,5%.

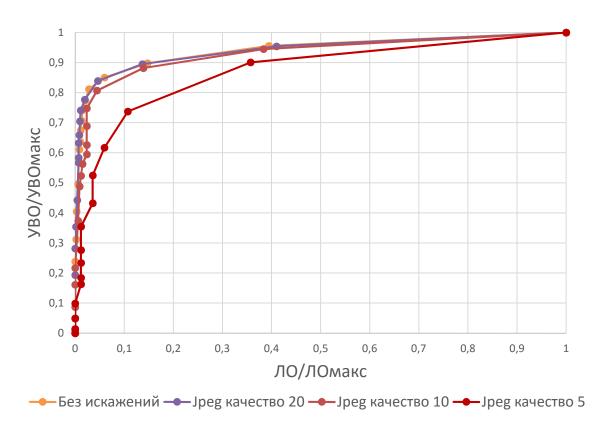


Рис. 2.10. ROC-кривые для сжатия JPEG

Так же было проделано сжатие тестового набора изображений алгоритмом JPEG2000 (рис. 2.11). Уровень артефактов сжатия как и ранее определялся внутренним параметром качества алгоритма, а также оценивался с помощью усреднённой метрики PSNR. Для получившихся наборов изображений показатели этой метрики составили: PSNR=33,68 дБ для качества 120, PSNR=32,61 дБ для качества 90, PSNR=31,79 дБ для качества 60. Площадь под ROC-кривой по результатам исследований составила: без шума = 0,942, качество 120 = 0,941, качество 90 = 0,940, качество 60 = 0,939. Таким образом, сжатие алгоритмом JPEG2000 оказывает малое влияние на работы детектора. Увеличение степени сжатия до PSNR=31,79 дБ понижает площадь под ROC-кривой на 0,3%.

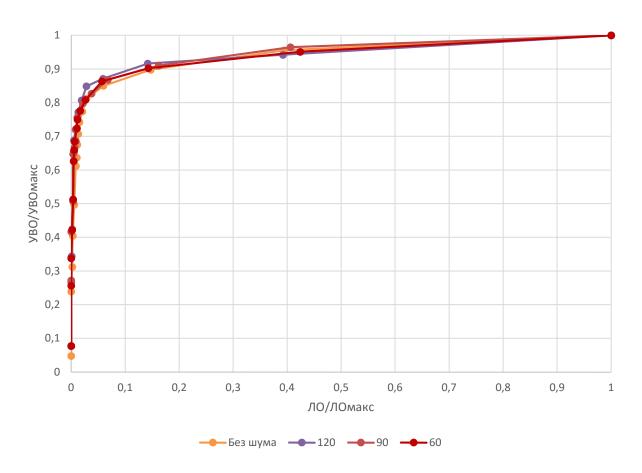


Рис. 2.11. ROC-кривые для сжатия JPEG2000

#### 2.6. Краткие выводы

Результаты проведенных исследований позволяют сделать следующие основные выводы:

- 1. Разработанный алгоритм на основе АРД увеличивает площадь под ROC-кривой на 13% по сравнению со стандартным алгоритмом детектирования Виолы-Джонса. Реализация алгоритма детектирования на основе локальных бинарных шаблонов из библиотеки Open CV показывает еще более худшие результаты.
- 2. Исследования робастности алгоритма детектирования на основе АРД показывают, что АБГШ, мультипликативный и импульсный шум вида «соль-и-перец» оказывают сильное влияние на работу алгоритма (уменьшение площади под ROC-кривой на 21,2% при снижении метрики PSNR до 17,99 дБ для АБГШ, на 8,8% при снижении метрики PSNR до 15,71 дБ для импульсного шума «соль-и-перец» и на 18,4%

при снижении метрики PSNR до 18,75 дБ для мультипликативного шума).

3. Исследования робастности алгоритма детектирования на основе АРД показывают, что искажения типа размытие и сжатие алгоритмами JPEG и JPEG2000 оказывают слабое влияние на работу алгоритма (уменьшение площади под ROC-кривой на 3,5% при снижении метрики PSNR до 21,58 дБ для размытия, на 7,5% при снижении метрики до PSNR до 24,73 дБ для JPEG и на 0,3% при снижении метрики PSNR до 31,79 дБ для JPEG2000).

# ГЛАВА 3. РАЗРАБОТКА И ИССЛЕДОВАНИЕ АЛГОРИТМА ДЕТЕКТИРОВАНИЯ ФИГУРЫ ЧЕЛОВЕКА

#### 3.1. Вводные замечания

При разработке комплексов охранного телевидения, видеонаблюдения, систем помощи водителю на дороге, алгоритмов технического зрения и при поиске изображений в сети Интернет широко применяют алгоритмы детектирования фигуры человека (пешехода) на цифровых изображениях [4]. Система автоматического детектирования пешеходов по произвольному изображению на входе системы должна определить, имеются ли на этом изображении пешеходы, и если да, то указать, где они находятся и каков их размер (рис. 3.1).



Рис. 3.1. Пример работы детектора фигуры человека

Решение этой задачи сопряжено с рядом трудностей. Во-первых, детектору приходится иметь дело с изменением точки наблюдения,

различным положением тела пешехода, одеждой, освещением. Во-вторых, неидеальные условия съемки создают шум на изображении и мешают корректной работе детектора. В-третьих, фигура пешехода часто перекрывается посторонними предметами (другими людьми), и только часть объекта оказывается в окне детектора.

Требования к точности алгоритма поиска пешехода на изображении также могут широко варьироваться. Так алгоритм, разработанный для работы в условиях неизменного положения камеры и постоянного фона, должен генерировать минимальное количество ошибок детектирования. В то же время для детектора, рассчитанного на широкий класс задач и большое разнообразие сцен, допустимо довольно большое количество ложных срабатываний на областях фона. Настройка алгоритма детектирования применительно к определенной задаче осуществляется за счет обучения классификатора, разделяющего предоставленные ему изображения на два класса: «пешеход» и «не пешеход» [76].

Существующие на данный момент алгоритмы детектирования пешеходов преимущественно опираются на описание изображения через различные системы признаков. Впервые качественно проработанная система признаков для детектирования пешеходов на цифровых изображениях предложена в работе Орена [77] и основывалась на вейвлетах Хаара. В дальнейшем Папагеоргио и Поггио [78] использовали новую систему хааровских признаков для распознавания лиц, машин и фигур людей. Благодаря работам Виолы и Джонса [11] использование хааровских признаков стало де-факто стандартом в задаче распознавания объектов на изображениях, в особенности, в задаче детектирования лиц [79]. В настоящее время для задачи детектирования пешеходов на изображениях помимо хааровских стали использоваться новые наборы признаков, такие как SIFT (Scale-invariant feature transform) [80] и HOG (Histograms of Oriented Gradients) [82].

Ключевой особенностью SIFT-признаков является инвариантность к повороту, масштабированию и отражению изображения вокруг особой точки [80]. Шашуа и его коллеги [81] вручную разделили изображение человека на 13 областей и вычислили SIFT-подобные признаки для каждого выделенного на изображении участка тела человека, а затем, используя эти признаки для обучения классификатора на основе AdaBoost, создали систему обнаружения пешеходов для камеры движущегося автомобиля.

При формировании HOG-признаков вычисляются направления градиента функции интенсивности в областях изображения, предварительно разбитом на блоки. Использование данных признаков в сочетании с классификатором на основе метода опорных векторов (МОВ) предложено Далалом и Триггсом [12] и позволило достичь степени детектирования пешеходов, недоступной более ранним алгоритмам. Позже Далал и Триггс расширили свой подход для обнаружения людей в видеопотоке с помощью вычисления гистограмм перемещения и появления [82]. Чжу и др. [83] построили каскад классификаторов, использующий систему HOG-признаков для значительного ускорения метода Далала. Для детектирования человека возможен также подход на основе обнаружения отдельных частей тела, который направлен на борьбу со смещением положения тела или частичным закрытием фигуры человека [84].

В данной главе предлагается модифицированный алгоритм детектирования пешеходов на цифровых изображениях, основанный на детекторе Далала-Триггса. Дается информация о методике тестирования алгоритмов детектирования пешеходов, и приводятся данные практического эксперимента по анализу эффективности разработанного детектора.

# 3.2. Реализация алгоритма вычисления НОG-признаков 3.2.1. Вычисление градиента

Первым шагом вычислений во многих детекторах особых точек является нормализация цвета и гамма-коррекция. Далал и Триггс установили,

что для дескриптора НОС этот шаг можно опустить, поскольку последующая нормализация даст TOT же результат. Поэтому на рассчитываются значения градиентов. Самым распространенным методом применение одномерной дифференцирующей является маски горизонтальном и/или вертикальном направлении. Этот метод требует фильтрации цветовой или яркостной составляющей при помощи следующих фильтрующих ядер:

$$[-1, 0, 1] \text{ M} [-1, 0, 1]^T$$
 (3.1)

Далал и Триггс использовали более сложные маски, такие как оператор Собеля 3×3 или диагональные маски, но эти маски показали более низкую производительность для данной задачи. Они также экспериментировали с размытием по Гауссу перед применением дифференцирующей маски, но также обнаружили, что пропуск этого шага увеличивает быстродействие без заметной потери качества [90].

## 3.2.2. Группировка направлений

На следующем шаге вычисляются гистограммы ячеек. Каждый пиксель в ячейке участвует во взвешенном голосовании для каналов гистограммы направлений, основанном на значении градиентов. Ячейки могут быть прямоугольной или круглой формы, каналы гистограммы равномерно распределяются от 0 до 180 или же от 0 до 360 градусов, в зависимости от того, вычисляется «знаковый» или «беззнаковый» градиент. Далал и Тригтс обнаружили, что беззнаковый градиент совместно с девятью каналами гистограммы дает лучшие результаты при распознавании людей. При распределении весов в голосовании вес пикселя может задаваться либо абсолютным значением градиента, либо некоторой функцией от него; в реальных тестах абсолютное значение градиента дает лучшие результаты. Другими возможными вариантами могут быть квадратный корень, квадрат или урезанное абсолютное значение градиента [82].

#### 3.2.3. Блоки дескрипторов

Для принятия во внимания яркости и контрастности градиенты следует локально нормировать, для чего ячейки нужно сгруппировать в более крупные связные блоки. Дескриптор НОG, таким образом, является вектором компонент нормированных гистограмм ячеек из всех областей блока. Как правило, блоки перекрываются, то есть каждая ячейка входит более чем в один конечный дескриптор.

Используются две основные геометрии блока: прямоугольные R-HOG и круглые C-HOG. Блоки R-HOG обычно являются квадратными сетками, характеризующимися тремя параметрами: количеством ячеек на блок, количеством пикселей на ячейку и количеством каналов на гистограмму ячейки. В эксперименте Далала и Триггса оптимальными параметрами являются блоки 3х3, ячейки 6х6 и 9 каналов на гистограмму. Более того, они обнаружили, что можно слегка повысить скорость вычислений, применяя гауссов фильтр внутри каждого блока до процедуры голосования, что, в свою очередь, снижает вес пикселей на границах блоков. Блоки R-HOG оказываются очень похожими на SIFT-дескрипторы; однако, несмотря на их похожую структуру, блоки R-HOG вычисляются на плотных сетках фиксированного масштаба без фиксированного направления, в то время как SIFT-дескрипторы вычисляются в разреженных, не чувствительных к масштабу ключевых точках изображения и используют поворот для выравнивания направления. Кроме того, для кодирования информации о форме объектов блоки R-HOG используются совместно, в то время как SIFTдескрипторы используются по отдельности.

Блоки С-НОG имеют 2 разновидности: с цельной центральной ячейкой и разделенной на сектора. Эти блоки могут быть описаны 4 параметрами: количество секторов и колец, радиус центрального кольца и коэффициент расширения для радиусов остальных колец. Далал и Триггс обнаружили, что обе разновидности показали одинаковый результат, и разделение на 2 кольца и 4 сектора с радиусом 4 пикселя и коэффициентом расширения 2 дало

лучший результат в их эксперименте. Кроме того, гауссово взвешивание не дало никаких улучшений при использовании блоков С-НОG. Эти блоки похожи на контексты формы, но имеют важное отличие: блоки С-НОG содержат ячейки с несколькими каналами направлений, в то время как контексты формы используют только наличие одного края.

#### 3.2.4. Нормализация блоков

Далал и Триггс исследовали четыре метода нормализации блоков. Пусть v — ненормированный вектор, содержащий все гистограммы данного блока,  $\|v\|_k$  — его k-норма при k=1,2 и e — некая малая константа (точное значение не так важно). Тогда нормировочный множитель можно получить одним из следующих способов:

- L2-норма: 
$$f = \frac{v}{\sqrt{\|v\|_2^2 + e^2}};$$
 (3.2)

– L2-норма ограничивается сверху (значения v, большие 0,2, полагаются равными 0,2) и перенормируется, как в

- L1-норма: 
$$f = \frac{v}{(\|v\|_1 + e)};$$
 (3.3)

– корень из L1-нормы: 
$$f = \sqrt{\frac{v}{(\|v\|_1 + e)}}$$
. (3.4)

Далал и Триггс установили, что L1-норма дает менее надежные результаты, чем остальные, которые работают приблизительно одинаково хорошо, однако все методы значительно улучшают результаты по сравнению с ненормализованными [82].

## 3.2.5. Метод бегущего окна и HOG-детектор

В данном разделе рассматривается использование НОG-признаков для детектирования объектов на изображении. Рассмотрим, каким образом можно решить задачу классификации изображений при помощи НОG-

дескриптора. Пусть требуется определить, является ли заданное изображение изображением пешехода или нет. Другими словами, нужно классифицировать изображения на те, которые содержат искомый объект (пешехода), и не содержат его. Для этого, можно поставить в соответствие изображению его признаковое описание, полученное с помощью HOG, а затем использовать алгоритмы машинного обучения для непосредственного осуществления классификации. Поскольку большинство алгоритмов обучения с учителем работают в пространстве признаков фиксированной размерности, вектора НОG-признаков для различных изображений должны быть одинаковой длины, а, следовательно, классифицируемые изображения быть одинакового размера. Более того, ДЛЯ обеспечения должны приемлемого качества решения этой задачи предполагается, что данные изображения содержат объекты одинаковых (близких) размеров, которые занимают одну и ту же область изображения.

Теперь перейдем непосредственно к решению задачи детектирования. В том случае, если выполнены изложенные выше предположения относительно размеров и расположения объектов, то данная задача эквивалентна задаче классификации и, следовательно, может быть решена описанным методом. Однако, как правило, это не так. Для обобщения на такие случаи может использоваться метод бегущего окна, который сводится к рассмотрению задачи детектирования как множества задач классификации. Пусть нужно осуществить поиск на изображении всех объектов заданного размера  $w \times h$ . Для этого с некоторым шагом по вертикали dy и по горизонтали dxрассмотрим прямоугольные области изображения размера  $w \times h$  с верхним левым углом с координатами  $(i \cdot dx, j \cdot dy), i = \overline{0, n}, j = \overline{0, m}$  и для каждого из них произведем классификацию. Таким образом, по изображению как бы «бежит» окно размера  $w \times h$ . Дальнейшее обобщение метода на случай поиска объектов разного размера возможно за счет многократного

масштабирования изображения и осуществления детектирования описанным выше методом.

# 3.3. Алгоритм слежения за пешеходами на основе метода Лукаса-Канаде

#### 3.3.1. Точечные особенности

Любое изображение в отдельности, несмотря на кажущуюся простую структуру – двумерную матрицу чисел, содержит очень сложную и комплексно-интегральную информацию о наблюдаемой сцене [4]. Извлечь из изображений какую-нибудь структурированную осмысленную И информацию наблюдаемой сцене достаточно сложно. Видеопоследовательность, полученную с движущейся камеры, особенно если на ней запечатлена динамическая сцена, анализировать еще сложнее. Поэтому просто необходимы технологии, которые позволили бы извлекать из видеопоследовательностей некоторую, пускай очень неполную, осмысленную и достаточно просто структурированную информацию об объектах сцены и ее динамике.

Одной из таких технологий является слежение за точечными особенностями в потоке изображений. В самом деле, точка — это наиболее простой и понятный геометрический элемент, с которым, как кажется, проще всего работать. Точка может рассматриваться как элемент дискретного представления некоторой функции. Если выделить в сцене некоторое количество «особых» точек, и определить их положение на каждом кадре, то полученые данные смогут дать огромное количество информации о структуре сцены, а также о параметрах камеры, с которой получены изображения.

Некоторые методы используют слежение за особенностями для определения траектории наблюдаемых объектов, для вычисления структуры наблюдаемой динамической сцены, выделения отдельных движений и движущихся объектов. Если известны некоторые ограничения на

наблюдаемую сцену (например, что она статическая, а камера движется по сцене), то можно определить параметры внутренней и внешней калибровки камеры. Если в сцене отслеживается большое количество особых точек, то даже может быть построена грубая трехмерная модель всей сцены.

#### 3.3.2. Алгоритм Лукаса-Канаде

После того как фигуры людей детектированы, они поступают на блок сопровождения, задачей которого является отслеживать изменения положения и размеров лиц от кадра к кадру. В работе использовался алгоритм, основанный на широко используемом в техническом зрении дифференциальном локальном методе вычисления оптического потока Лукаса-Канаде.

После детектирования люди заносятся в список сопровождаемых лиц. В случае если рамка вновь детектированного объекта пересекается с рамкой уже сопровождаемого объекта, делается вывод о том, что это одно и то же лицо, и новое лицо к списку не добавляется, а координаты и размеры рамки сопровождаемого лица уточняются на основе информации, полученной от детектора. Если сопровождаемое лицо не подтверждается детектированием в течение заданного числа кадров, TO ЛИЦО удаляется ИЗ списка сопровождаемых.

Алгоритм Лукаса-Канаде, основанный на работах 1981 года [4], является одним из самых простых. В этом алгоритме движение рассчитывается самым простым образом и не учитывает возможные изменения структуры региона вокруг отслеживаемой точки.

Метод Лукаса и Канаде подразумевает, что смещение между двумя кадрами мало и одинаково в пределах соседей некоторой точки p по определению. Этот алгоритм в принципе применим для функций любой размерности n. Пусть x — особенность первой функции F, необходимо найти такую точку x+h функции G, чтобы разность окрестностей этих точек по мере была минимальна.

Расстояние между окрестностями записывается в виде:

$$E = \sum [F(x+h) - G(x)]^{2}, \qquad (3.5)$$

где F(x), G(x) — две функции.

Функцию F(x) с помощью разложения в ряд Тейлора можно приближенно представить в виде:

$$F(x+h) = F(x) + h \frac{\partial F(x)}{\partial x}, \qquad (3.6)$$

где 
$$\frac{\partial}{\partial x} = \left[\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2}, \dots, \frac{\partial}{\partial x_n}\right]^T$$
 – градиент.

Используя это приближение, ищется минимум E путем дифференцирования и приравнивания производной к нулю:

$$0 = \frac{\partial}{\partial x} E \approx \frac{\partial}{\partial x} \sum_{x} \left[ F(x) + h \frac{\partial F(x)}{\partial x} - G(x) \right]^{2} =$$

$$= \sum_{x} 2 \frac{\partial F}{\partial x} \left[ F(x) + h \frac{\partial F(x)}{\partial x} - G(x) \right]. \tag{3.7}$$

Отсюда смещение h можно получить как

$$h = \left[ \sum_{x} \left( \frac{\partial F}{\partial x} \right)^{T} \left[ G(x) - F(x) \right] \right] \left[ \sum_{x} \left( \frac{\partial F}{\partial x} \right)^{T} \left( \frac{\partial F}{\partial x} \right) \right]^{-1}. \tag{3.8}$$

Как было указано ранее, задача слежения за особенностями без учета аффинных искажений является поиском величины оптического потока в наборе точек. Поэтому метод Лукаса-Канаде часто применяется для поиска оптического потока во всем изображении.

Рассмотрим пример использования алгоритма Лукаса-Канаде. Найдем такую точку, которая минимизирует взвешенную квадратичную модель ограничений на производные первого порядка в некотором окне поиска  $\Omega$ 

$$\sum_{x \in \Omega} W^{2}(x) \cdot \left[ \nabla I(x) \times d + I_{t}(x) \right]^{2}, \tag{3.9}$$

где W(x) — функция весов в окне, отражающая важность близких и дальних соседей,  $I_t$  — полуразность первого и второго кадра,  $\nabla I$  — градиент функции I , приближаемый сверткой по ядру

$$\frac{1}{2} \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}. \tag{3.10}$$

Решение имеет вид

$$A^T \cdot W^2 \cdot A \cdot v = A^T \cdot W^2, \tag{3.11}$$

где для n точек из  $\Omega$  на левом кадре

$$A = \left[\nabla I(x_1) \dots \nabla I(x_n)\right]^T \tag{3.12}$$

$$W = diag [W(x_1)...W(x_n)]$$
 (3.13)

$$b = -[I_t(x_1)...I_t(x_n)]^T$$
(3.14)

$$v = [A^{T} \cdot W^{2} \cdot A]^{-1} \cdot A^{T} \cdot W^{2} \cdot b$$
 (3.15)

Это решение находится приближенно, когда матрица  $A^T \cdot W^2 \cdot A$  не вырождена, так как это матрица размером 2×2.

$$A^{T} \cdot W^{2} \cdot A = \begin{pmatrix} \sum W^{2}(x)I_{x}^{2}(x) & \sum W^{2}(x) \cdot I_{x}(x) \cdot I_{y}(x) \\ x \in \Omega & x \in \Omega \\ \sum W^{2}(x) \cdot I_{x}(x) \cdot I_{y}(x) & \sum W^{2}(x)I_{x}^{2}(x) \\ x \in \Omega & x \in \Omega \end{pmatrix}. \tag{3.16}$$

Перед применением алгоритма требуется сгладить входное изображение, чтобы убрать случайные шумы, т. к. алгоритм локальный и сильно зависит от любых шумов.

Основными ограничениями алгоритма являются его предположения — постоянная яркость движущихся точек и одинаковое направление движения в пределах окна поиска. Также алгоритм плохо срабатывает на слаботекстурированных областях (когда матрица близка к вырожденной).

### 3.4. Алгоритм Далала-Триггса

Далала-Триггса работу [82], Рассмотрим алгоритма который демонстрирует один из лучших результатов среди известных алгоритмов детектирования пешеходов [85, 86]. На первой стадии работы алгоритма нужно сформировать обучающую выборку из изображений «пешеходов» и «не пешеходов» разрешением 64х128 пикселей и вычислить признаки. Каждое изображение в обучающем наборе делится на ячейки 8х8 пикселей. Из ячеек методом бегущего окна формируются пересекающие блоки размером 2х2 (т. е. 16х16 пикселей). Всего изображение разбивается на 7x15=105блоков. Каждая ячейка представляется В виде  $\mathbf{v} = (v_1, v_2, ..., v_9)$ . Внутри ячейки для каждого из 64 пикселей вычисляются модуль и направление градиента. Найденное значение модуля прибавляется к одной из координат вектора у. К какой именно координате – зависит от найденного направления градиента.

Для каждого блока путем конкатенации четырех векторов, соответствующих четырем входящим в блок ячейкам, формируется вектор размерности 36 и затем нормируется. Соединяя полученные со всех 105 блоков вектора в один длинный вектор, состоящий из 3780 элементов, получим описание изображения через 3780 НОG-признаков. Таким образом, каждому обучающему изображению соответствует точка в 3780-мерном пространстве признаков, которой, в свою очередь, соответствует значение «пешеход» (+1) или «не пешеход» (-1).

На втором этапе работы алгоритма с помощью метода опорных векторов [87] между точками двух классов строится разделяющая гиперплоскость — решающая поверхность. Поиск пешеходов на тестовом изображении с помощью обученного классификатора осуществляется методом бегущего окна, приводимого с помощью масштабирования к стандартному разрешению 64х128, и вычислением НОG-признаков внутри каждого окна. В зависимости от вычисленных признаков классификатор решает, является или нет область, покрываемая окном детектора, изображением пешехода.

### 3.5. Разработка модифицированного алгоритма

Многие исследователи пытаются улучшить качество и скорость работы алгоритма Далала-Триггса путем замены алгоритма классификации на основе метода опорных векторов алгоритмом бустинга [88]. При этом подходы к выбору слабых классификаторов, используемых в цепочке формирования сильного, могут значительно отличаться. Так, например, в работе [86] роль слабых классификаторов играли деревья принятия решения.

В данной работе в качестве слабых классификаторов выбраны линейные машины опорных векторов, которые переобучались на сложных примерах. Для этого в тестовой выборке отбирались те части изображений, на которых детектор на основе алгоритма Далала-Триггса дал неверные ответы. Эти сложные примеры приводились к стандартному разрешению окна детектора 64х128 пикселей путем масштабирования. В процессе обучения финального классификатора модифицированного алгоритма акцент делается именно на эти сложные эталоны, которые распознаются хуже.

подробнее Рассмотрим процесс формирования классификатора модифицированного алгоритма. Исходная обучающая выборка, использованная для обучения детектора Далала-Триггса, состояла из М изображений  $(\mathbf{x}_i, y_i)$ , характеризуемых вектором НОG-признаков  $\mathbf{x}_i$ принадлежностью к классу «пешеходов»  $y_i \in \{-1; 1\}$ . К обучающей выборке добавлялось и изображений, на которых детектор Далала-Триггса дал некорректные ответы. Выбранные из базы для контрольной выборки изображения в последующем исключались из тестовой выборки. Так формируется новая база из M + N изображений для обучения классификатора с помощью алгоритма на основе бустинга [88] с изначальными весами, инициализированными следующим образом:

$$D_1(i) = \frac{1}{M+kN} \, \text{при} \,\, i = 1 \dots M \quad \text{и} \,\, D_1(i) = \frac{k}{M+kN} \,\, \text{при}$$
 
$$i = M \,\, + 1 \dots M \,\, + N \,\, ,$$

где k — коэффициент усиления сложных примеров, выбираемый эмпирически.

Затем производится поиск слабого классификатора  $h_1(x)$ , разделяющего объекты классов «пешеход» и «не пешеход» согласно минимуму взвешенной ошибки  $\varepsilon_i$ :

$$h_1(x) = \arg\min_{h_j \in H} \varepsilon_j = \sum_{i=1}^{M+N} D_1(i) [y_i \neq h_j(x_i)]$$
(3.17)

Аналогичным образом производится поиск еще T-1 классификаторов. При этом на каждой стадии  $t=1\dots T$  поиска слабого классификатора происходит изменение весов объектов  $D_t(i)$  согласно соотношению

$$D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i h_t(x_i))}{Z_t},$$
(3.18)

где  $\alpha$  — коэффициент точности слабого классификатора  $h_i(x)$ , вычисляемый на основе ошибки, выдаваемой этим классификатором

$$\alpha_t = \frac{1}{2} \ln \left( \frac{1 - \varepsilon_t}{\varepsilon_t} \right), \tag{3.19}$$

а  $Z_{t}$  — нормировочный коэффициент, выбранный таким образом, чтобы совокупность весов  $D_{t+1}(i)$  удовлетворяла соотношению

$$\sum_{i=1}^{M+N} D_{t+1}(i) = 1. (3.20)$$

Изменение весов объектов происходит таким образом, что ошибки слабого классификатора предыдущей стадии получают больший вес на новой стадии, правильно обнаруженные объекты — меньший. Изменение весов объектов и процесс формирования новых слабых классификаторов на примере объектов в двумерном пространстве признаков иллюстрируется на рис. 3.2.

Итоговый сильный классификатор H(x), который будет формировать решающую поверхность в пространстве НОG-признаков, строится в виде линейной комбинации всех слабых классификаторов  $h_t(x)$ :

$$H(x) = sign\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right). \tag{3.21}$$

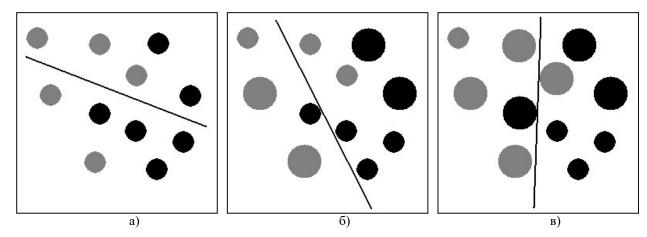


Рис. 3.2. Формирование слабых классификаторов в процедуре бустинга: а) для объектов с равными весами; б, в) для объектов с увеличенным весом ошибок предыдущего классификатора

#### 3.6. Оценка качества работы детектора фигуры человека

Качество работы детектора оценивается на тестовой выборке, не пересекающейся с выборкой, используемой при обучении. Для оценки качества детекторов используются такие характеристики как recall и precision, а также их среднее гармоническое – F-мера [89].

Проведем сравнение алгоритма Далала-Триггса и модифицированного алгоритма. В качестве дополнительного участника сравнения возьмем алгоритм Виолы Джонса [5], использующий хааровские признаки и бустинг. Для анализа работы алгоритмов использовалось две тестовых базы изображений:

– INRIA Person Dataset [18] – набор тестовых изображений со средним разрешением 640×480, снятых статичной камерой (рис. 3.3);

– CVC-02 Pedestrian Dataset [19] – набор тестовых изображений,
 полученных из видеопоследовательности с разрешением 640×480, снятой с
 помощью видеорегистратора автомобиля (рис. 3.4).









– Рис. 3.3. Примеры изображений из тестовой базы INRIA Person









Рис. 3.4. Примеры изображений из тестовой базы CVC-02 Pedestrian

Из базы INRIA Person для тестирования обученных каскадов было отобрано 200 изображений. На каждом из них вручную подсчитывалось количество фигур пешеходов, пригодных для детектирования, количество верных обнаружений каждого детектора и количество ложных срабатываний классификаторов. Аналогичным образом из базы CVC-02 Pedestrian взято 400 изображений, на которых с помощью исследуемых алгоритмов выделялись и подсчитывались правильно и неправильно выделенные пешеходы. Затем данные по всем изображениям суммировались и обобщались.

После тестирования алгоритмов на незашумленных изображениях была смоделирована ситуация их работы в условиях аддитивного белого гауссовского шума (АБГШ) с плотностью вероятности

$$p(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(z-\mu)^2}{2\sigma^2}},$$
 (3.22)

где  $\mu$  — математическое ожидание случайной величины z, определяющей интенсивность шумового воздействия,  $\sigma$  — среднеквадратическое отклонение (СКО).

К тестовым изображениям из базы INRIA Person добавлялся АБГШ с нулевым математическим ожиданием  $\mu = 0$  и различными СКО. Значения СКО менялись в интервале от 0 до 50 с шагом 5. Затем на изображениях проводилось детектирование пешеходов с помощью алгоритма Далала-Триггса и модифицированного алгоритма. Детектор Виолы-Джонса был исключен из тестирования, так как показал неудовлетворительные результаты уже на незашумленных изображениях.

### 3.7. Результаты тестирования

Проведенный эксперимент на изображениях из базы INRIA показал значительный перевес алгоритмов, которые используют вектор НОG-признаков для описания изображения, над алгоритмом, использующим хааровские признаки. Модификация алгоритма Далала-Тригтса позволила увеличить метрику Recall на 9 %, а метрику Precision на 10 % в сравнении с оригинальным алгоритмом. Результаты детектирования, полученные по итогам эксперимента, приведены в табл. 3.1.

Таблица 3.1. Сравнительный анализ работы детекторов на изображениях из базы INRIA Person

Алгоритм детектирования	Recall, %	Precision, %	F-мера, %
Алгоритм Виолы-Джонса	27	62	38
Алгоритм Далала-Триггса	57	83	68
Модифицированный алгоритм	66	93	77

Результаты работы исследуемых алгоритмов на изображениях из базы CVC-02 Pedestrian приведены в табл. 3.2. В случае с видеопоследовательностями алгоритм Виолы-Джонса дает близкую к нулю

обнаружения пешеходов, ЧТО показывает неэффективность степень применения системы хааровских признаков в задачах детектирования пешеходов. Столь низкий результат объясняется особенностями базы CVC-02 Pedestrian, на изображениях которой пешеходы редко оказываются обращенными лицом или спиной в камеру, а хааровские признаки хорошо подходят только для описания фронтально расположенных к камере пешеходов. Кроме τογο, качество изображений кадрах на видеорегистратора оказывается заметно хуже качества изображений, полученных со статичной камеры, что критично сказывается на работе всех рассматриваемых детекторов. Предложенный модифицированный алгоритм демонстрирует значительно более высокую метрику Precision (91 % против 65 %), чем алгоритм Далала-Тригса, за счет переобучения исходного классификатора на ложных отрицательных срабатываниях детектора в области фона, но при этом требует больше времени для обучения и настройки.

Таблица 3.2. Сравнительный анализ работы детекторов на изображениях из базы CVC-02 Pedestrian

Алгоритм детектирования	Recall, %	Precision, %	F-мера, %
Алгоритм Виолы-Джонса	3	50	6
Алгоритм Далала-Триггса	25	65	36
Модифицированный алгоритм	26	91	40

Результаты работы алгоритма Далала-Триггса и модифицированного алгоритма на изображениях базы INRIA Person в условиях наличия аддитивного белого гауссовского шума с нулевым математическим ожиданием  $\mu = 0$  приведены на рис. 3.5 и рис. 3.6. Таким образом, результаты моделирования показывают, что модифицированный алгоритм на базе бустинга оказывается более робастным к гауссовскому шуму, что особенно заметно на средних степенях зашумления  $\sigma = 10$ –20, что соответствует изображениям, получаемым в практических приложениях прикладного телевидения.

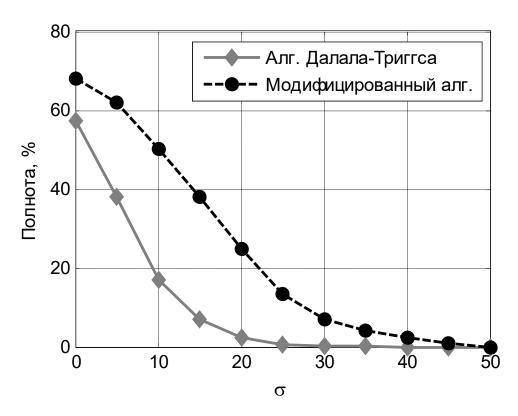


Рис. 3.5. Зависимость метрики Recall (полнота) от СКО гауссовского шума для алгоритма Далала-Триггса и модифицированного алгоритма

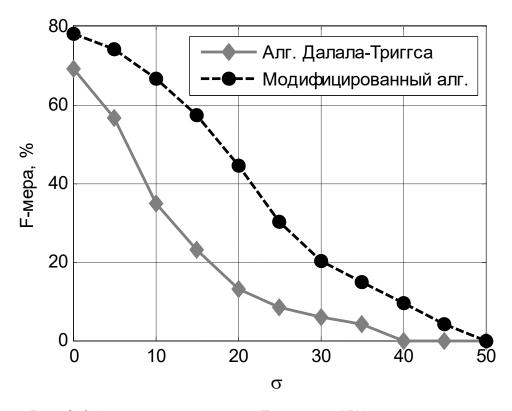


Рис. 3.6. Зависимость метрики F-мера от СКО гауссовского шума для алгоритма Далала-Триггса и модифицированного алгоритма

### 3.8. Краткие выводы

Результаты проведенных исследований алгоритмов детектирования фигуры человека позволяют сделать следующие основные выводы.

- 1. Модификация алгоритма детектирования фигуры человека на основе метода Далала-Триггса с использованием классификатора на базе бустинга и переобучения на сложных примерах позволила значительно повысить результаты по метрике Recall (с 83 % до 93 % для базы INRIA Person и с 65 % до 91 % для базы CVC-02 Pedestrian).
- 2. Использование модифицированного алгоритма позволяет достичь также увеличение метрики Precision (с 57 % до 67 % для базы INRIA Person и с 25 % до 26 % для базы CVC-02 Pedestrian).
- Модифицированный алгоритм детектирования фигуры человека показывает большую робастность к присутствию на изображении гауссовского шума, что особенно заметно на средних степенях зашумления, при σ = 10–20.

# ГЛАВА 4. РАЗРАБОТКА АЛГОРИТМОВ ДЕТЕКТИРОВАНИЯ ЧЕЛОВЕКА НА ВИДЕОИЗОБРАЖЕНИЯХ С КУПОЛЬНЫХ КАМЕР

#### 4.1. Вводные замечания

Анализ видеопотока с вертикально направленных камер — достаточно новое направление исследований. В связи с этим на данный момент в свободном доступе нет тестовых и обучающих баз, необходимых для разработки и анализа алгоритмов детектирования людей на видеоизображениях с купольных камер. Для решения данной проблемы было принято решение собрать необходимые видеоматериалы самостоятельно.

В дальнейшей работе использовались следующие видеопоследовательности, подготовленные автором:

- видеозапись с разрешением 1280×480 пикселей, снятая в г. Ярославль на улице Первомайская, высота подвеса камеры – 4 метра;
- видеозапись с разрешением 640×480 пикселей, снятая в лаборатории кафедры динамики электронных систем,
   ЯрГУ им. П.Г. Демидова, высота подвеса камеры 2,5 метра;
- видеозапись с разрешением 704×576 пикселей, снятая в г. Ярославль в школе № 3, высота подвеса камеры 2,5 метра;
- видеозапись с разрешением 1920×1080 пикселей; снятая в г. Иваново,
   высота подвеса камеры − 3,4 метра.

На основе указанных видеозаписей были составлены обучающие (рис. 4.1) и тестовые наборы для детектирования людей, описанные в табл. 4.1. Описание обучающих (рис. 4.2) и тестовых (рис. 4.3) наборов для детектирования человеческих голов приведено в табл. 4.2. Для тестирования алгоритмов также использовалась видеозапись, сделанная в спортзале ЯрГУ им. П.Г. Демидова, показанная на рис. 4.4.

 Таблица 4.1

 Описание обучающих и тестовых баз для детектирования человека

Название	Число обучающих	Число тестовых	Разрешение
	изображений	изображений	
«Улица200»	200	_	245×245
«Кафедра200»	200	_	260×260
«Школа200»	200	100	240×240

 Таблица 4.2

 Описание обучающих и тестовых баз для детектирования голов

Название	Число	Число тестовых	Разрешение
	обучающих	изображений	
	изображений		
«Улица»	588	84	40×40
«Кафедра»	248	74	100×100
«Школа»	407	181	97×97
«Иваново»	1328	159	139×139







Рис. 4.1. Примеры обучающих и тестовых баз для детектирования человека

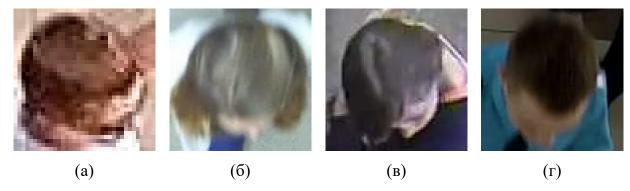
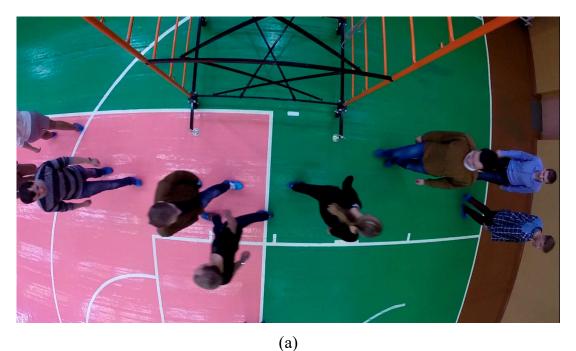


Рис. 4.2. Примеры обучающих баз для детектирования голов: (a) «Улица», (б) «Кафедра», (в) «Школа», (г) «Иваново»



Рис. 4.3. Примеры изображений из тестовых наборов: (а) «Улица», (б) «Кафедра», (в) «Школа», (г) «Иваново»



Direct 2021 and Section 1997.

Рис. 4.4. Примеры кадров тестовых видеопоследовательностей, снятых в спортзале

**(**6)

## 4.2. Разработка алгоритма на основе детектирования движения

Первый используемый подход основан на детектировании движущихся объектов с последующей бинарной классификацией кандидатов (объект / не объект). Данный алгоритм можно разделить на три основных этапа, которые проиллюстрированы на рис. 4.5.

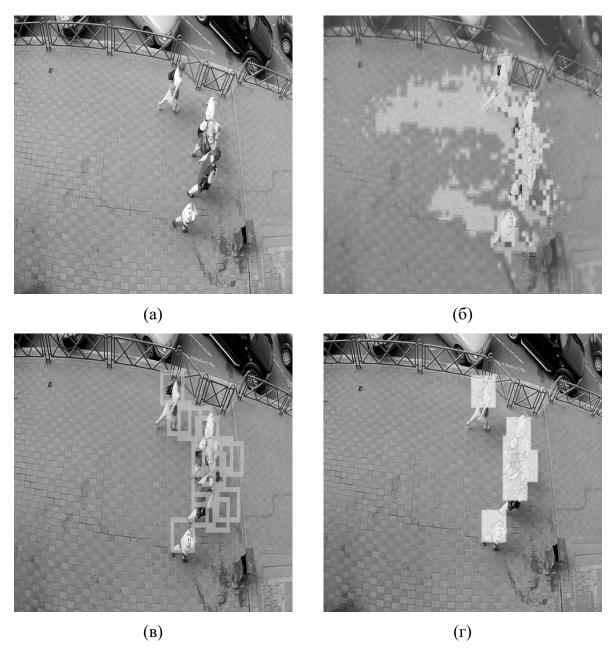


Рис. 4.5. Иллюстрация алгоритма на основе детектирования движения:
(а) исходный кадр; (б) первый этап алгоритма; (в) второй этап алгоритма; (г)
третий этап алгоритма

На первом этапе анализируется каждый третий кадр. Осуществляется деление блоков пикселей 8×8 на «движущийся объект», «неподвижный объект» и «статичный фон». Вычисляется разница между текущим кадром и сохраненным фоном, а также между двумя соседними кадрами. На основе этих данных происходит классификация блоков (рис. 4.5б). В результате получаем маску объекта. Настраиваемые параметры: порог срабатывания и время хранения фона.

На втором этапе выделяются кандидаты. Происходит анализ количества принадлежащих объекту внутри сканирующего окна. превышении определенного порога происходит выделение маски кандидата размером 56×56 пикселей. Для улучшения результатов может применяться фильтрация морфологическим фильтром, предварительная также децимация вырезанных фрагментов путем фиксированного отступа от ранее найденных кандидатов. В результате получаем области-кандидаты (рис. 4.5в). Настраиваемые параметры: порог срабатывания, процент допустимого перекрытия между фрагментами.

На третьем этапе происходит распознавание голов. В качестве признаков вычисляются гистограммы визуальных слов, а в качестве классификатора используется нелинейная машина опорных векторов. Результат – отобранные фрагменты (рис. 4.5г). Настраиваемые параметры: число обучающих изображений, число визуальных слов, параметры ядра машины опорных векторов.

Итоговая блок-схема алгоритма приведена на рис. 4.6.



Рис. 4.6. Блок-схема алгоритма на основе детектирования движения

Так как данный подход использует детектирование движения, при котором необходимо сравнение соседних кадров, тестирование проводилось на видеозаписи «Улица», где анализировался каждый 15 кадр. Полученный результат по метрике F-мера — 88,2%.

Основным недостатком этого подхода является зависимость работы алгоритма от скорости перемещения объекта. В случае медленного движения или вовсе отсутствия заметного движения данный подход дает неудовлетворительные результаты [91].

### 4.3. Разработка алгоритма на основе контекста формы

Второй подход основан на такой априорной информации об объекте, как форма. Существуют различные способы описания формы, в данном подходе используется метод контекста формы [92].

Основная идея подхода заключается в выборе n точек на границе объекта и описании их положения относительно друг друга. Используя полученное описание, можно установить соответствие между  $p_i$  точкой неизвестного объекта и  $q_j$  точкой эталонного объекта. Проще всего описать соответствие с помощью локальных дескрипторов, описанных, например в [93]. В данной работе предложен дескриптор контекста формы. В качестве него можно использовать n—1 вектор от одной точки до всех остальных, но такое описание было бы слишком громоздким. Поэтому было принято решение использовать гистограмму относительного положения точек (рис. 4.7в).

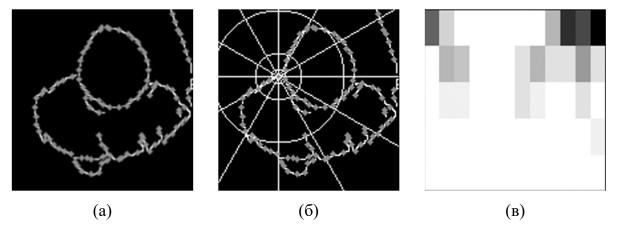


Рис. 4.7. Иллюстрация алгоритма на основе контекста формы: (а) п точек, расставленных на границе объекта; (б) диаграмма, состоящая из 5 окружностей, каждая из которых разделена на 12 секторов; (в) гистограмма контекста формы

Предлагаемый алгоритм можно условно разделить на 5 этапов.

– Выделение границы неизвестного объекта. Для этой задачи был использован детектор границ Кэнни [94], на выходе которого получим бинарное изображение границ объекта. После чего на границе равномерно расставляются *п* точек (рис. 4.7а).

— Далее для каждой точки неизвестного объекта  $p_i$  требуется вычислить дескриптор контекста формы. Для решения этой задачи используются диаграммы (рис. 4.76). В итоге получаем гистограмму  $5 \times 12$  (рис. 4.7в), визуализирующую относительное расположение точек, относительно  $p_i$ -й которой вычислялся дескриптор. Элементы гистограммы соответствуют сегментам диаграммы, а интенсивность черного цвета тем больше, чем больше количество точек внутри сегмента. Кроме того, при изображенную на рис. 4.7б, вычислении гистограммы диаграмму, необходимо выравнивать относительно касательной к границе объекта для достижения инвариантности к повороту объекта в пространстве. Для полученных гистограмм и гистограмм точек  $q_i$  эталонного объекта логично использование критерия хи-квадрат [95], как вес контекста формы, для того чтобы нормализовать K областей гистограмм g(k) неизвестного объекта и h(k) — эталонного объекта

$$C_S = \frac{1}{2} \sum_{k=1}^{K} \frac{[g(k) - h(k)]^2}{g(k) + h(k)}.$$
 (4.1)

– Полученные ранее веса необходимо минимизировать путем сопоставления точек неизвестного объекта с наиболее соответствующими им точками эталонного объекта. Для этого возможно применение алгоритма, предложенного R. Jonker и A. Volgenant [96]. Для оптимизации этого процесса можно применять преобразование исходного изображения, используя аффинные или аналогичные преобразование [97], но тогда необходимо учесть влияние данного преобразования при расчете конечного расстояния форм

$$D_{SC}(P,Q) = \frac{1}{n} \sum_{p \in P} \arg\min_{q \in Q} C(p, T(q)) + \frac{1}{m} \sum_{q \in Q} \arg\min_{p \in P} C(p, T(q)), \quad (4.2)$$

где  $T(\cdot)$  – оценка преобразования.

Так же с помощью данных методов можно отсеивать точки случайных выбросов, которые не имеют отношения к распознаваемому объекту.

– На последнем этапе строим бинарный классификатор для принятия решения, является ли распознаваемый объект человеком или нет. Для этого можно использовать алгоритм ближайших соседей, машину опорных векторов и другие алгоритмы машинного обучения для построения классификаторов [98].

Блок-схема разработанного алгоритма приведена на рис. 4.8.



Рис. 4.8. Блок-схема алгоритма на основе контекста формы

Данный алгоритм показал неудовлетворительные результаты как по скорости работы, так и по качеству, поэтому была реализована и протестирована его модификация, где на первом этапе происходила классификация движущихся объектов и удаление фона аналогично предыдущему подходу, а тестирование проводилось на видеозаписи «Кафедра». Полученный результат по метрике F-мера равен 71,8%.

### 4.4. Разработка алгоритма на основе НОС-признаков

Для решения данной задачи также был реализован подход, аналогичный алгоритму Далала-Триггса [12], который был подробно рассмотрен ранее. Используемые параметры приведены в табл. 4.3. Обучение классификатора производилось в среде MATLAB.

Тестирование проводилось на тестовом наборе изображений «Школа». Полученный результат по метрике F-мера — 95,8%. Проведенные исследования показали, что данный подход дает неудовлетворительные результаты в случае анализа видеоизображений с высокой плотностью расположения людей в сцене.

Параметр	Значение	Описание
CellSize	8×8	Размер ячейки в пикселях
BlockSize	2×2	Размер блока в ячейках
BlockType	R-HOG	Прямоугольная форма блока
positiveData	200	Количество положительных примеров
negativeData	200	Количество отрицательных примеров
kernel_function	rbf	Радиальная базисная ядерная
		функция Гаусса
method	SMO	Метод поиска разделяющей
		гиперплоскости – последовательная
		минимальная оптимизация

### Блок-схема алгоритма приведена на рис. 4.9.



Рис. 4.9. Блок-схема алгоритма на основе НОС-признаков

## 4.5. Сравнительный анализ работы алгоритмов

Реализованы и протестированы 3 подхода для детектирования людей на видеоданных с купольных камер, результаты тестирования по трем метрикам приведены в табл. 4.4.

Таблица 4.4 Результаты тестирования алгоритмов детектирования людей на видеоизображениях с купольных камер

Алгоритм	Recall, %	Precision, %	F-мера, %
На основе	88,0	88,5	88,2
детектирования			
движения			
На основе контекста	78,6	66,0	71,8
формы			
На основе HOG-	92,0	97,0	95,8
признаков			

Установлено, что наилучшие предварительные результаты получены для алгоритма на основе НОG-признаков. Стоит отметить, что все перечисленные выше алгоритмы слабо устойчивы к ситуациям плотного размещения людей в сцене, в тестовой базе алгоритма на основе НОG-признаков таких ситуаций не было, этим, в частности, и был обусловлен высокий уровень первоначальных результатов.

### 4.6. Разработка алгоритма на основе бустинга

В отличие от предыдущих подходов объектом детектирования выбран не весь человеческий контур, обладающий очень большой вариативностью, а человеческая голова. Данный шаг позволил анализировать видеопотоки с высокой плотностью расположения людей в сцене. Обучение каскадного классификатора производилось с помощью стандартной библиотеки OpenCV с параметрами, перечисленными в табл. 4.5.

Таблица 4.5 Параметры обучения каскадного классификатора бустинга

Параметр	Значение	Описание
numPos	3400	Количество положительных примеров
numNeg	4187	Количество отрицательных примеров
numStages	16	Количество уровней обучения
MinHitRate	0,995	Минимальная доля истинно
		положительных примеров на уровень
MaxFalseAlarm	0,4	Максимальная доля ложно положительных
		примеров на уровень
Mode	All	Использование полного набора метрик
		Xaapa
Width, Height	20	Разрешение положительных примеров
BoostTypes	GAB	Модификация Gentle AdaBoost

Основные параметры алгоритма детектирования человеческих голов на основе бустинга приведены в табл. 4.6.

Таблица 4.6 Основные параметры алгоритма детектирования бустинга

Параметр	Значение	Описание
scaleFactor	1,27	Шаг масштабирования изображения
minNeighbors	5	Количество соседних выделений
minSize	60×60	Минимальный размер объекта
		в пикселях
maxSize	200×200	Максимальный размер объекта
		в пикселях

Результаты тестирования алгоритма детектирования человеческих голов с купольных камер на основе бустинга приведены в табл. 4.7.

 Таблица 4.7

 Результаты тестирования алгоритма детектирования бустинга

Тестовая база	Recall, %	Precision, %	F-мера, %
«Иваново»	94,6	97,8	96,2
«Кафедра»	75,4	86,0	80,4
«Школа»	70,8	69,0	69,9
Среднее значение	84,3	86,8	85,5

## 4.7. Разработка алгоритма на основе локальных бинарных шаблонов

Реализован и протестирован каскадный классификатор с ЛБШ признаками аналогично предыдущему подходу. Параметры обучения приведены в табл. 4.8.

Таблица 4.8 Параметры обучения каскадного классификатора LBP

Параметр	Значение	Описание
numPos	3400	Количество положительных примеров
numNeg	4187	Количество отрицательных примеров
numStages	13	Количество уровней обучения
Width, Height	20	Разрешение положительных примеров
featureType	LBP	Обучение с использованием LBP признаков

Основные параметры алгоритма детектирования человеческих голов на основе ЛБШ признаков совпадают с аналогичными параметрами алгоритма на основе бустинга, исключение составляет использование функции выравнивания гистограмм распределения пикселей относительно значений их интенсивности тестовых изображений (рис. 4.10).

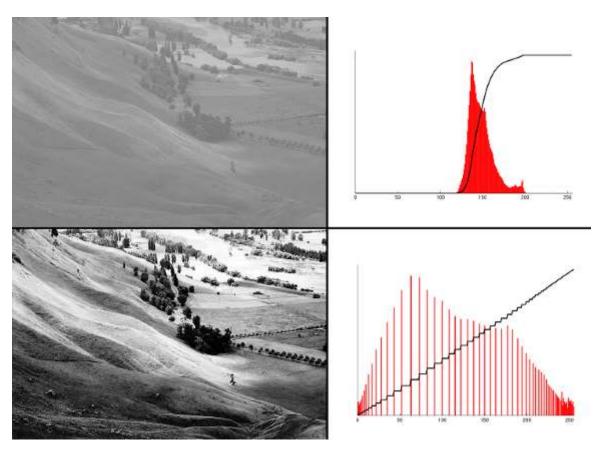


Рис. 4.10. Пример выравнивания гистограммы изображения

Применение данной функции для алгоритма бустинга оказывало негативное влияние на чувствительность классификатора.

Полученные результаты тестирования алгоритма детектирования человеческих голов на основе ЛБШ признаков приведено в табл. 4.9.

 Таблица 4.9

 Результаты тестирования алгоритма детектирования ЛБШ

Тестовая база	Recall, %	Precision, %	F-мера, %
«Иваново»	88,2	89,5	88,9
«Кафедра»	58,3	77,8	66,7
«Школа»	54,1	51,7	52,9
Среднее значение	75,6	75,4	75,5

Стоит отметить также, что основным достоинством применения ЛБШ признаков является увеличение скорости работы в 2–2,5 раза.

# 4.8. Разработка алгоритма постклассификации ложноположительных результатов

Для улучшения качества работы алгоритмов детектирования на основе бустинга и ЛБШ признаков, был реализован дополнительный классификатор на основе НОG-признаков (рис. 4.11).



Рис. 4.11. Блок-схема алгоритма с постклассификацией ложноположительных результатов

На вход алгоритма постклассификации подаются истинно положительные и ложно положительные результаты с алгоритмов классификации на основе бустинга или ЛБШ признаков. Необходимо выделить из всей выборки только истинно положительные результаты.

На первом этапе изображение приводится к разрешению  $100 \times 100$  пикселей. Затем вычисляются НОG-признаки и подаются в качестве дескриптора на машину опорных векторов (рис. 4.12).

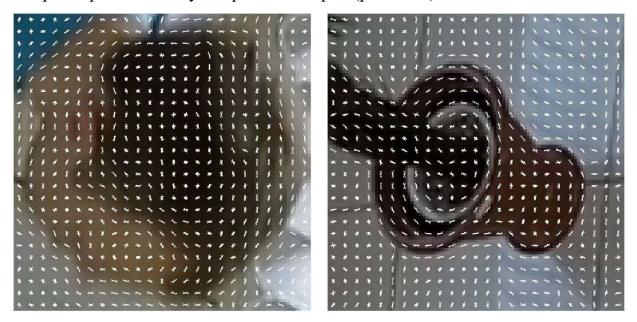


Рис. 4.12. Визуализация НОG-признаков для истинно положительных и ложно положительных результатов

Основные параметры алгоритма постклассификации перечислены в табл. 4.10.

Таблица 4.10 Основные параметры алгоритма постклассификации на основе НОG-признаков

Параметр	Значение	Описание
CellSize	4×4	Размер ячейки в пикселях
BlockSize	2×2	Размер блока в ячейках
BlockType	R-HOG	Прямоугольная форма блока
positiveData	900	Количество положительных примеров
negativeData	900	Количество отрицательных примеров
kernel_function	linear	Линейная ядерная функция
method	SMO	Метод поиска разделяющей
		гиперплоскости – последовательная
		минимальная оптимизация.

Тестирование на 100 изображениях истинно положительных и ложно положительных результатов показало уровень F-меры, близкий к 100%.

# 4.9. Сравнение алгоритмов бустинга и ЛБШ с применением и без применения алгоритма постклассификации

Результаты тестирования алгоритмов на основе бустинга и ЛБШ признаков с применением и без применения алгоритма постклассификации приведены в табл. 4.11.

Таблица 4.11 Результаты тестирования алгоритмов бустинга и ЛБШ признаков с применением и без применения алгоритма постклассификации

Алгоритм	Recall, %	Precision, %	F-мера, %
Бустинг	84,3	86,8	85,5
ЛБШ	75,6	75,4	75,5
Бустинг+HOG	84,3	98,5	90,8
ЛБШ+HOG	75,6	98,0	85,3

Видно, что применение дополнительного классификатора на основе гистограмм направленных градиентов позволило увеличить F-меру для бустинга на 6%, а для ЛБШ на 10%.

# 4.10. Тестирование алгоритма на основе бустинга для спортивной видеоаналитики

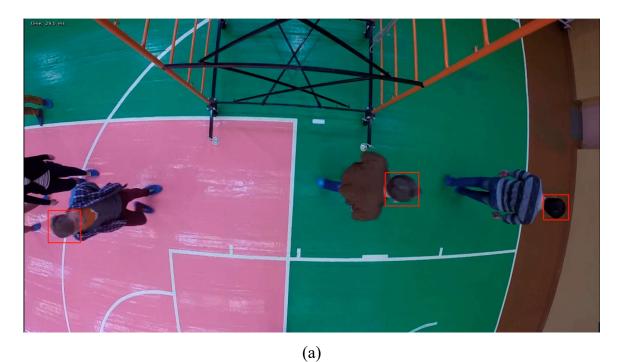
Для тестирования алгоритма проведены съемки в спортивном зале при различных условиях:

- при различной высоте подвеса камеры;
- при различных скоростях передвижения людей;
- при разных уровнях освещения.

Съемки производились на экшн-камеру GoPro HERO3+ Silver Edition при параметрах записи видео, указанных в табл. 4.12. Пример кадров из тестовой видеопоследовательности приведен на рис. 4.13.

Таблица 4.12 Параметры тестовых видео для спортивного зала

Параметр	Значение
Разрешение видео	1920×1080
Частота кадров в секунду	30
Баланс белого	Автоматический
Угол обзора	Wide(≈150°)



(6)

Рис. 4.13. Примеры обработанных кадров видеопоследовательности

Для оценки качества работы алгоритма к основному алгоритму детектирования людей добавлен алгоритм трекинга на основе алгоритма Лукаса-Канаде. В качестве его реализации взята версия из открытой библиотеки OpenCV. Производился подсчет количества верно детектированных людей из общего их количества, при этом общее количество людей в каждом из тестов – 100 человек.

Результаты тестирования при различной высоте подвеса приведены в табл. 4.13.

Таблица 4.13 Результаты тестирования алгоритмов бустинга для различных высот подвеса камеры

Высота	Процент подсчитанных людей
2,6 м	76%
3,2 м	83%
7 м	81%

При различных уровнях подвеса камеры алгоритм дает сопоставимые результаты по качеству работы, но с увеличением высоты подвеса падает скорость его работы. Результаты тестирования при различных скоростях передвижения людей приведены в табл. 4.14.

Таблица 4.14 Результаты тестирования алгоритмов бустинга для различных скоростей передвижения людей при высоте подвеса 3,2 м

Скорость передвижения	Процент подсчитанных людей
Медленный темп ходьбы	93%
Быстрый темп ходьбы	83%
Легкий бег	80%

Ухудшение точности работы алгоритма детектирования при переходе на легкий бег связано с неестественным положением волос у девушек, а также появляющимися при увеличении скорости передвижения объектов их взаимными пересечениями.

Результаты тестирования при различных уровнях освещения приведены в табл. 4.15.

Таблица 4.15 Результаты тестирования алгоритмов бустинга для различных уровней освещения при высоте подвеса 3,2 м

Освещение	Процент подсчитанных людей
Без дополнительного освещения	86%
С дополнительным освещением	83%

В данном случае увеличение точности работы алгоритма в отсутствии неестественного освещения связана с тем, что в обучающей базе освещение было ближе к естественному освещению спортивного зала.

### 4.11. Краткие выводы

Результаты проведенных исследований позволяют сделать следующие основные выводы:

- 1. Подготовлены различные тестовые наборы видеоизображений с вертикально расположенных купольных камер.
- 2. Разработаны и исследованы два алгоритма детектирования голов на видеоизображениях с купольных камер (на основе бустинга и ЛБШ). Алгоритм на основе бустинга показывает более высокие результаты по средним значениям метрик: Recall 84,3%, Precision 86,8%, F-мера 85,5%. Достоинством алгоритма ЛБШ является повышение скорости работы в 2-2,5 раза.
- 3. Разработан и исследован алгоритм постклассификации истинно положительных и ложноположительных результатов на основе вычисления гистограммы направленных градиентов. Его применение позволяет для алгоритма на основе бустинга повысить значение метрики Precision на 12%, F-меры на 5%, а для алгоритма на базе локальных бинарных шаблонов повысить значение метрики Precision на 23%, F-меры на 10%.
- 4. Установлено, что уровень подвеса камеры при моделировании спортивной съемки не оказывает существенного влияния на работу предложенного алгоритма детектирования на основе бустинга. Также установлено, что увеличение скорости передвижения людей (с ходьбы на легкий бег) снижает процент подсчитанных людей на 13%.

### ЗАКЛЮЧЕНИЕ

Основные выводы и результаты диссертационной работы можно сформулировать в следующем виде.

- 1. Разработана методика и программное обеспечение для исследования алгоритмов детектирования человека на телевизионных изображениях при различных ракурсах видеокамер.
- 2. Разработанный алгоритм на основе АРД увеличивает площадь под ROC-кривой на 13% по сравнению со стандартным алгоритмом детектирования Виолы-Джонса. Реализация алгоритма детектирования на основе локальных бинарных шаблонов из библиотеки Open CV показывает еще более худшие результаты.
- 3. Исследования робастности алгоритма детектирования на основе АРД показывают, что АБГШ, мультипликативный и импульсный шум вида «соль-и-перец» оказывают сильное влияние на работу алгоритма (уменьшение площади под ROC-кривой на 21,2% при снижении метрики PSNR до 17,99 дБ для АБГШ, на 8,8% при снижении метрики PSNR до 15,71 дБ для импульсного шума «соль-и-перец» и на 18,4% при снижении метрики PSNR до 18,75 дБ для мультипликативного шума).
- 4. Исследования робастности алгоритма детектирования на основе АРД показывают, что искажения типа размытие и сжатие алгоритмами JPEG и JPEG2000 оказывают слабое влияние на работу алгоритма (уменьшение площади под ROC-кривой на 3,5% при снижении метрики PSNR до 21,58 дБ для размытия, на 7,5% при снижении метрики до PSNR до 24,73 дБ для JPEG и на 0,3% при снижении метрики PSNR до 31,79 дБ для JPEG2000).
- 5. Модификация алгоритма детектирования фигуры человека на основе метода Далала-Триггса с использованием классификатора на базе бустинга и переобучения на сложных примерах позволила значительно

повысить результаты по метрике Recall (с 83 % до 93 % для базы INRIA Person и с 65 % до 91 % для базы CVC-02 Pedestrian). Использование модифицированного алгоритма позволяет достичь также увеличение метрики Precision (с 57 % до 67 % для базы INRIA Person и с 25 % до 26 % для базы CVC-02 Pedestrian).

- Модифицированный алгоритм детектирования фигуры человека показывает большую робастность к присутствию на изображении гауссовского шума, что особенно заметно на средних степенях зашумления, при σ = 10–20.
- 7. Подготовлены наборы видеоизображений различные тестовые  $\mathbf{c}$ Разработаны вертикально расположенных купольных камер. исследованы два алгоритма детектирования голов на видеоизображениях с купольных камер (на основе бустинга и ЛБШ). Алгоритм на основе бустинга показывает более высокие результаты по средним значениям метрик: Recall -84,3%, Precision -86,8%, F-мера -85,5%. Достоинством алгоритма ЛБШ является повышение скорости работы в 2-2,5 раза.
- 8. Разработан исследован алгоритм постклассификации истинно ложноположительных положительных И результатов на основе вычисления гистограммы направленных градиентов. Его применение позволяет для алгоритма на основе бустинга повысить значение метрики Precision на 12%, F-меры на 5%, а для алгоритма на базе локальных бинарных шаблонов повысить значение метрики Precision на 23%, F-меры на 10%.
- 9. Установлено, что уровень подвеса камеры при моделировании спортивной съемки не оказывает существенного влияния на работу предложенного алгоритма детектирования на основе бустинга. Также установлено, что увеличение скорости передвижения людей (с ходьбы на легкий бег) снижает процент подсчитанных людей на 13%.

10.Цель работы задачи диссертационной успешно выполнены. Разработанные алгоритмы детектирования объектов интегрированы в программу VIStat 1.0 для автоматизированного расчета и визуализации статистики, которая протестирована на ряде тестовых видеоданных, характерных для систем прикладного телевидения. Получено свидетельство о государственной регистрации программы для ЭВМ.

### СПИСОК ЛИТЕРАТУРЫ

- 1. Дворкович В.П., Дворкович А.В. Цифровые видеоинформационные системы (теория и практика). М.: Техносфера, 2012. 1009 с.
- 2. Безруков В.Н., Балобанов В.Г. Системы цифрового вещательного и прикладного телевидения. Учебное пособие для вузов. М.: Горячая линия Телеком, 2015. 608 с.
- 3. Форсайт Д.А., Понс Д. Компьютерное зрение. Современный подход. М.: Вильямс, 2004. 928 с.
- 4. Лукьяница А.А., Шишкин А.Г. Цифровая обработка видеоизображений. М.: Ай-Эс-Эс Пресс, 2009. 518 с.
- 5. Кухарев Г.А. Методы обработки и распознавания изображений лиц в задачах биометрии. СПб.: Политехника, 2013. 388 с.
- 6. Гонсалес Р., Вудс Р. Цифровая обработка изображений. М.: Техносфера, 2005. – 621 с.
- 7. Методы компьютерной обработки изображений / Под ред. В.А. Сойфера. – М.: Физматлит, 2001. – 784 с.
- 8. Прэтт У. Цифровая обработка изображений. М.: Мир, 1982.  $311+479~\mathrm{c}.$
- 9. Журавлев Ю.И., Рязанов В.В., Сенько О.В. Распознавание. Математические методы. Программная система. Практические применения. М.: Фазис, 2005. 159 с.
- 10. Вапник В.Н., Червоненкис А.Я. Теория распознавания образов (статистические проблемы обучения). М.: Наука, 1974. 416 с.

- 11. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features // Proc. International Conference on Computer Vision and Pattern Recognition. 2001. № 1. P. 511–518.
- Dalal N., Triggs B. Histograms of Oriented Gradients for Human Detection // Proc. of the IEEE Conference Computer Vision and Pattern Recognition. 2005. P. 886–893.
- 13. Kanade T. Picture processing system by computer complex and recognition of human faces. PhD thesis. Kyoto University, 1973.
- 14. Голубев М.Н. Разработка и анализ алгоритмов детектирования и классификации объектов на основе методов машинного обучения // Диссертация на соискание ученой степени кандидата технических наук. Ярославль. 2012.
- 15. Open Source Computer Vision Library (OpenCV), http://opencv.org.
- Howse J. OpenCV Computer vision with Python. Packt Publishing Ltd.,
   UK. 2013. 122 p.
- 17. Сэломон Д. Сжатие данных, изображений и звука. М.: Техносфера, 2004. 368c.
- 18. Кругль Г. Профессиональное видеонаблюдение. Практика и технологии аналогового и цифрового ССТV. М.: Секьюрити Фокус, 2010. 640 с.
- 19. Дамьяновски В. ССТV. Библия видеонаблюдения. Цифровые и сетевые технологии. М.: Ай-Эс-Эс Пресс, 2006. 480 с.
- Концепция построения и развития аппаратно-программного комплекса «Безопасный город» // Распоряжение Правительства Российской Федерации от 3.12.2014 № 2446-р.

- 21. Трубин И.С., Медведева Е.В., Булыгина О.П. Нелинейная фильтрация видеопоследовательностей цифровых полутоновых изображений // Инфокоммуникационные технологии Том 5, №4, 2007 29-35.
- 22. Ганин А.Н. Сопровождение и распознавание объектов на телевизионных изображениях // Диссертация на соискание ученой степени кандидата технических наук. Ярославль. 2013.
- 23. Seemann E., Leibe B., Mikolajczyk K., and Schiele B., "An evaluation of local shape-based features for pedestrian detection," in British Machine Vision Conf., 2005.
- 24. Gu C., Lim J., Arbelaez P., and Malik J., "Recognition using regions," in IEEE Conf. Computer Vision and Pattern Recognition, 2009.
- 25. Felzenszwalb P.F., Girshick R.B., McAllester D., Ramanan D. Object Detection with Discriminatively Trained Part Based Models // Pattern Analysis and Machine Intelligence. 2010. V. 32, № .9. P. 1627-1645.
- 26. Dalal N., Triggs B. Histograms of oriented gradients for human detection // Computer Vision and Pattern Recognition (CVPR). 2005. V. 1. P. 886-893.
- 27. Bay H., Tuytelaars T., and Van Gool L.. SURF: Speeded Up Robust Features // Computer Vision and Image Understanding, 2008, V. 110, № 3, P. 346-359.
- 28. Jones M., Viola P. Fast Multi-view Face Detection // Mitsubishi Electric Research Laboratories TR2003-96, 2003, V.3.
- 29. Ahonen T., Hadid A., Pietikäinen M. Face Recognition with Local Binary Patterns // Proc. 8th European Conference on Computer Vision (ECCV), 2004, P. 469–481.

- 30. Jones M., Viola P. Rapid object detection using a boosted cascade of simple features // Proc. International Conference on Computer Vision and Pattern Recognition, 2001, № 1, P. 511-518.
- 31. Zhang C. and Zhang Z. A survey of recent advances in face detection // Technical report, Microsoft Research, 2010.
- 32. Osuna E., Freund R., Girosi F. Training Support Vector Machines: An Application to Face Detection // Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1997, P. 130-136.
- Sung K., Poggio T. Example-Based Learning for View-Based Human Face Detection // Technical Report AI Memo 1521, Massachusetts Inst. of Technology AI Lab, 1994.
- 34. Oren M., Papageorgiou C., Sinha P., Osuna E., Poggio T. Pedestrian Detection Using Wavelet Templates // Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1997, P. 193-199.
- 35. Papageorgiou C., Oren M., Poggio T. A General Framework for Object Detection // Proc. 6th IEEE International Conference on Computer Vision, 1998, P. 555-562.
- 36. Papageorgiou C., Poggio T. A Trainable System for Object Recognition //
  International Journal Computer Vision, 2000, V. 38, № 1, P. 15-33.
- 37. Kienzle W., Bakir G., Franz M., Schölkopf B. Face Detection Efficient and Rank Deficient // Advances in Neural Information Processing Systems, 2005. V.17, P. 673-680.
- 38. Graf H., Cosatto E., Bottou L., Durdanovic I., Vapnik V. Parallel Support Vector Machines: The Cascade SVM // Proc. Conference on Neural Information Processing Systems (NIPS), 2004, P. 521-528.

- 39. Leung T., Burl M., Perona P. Finding Faces in Cluttered Scenes Using Random Labeled Graph Matching // Proc. 5th International IEEE Conference on Computer Vision, 1995, P. 637-644.
- 40. Rowley H., Baluja S., Kanade T. Neural Network-Based Face Detection // Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1996, P. 203-208.
- 41. LeCun Y., Bottou L., Bengio Y., Haffner P. Gradient-based learning applied to document recognition // Proceedings of the IEEE, 1998, V. 86, P. 2278-2324.
- 42. Конушин А.С. Слежение за точечными особенностями сцены // Компьютерная графика и мультимедиа. Выпуск №1(5). 2003.
- 43. Khryashchev V., Shmaglit L., Shemyakov A., Lebedev A. Gender classification for real-time audience analysis system // Proc. IEEE 15th Conf. Open Innovations Association (FRUCT15), 2014, P. 52-59.
- 44. Khryashchev V., Ganin A., Stepanova O., Lebedev A. Age estimation from face images: challenging problem for audience measurement systems // Proc. IEEE Conf. Open Innovations Association (FRUCT16), 2014, P. 31-37.
- 45. Лебедев А.А, Матвеев Д.В., Хрящев В.В. Обработка видеопотока с потолочных камер. Цифровая обработка сигналов и ее применение (DSPA-2015): докл. 17-й междунар. конф. Москва, 2015. Т. 2. С. 703-707.
- 46. Хрящев В.В., Лебедев А.А., Распознавание лиц на изображениях при наличии искажений. Системы синхронизации, формирования и обработки сигналов в инфокоммуникациях (СИНХРОИНФО 2015): Труды междунар. конф. Санкт-Петербург, 2015. С. 194-196.

- 47. Лебедев А.А. Разработка программно-аппаратного комплекса для анализа видеоданных с купольных камер. Материалы 2-ой Международной научно-практической конференции Путь в науку Ярославль, 2014. С. 84.
- 48. Лебедев А.А. Разработка алгоритмов для анализа видеоданных с купольных камер. 67-ая всероссийская научно-техническая конференция студентов, магистрантов и аспирантов с международным участием: сборник тезисов конф. Ярославль, 2014. С. 302.
- 49. Eliazar I., Sokolov I. Measuring statistical heterogeneity: The Pietra index. // Physica A-Statistical Mechanics and Its Applications No. 389, 2010, P. 117–125.
- 50. Wen-Chung Lee. Probabilistic Analysis of Global Performances of Diagnostic Tests: Interpreting the Lorenz Curve-Based Summary Measures. // Statistics in Medicine No. 18, 1999, P. 455–471.
- Quinlan, J. R. Induction of Decision Trees. // Machine Learning No. 1, 1986,
   P. 81-106.
- 52. Laptev I., Marszalek M., Schmid C., and Rozenfeld B.. Learning realistic human actions from movies. In Proceedings CVPR. IEEE, 2008.
- 53. Harris C. and Stephens M.. A combined corner and edge detector. In Proceedings of Fourth Alvey Vision Conference, pages 147–151, 1988.
- 54. Laptev I. and Lindeberg T. Interest point detection and scale selection in spacetime. In Proceedings of the International Conference on Scale Space Methods in Computer Vision, pages 372–387. Springer-Verlag, 2003.
- 55. Lowe D.. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91–110, November 2004.

- 56. Scovanner P., Saad A., and Shah M. A 3-dimensional sift descriptor and its application to action recognition. In Proceedings of the 15th International Conference on Multimedia, pages 357–360. ACM, 2007.
- 57. Lan T., Wang Y., Mori G., and Stephen N. Robinovitch. Retrieving actions in group contexts. In Trends and Topics in Computer Vision, P 181–194. Springer, 2012.
- 58. Choi W., Shahid K, and Savarese S. Learning context for collective activity recognition. In Proceedings CVPR, P 3273–3280. IEEE, 2011.
- 59. Zhu Y., M. Nayak N., and Amit K. Roy-Chowdhury. Context-aware modeling and recognition of activities in video. In In Proceedings CVPR. IEEE, June 2013.
- 60. Alina Bialkowski, Patrick Lucey, Peter Carr, Simon Denman, Iain Matthews, and Sridha Sridharan. Recognising team activities from noisy data. In Proceedings CVPR Workshops, pages 984–990. IEEE, 2013.
- 61. Matej Perše, Matej Kristan, Stanislav Kovačič, and Janez Perš. A trajectory-based analysis of coordinated team activity in basketball game. Computer Vision and Image Understanding, 113(5):612–621, 2008.
- 62. Matej Perše, Matej Kristan, Janez Perš, and Stanislav Kovačič. A template-based multiplayer action recognition of the basketball game. In Proceedings ECCV Workshop on Computer Vision Based Analysis in Sport Environments, P 71–82. Springer, 2006.
- 63. Rikke Gade and Thomas B. Moeslund. Sports type classification using signature heatmaps. In Proceedings CVPR Workshops, pages 999–1004. IEEE, June 2013.
- 64. http://vis-www.cs.umass.edu/fddb/results.html. Исследование в области детектирования лиц Массачусетского технологического института.

- 65. Markus N., Frljak M., Pandzic I., Ahlberg J., Forchheimer R. A Method for Object Detection Based on Pixel Intensity Comparisons Organized in Decision Trees // Computing Research Repository, 2014.
- 66. Amit Y. and Geman D.. Shape quantization and recognition with randomized trees // Neural Computation, 1997, V.9, № 7, P. 1545–1588.
- 67. Shotton J., Fitzgibbon A., Cook M., Sharp T., Finocchio M., Moore R., Kipman A., Blake A. Real-time human pose recognition in parts from single depth images // Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.
- 68. Ozuysal M., Fua P., Lepetit V. Fast keypoint recognition in ten lines of code // Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007, P. 1–8.
- 69. Markus N., Frljak M., Pandzic I., Ahlberg J., Forchheimer R. Eye pupil localization with an ensemble of randomized trees // Pattern Recognition, 2014, V. 47, №2, P. 578–578.
- 70. Friedman J., Hastie T., Tibshirani R. Additive logistic regression: a statistical view of boosting // The Annals of Statistics, 2000, V. 28, № 2, P. 337-407.
- 71. http://lrs.icg.tugraz.at/research/aflw/downloads.php Набор изображений AFLW.
- 72. http://robotics.csie.ncku.edu.tw/Databases/FaceDetect\_PoseEstimate.htm#Ou r\_Database. Набор изображений Robotics.
- 73. http://www.ics.uci.edu/~xzhu/face/. Набор изображений AFW.
- 74. Fawcett T.. An introduction to ROC analysis // Pattern Recognition Letters, 2006, V. 27, № 8, 861–874.
- 75. Breiman L. Bagging Predictors // Technical Report No. 421, 1994.

- 76. Geronimo D., Lopez A.M., Sappa A.D., Graf T. Survey of Pedestrian Detection for Advanced Driver Assistance Systems // IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 7, 2010. P. 1239–1258.
- 77. Oren M., Papageorgiou C., Sinha P., Osuna E., Poggio T. Pedestrian detection using wavelet templates // Conference on Computer Vision and Pattern Recognition (CVPR), 1997. P. 193–199.
- 78. Papageorgiou C., Poggio T. A trainable system for object detection // International Journal of Computer Vision, 38(1), 2000. P. 15–33.
- 79. Приоров А.Л., Игнатов И.С., Голубев М.Н., Хрящев В.В. Разработка и анализ алгоритмов выделения лиц на изображениях // Проектирование и технология электронных средств. 2008. № 2. С. 58–62.
- 80. Lowe D. Distinctive image features from scale invariant keypoints //
  International Journal of Computer Vision, 60(2), 2004. P. 91–110.
- 81. Shashua A., Gdalyahu Y., and Hayon G. Pedestrian detection for driving assistance systems: Single-frame classification and system level performance // IEEE Intelligent Vehicles Symposium, 2004. P. 1–6.
- 82. Dalal N., Triggs B., Schimid C. Human detection using oriented histograms of flow and appearance // European Conference on Computer Vision (ECCV), 2006. P. 428–441.
- 83. Zhu Q., Avidan S., Yeh M.C., Cheng K.T. Fast human detection using a cascade of histograms of oriented gradients // Conference on Computer Vision and Pattern Recognition (CVPR), 2006. P. 1491–1498.
- 84. Mikolajczyk K., Schmid C., and Zisserman A. Human detection based on a probabilistic assembly of robust part detectors // European Conference on Computer Vision (ECCV), 2004. P. 69–82.

- 85. Dollar P., Wojek C., Schiele B., Perona P. Pedestrian detection: a benchmark // Conference on Computer Vision and Pattern Recognition (CVPR), 2009. P. 304-311.
- 86. Казаков А., Бовырин А. Быстрый алгоритм обнаружения пешеходов по видеоданным // International Conference on Computer Graphics and Vision, GraphiCon, 2012. P. 144–148.
- 87. Cortes C., Vapnik V. Support-Vector Networks // Machine Learning, 20, 1995. P. 273-297.
- 88. Freund Y., Schapire R.E. A short introduction to boosting // Journal of Japanese Society for Artificial Intelligence, 14(5):771-780, September, 1999.
- 89. Olson D.L., Delen D. Advanced data mining techniques. Berlin: Springer-Verlag, 2008. 180 p.
- 90. INRIA Person Dataset (http://pascal.inrialpes.fr/data/human/). Набор изображений INRIA.
- 91. Лебедев А.А, Хрящев В.В., Шмаглит Л.А. Разработка алгоритмов анализа видеопотока с купольных камер // Радиоэлектронные средства получения, обработки и визуализации информации (РСПОВИ-2014): докл. 4-й всерос. конф. Нижний Новгород, 2014. С. 139–142.
- 92. Belongie S., Malik J. Shape Matching and Object Recognition Using Shape Contexts // IEEE Transactions on pattern analysis and machine intelligence, vol. 24, No. 24, April 2002.
- 93. Jones D. and Malik J. Computational Framework to Determining Stereo Correspondence from a Set of Linear Spatial Filters // Image and Vision Computing, vol. 10, no. 10, pp. 699–708, December 1992.

- 94. Canny J. A Computational Approach to Edge Detection // IEEE Transactions on pattern analysis and machine intelligence, vol. PAMI-8, no. 6, November 1986.
- 95. Bagdonavicius V.B. and Nikulin M.S. Chi-square goodness-of-fit test for right censored data // The International Journal of Applied Mathematics and Statistics, pp. 30–50, 2011.
- 96. Jonker R. and Volgenant A. A Shortest Augmenting Path Algorithm for Dense and Sparse Linear Assignment Problems // Computing, vol. 38, pp. 325–340, 1987.
- 97. Wahba G. Spline models for observational data // Philadelphia: Society for Industrial and Applied Mathematics, 1990.
- 98. Лебедев А.А., Хрящев В.В. Анализ видеопотока с потолочных камер методом контекста формы // 68-я всероссийская научно-техническая конференция студентов, магистрантов и аспирантов высших учебных заведений с международным участием: сборник материалов конф. Ярославль, 2015. С. 1367–1370.
- 99. Шмаглит Л.А., Хрящев В.В., Ганин А.Н., Матвеев Д.В. Распознавание пола человека по выделенной области лица на изображениях // Всероссийский научно-технический журнал "Проектирование и технологии электронных средств". 2011. № 4. С. 36-43.
- 100. Шмаглит Л.А., Приоров А.Л., Хрящев В.В., Матвеев Д.В. Детектирование лиц на изображениях в условиях аддитивного белого гауссовского шума // Электромагнитные волны и электронные системы. -2014.-T.19, №5. -C.62-70.

- 101. Никитин А.Е., Хрящев В.В., Приоров А.Л., Матвеев Д.В. Разработка и анализ алгоритма распознавания лиц на основе локальных квантованных шаблонов // Нелинейный мир. 2014. № 8. С. 35–42.
- 102. Приоров А.Л., Хрящев В.В., Никитин А.Е., Матвеев Д.В. Анализ и модификация алгоритмов детектирования пешеходов на цифровых изображениях // Радиотехнические и телекоммуникационные системы. 2014. № 4 (16). С. 40-47.
- 103. Матвеев Д.В., Аминова Е.А., Ганин А.Н. Алгоритм отслеживания движения точечных объектов // Докл. 13-й междунар. конф. «Цифровая обработка сигналов и ее применение» (DSPA-2011), Москва, 2011. Т.1. С. 275-277.
- 104. Ганин А.Н., Вдовченко П.В., Матвеев Д.В. Разработка и анализ алгоритмов гендерной классификации по изображениям лиц // Матер. междунар. науч.-практ. конф. "Молодежь и наука: модернизация и инновационное развитие страны". Пенза, 2011 г., Ч. 1, С. 221-223.
- 105. Ганин А.Н., Кисельникова О.А., Матвеев Д.В., Носков А.А. Применение метода цифровых гистограмм для детектирования повторного появления объекта на видеоданных // Цифровая обработка сигналов и ее применение (DSPA-2013): докл. 15-й междунар. конф. Москва, 2013. Т. 2. С. 96-100.
- 106. Лебедев А.А., Матвеев Д.В., Никитин А.Е., Хрящев В.В., Шемяков А.М. Анализ и модификация алгоритмов детектирования пешеходов на цифровых изображениях // Цифровая обработка сигналов и ее применение (DSPA-2014): докл. 16-й междунар. конф. Москва, 2014. Т. 2. С. 560-564.
- 107. Матвеев Д.В. Разработка и анализ алгоритмов детектирования людей на видеоданных // Радиоэлектронные устройства и системы для

- инфокоммуникационных технологий РЭУС-2015: сб. докл. междунар. конф. Москва, 2015. С. 338-341.
- 108. Хрящев В.В., Приоров А.Л., Матвеев Д.В. Модификация алгоритма Далала-Тригтса для обнаружения пешеходов на видеоданных // Оптико-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации. Распознавание 2015: сб. материалов XII Междунар. науч.-техн.конф. Курск, 2015. С. 383-385.
- 109. Хрящев В.В., Ганин А.Н., Манов И.А., Матвеев Д.В. Система анализа аудитории на основе алгоритмов детектирования, слежения и гендерной классификации // II всерос. науч. конф. молодых ученых с международным участием: сб. тр. Рыбинск, 2012. Т.2. С. 136-142.
- 110. Матвеев Д.В. Никитин А.Е. Алгоритм Далала-Триггса его модификация человека ДЛЯ детектирования на видеоданных // Радиоэлектронные средства получения, обработки и визуализации информации: докл. 4-й всероссийской конференции. - Нижний Новгород, 2014. С. 196-201.
- 111. Матвеев Д.В., Лебедев А.А., Хрящев В.В. Разработка алгоритмов детектирования объектов с потолочных камер видеонаблюдения // Радиоэлектронные средства получения, обработки и визуализации информации: докл. 5-ой всероссийской конференции. Москва, 2015. С. 135-138.
- 112. Матвеев Д.В. Разработка алгоритмов детектирования людей на видеоданных с купольных камер // Перспективные технологии в средствах передачи информации ПТСПИ-2015: докл. 11-ой международной научно-технической конференции. Суздаль, 2015. С. 164-166.

- 113. Матвеев Д.В. Об одном алгоритме распознавания движения на последовательности кадров // Математические методы распознавания образов: сборник докладов 14-ой Всероссийской конференции М.: Макс-Пресс, 2009. С. 408–410.
- 114. Хрящев В.В., Приоров А.Л., Никитин А.Е., Матвеев Д.В. Алгоритм распознавания лиц с использованием информации о расположении центра глаз // Радиолокация, Навигация, Связь: Сборник трудов XXI Международной научно-технической конференции. Воронеж, 2015. Т. 1. С. 177-187.
- 115. Хрящев В.В., Матвеев Д.В., Ганин А.Н., Казина Е.М. VIStat 1.0 автоматизированная система расчета и визуализации статистики // Свидетельство о государственной регистрации программы для ЭВМ №2013612188 от 18 февраля 2013.

# ПРИЛОЖЕНИЕ 1. АКТЫ ВНЕДРЕНИЯ РЕЗУЛЬТАТОВ РАБОТЫ

«УТВЕРЖДАЮ»

Первый проректор Ярославского государственного университета

им. П.Г. Демидова

С.А. Кащенко

2015 г.

### AKT

### внедрения результатов диссертационной работы Матвеева Дмитрия Вячеславовича на тему

«Разработка алгоритмов анализа аудитории для систем прикладного телевидения» в учебный процесс

Мы, нижеподписавшиеся, заведующий кафедрой динамики электронных систем, профессор, д.т.н. Брюханов Ю.А. и доцент кафедры динамики электронных систем, к.т.н. Волохов В.А. составили настоящий акт о том, что результаты диссертационной работы Матвеева Д.В. внедрены в учебный процесс на кафедре динамики электронных систем физического факультета ЯрГУ (специальность «Радиотехника»):

- в курсе «Цифровая обработка изображений» алгоритм детектирования лиц на основе каскада ансамблей решающих деревьев;
- в курсе «Системы технического зрения» результаты исследования алгоритмов детектирования пешеходов на видеопоследовательностях в условиях искажений и помех;
- при выполнении студентами курсовых и выпускных квалификационных работ используется разработанная Матвеевым Д.В. методика для исследования алгоритмов детектирования объектов на телевизионных изображениях.

Заведующий кафедрой динамики электронных систем, д.т.н., профессор

Ю.А. Брюханов

Доцент кафедры динамики электронных систем, к.т.н.

front.

В.А. Волохов

### Общество с ограниченной ответственностью «Пиклаб»



Юридический адрес: 150040, г. Ярославль, пр. Октября, 56, оф. 406. Почтовый адрес: 150040, г. Ярославль, пр. Октября, 56, оф. 406. тел: +7 (4852) 64 07 06, e-mail: <a href="mailto:connect@piclab.ru">connect@piclab.ru</a> ИНН 7606083311 КПП 760601001 БИК 047888728.

**УТВЕРЖДАЮ** 

Директор ООО «Пиклаб»

Саутов Е.Ю.

«20» ноября 2015 г.

AKT

внедрения результатов диссертационной работы Матвеева Дмитрия Вячеславовича, выполненной в Ярославском государственном университете имени П.Г. Демидова (ЯрГУ), на тему «Разработка алгоритмов анализа аудитории для систем прикладного телевидения».

Комиссия в составе: председатель комиссии — директор по развитию Хрящев В.В., члены комиссии — руководитель направления видеоаналитики Храбров Д.Е., ведущий инженер Чарчан А.В., рассмотрев диссертационную работу Матвеева Д.В. составила настоящий акт о том, что ее результаты нашли применение в работе малого инновационного предприятия ООО «Пиклаб». Особый практический интерес представляет следующий результат диссертации:

 алгоритм детектирования объектов на видеоданных, полученных с помощью вертикально расположенных камер, на основе бустинга с дополнительным классификатором на основе НОG-признаков.

Данный алгоритм использован при разработке программного-аппаратного комплекса — Piclab.Video. Использование предложенного Матвеевым Д.В. алгоритма позволило оптимизировать параметры системы для разных условий освещения объектов и высоты подвеса видеокамеры.

Председатель комиссии

Члены комиссии

Хрящев В.В.

У Храбров Д.Е.

Чарчан А.В.



г. Ярославль, Ленинградский пр. 33, офис 209-А Тел/Факс (4852)-58-03-36, (4852)-58-03-37 E-Mail: comp@yaroslavl.ru

### 000 "БАЙТ-ПРО"

ИНН 7602104218 КПП 760201001 Р/Сч. № 40702810340100000205 в ОАО АКБ "Авангард" г.Москва БИК 044525201 Кор.Сч. 30101810000000000201

### AKT

внедрения результатов диссертационной работы Матвеева Дмитрия Вячеславовича, выполненной в Ярославском государственном университете имени П.Г. Демидова (ЯрГУ), на тему «Разработка алгоритмов анализа аудитории для систем прикладного телевидения».

Результаты диссертационной работы Матвеева Д.В. «Разработка алгоритмов анализа аудитории для систем прикладного телевидения» нашли применение в работе ООО «БАЙТ-ПРО». Особый практический интерес представляют следующие результаты диссертации:

- 1. Проведенный анализ работы алгоритмов детектирования лиц на видеоизображениях и предложенный новый алгоритм с использованием ансамбля решающих деревьев, позволяющий улучшить характеристики системы детектирования на 13% по сравнению с классическим алгоритмом из библиотеки OpenCV.
- 2. Реализованные на языках высокого уровня и протестированные в различных условиях робастные алгоритмы детектирования объектов для диагонального и вертикального способов подвеса видеокамер, позволяющие повысить точность обнаружения объектов по сравнению с известными подходами.

Разработанные Матвеевым Д.В. алгоритмы анализа аудитории на основе алгоритмов компьютерного зрения применяются в составе программно-аппаратного комплекса для бизнес-аналитики по видеоизображениям.

Руководитель отдела перспективных разработок

Воробьев А.В.

Директор ООО «Байт-ПРО»

Додонов А.Е.