



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н. Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ

НА ТЕМУ:

*«Метод распознавания спортивных действий человека
на видео с использованием локального дескриптора»*

Студент ИУ7-82Б
(Группа)

(Подпись, дата)

Сысоева В. Р.
(И. О. Фамилия)

Руководитель ВКР

(Подпись, дата)

Мальцева Д. Ю.
(И. О. Фамилия)

Нормоконтролер

(Подпись, дата)

Мальцева Д. Ю.
(И. О. Фамилия)

2023 г.

РЕФЕРАТ

Расчетно-пояснительная записка 69 с., 20 рис., 1 табл., 34 источн., 1 прил.

В данной работе представлена разработка метода распознавания спортивных действий человека на видео с использованием локального дескриптора.

Рассмотрены методы детектирования объектов на изображениях с помощью дескриптора и алгоритмы классификации, проведена классификация существующих методов распознавания действий человека.

Представленный метод реализуется в виде программного обеспечения и проводится оценка его эффективности и характеристик.

Ключевые слова: распознавание действий, классификатор, HOG дескриптор, случайный лес, машинное обучение, метрики.

СОДЕРЖАНИЕ

РЕФЕРАТ	5
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	8
ВВЕДЕНИЕ	9
1 Аналитический раздел	11
1.1 Обзор областей применения методов распознавания физической активности человека	11
1.1.1 Безопасность	11
1.1.2 Медицина	12
1.1.3 Производство и образование	12
1.2 Алгоритмы классификации	13
1.2.1 Метод Байеса	13
1.2.2 Метод опорных векторов	14
1.2.3 Метод k-ближайших соседей	15
1.2.4 Деревья решений	17
1.2.5 Метод случайного леса	18
1.3 Обзор алгоритмов детектирования объектов с помощью дескрипторов	20
1.3.1 Алгоритм SIFT (масштабно-инвариантное преобразование признаков)	21
1.3.2 Алгоритм SURF (ускоренная надежная функция)	22
1.3.3 Метод RIFF (Rotation Invariant Fast Features)	23
1.3.4 HOG (Гистограмма направленных градиентов)	24
1.4 Обзор методов распознавания действий человека	26
1.4.1 Пространственно-временной метод	27
1.4.2 Стохастический метод	29
1.4.3 Методы, основанные на правилах	31
1.4.4 Методы, основанные на форме	32
1.5 Сравнение методов	33
1.6 Выбор данных для обучения модели	35
1.6.1 Требования	35

1.6.2	Структура	35
2	Конструкторский раздел	37
2.1	Проектирование метода распознавания	37
2.1.1	IDEF0	37
2.1.2	Предобработка видеоролика	39
2.1.3	Формирование НОГ дескриптора видео	41
2.1.4	Обучение модели	43
2.2	Набор данных	46
3	Технологический раздел	48
3.1	Входные и выходные данные	48
3.2	Выбор языка программирования	48
3.3	Разработка ПО	49
3.3.1	Модуль предобработки данных	50
3.3.2	Модуль формирования дескрипторов	51
3.3.3	Модуль обучения модели	52
3.4	Тестирование модели	54
3.5	Пользовательский интерфейс	55
4	Исследовательский раздел	59
4.1	Исследование метрик модели в зависимости от разметки данных	59
4.2	Исследование результатов распознавания модели в зависимости от контрастности видео	61
4.3	Исследование результатов распознавания модели в зависимости от качества видео	62
4.4	Выводы	63
	ЗАКЛЮЧЕНИЕ	64
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	68
	ПРИЛОЖЕНИЕ А	69

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей расчетно-пояснительной записке применяют следующие сокращения и обозначения.

- HOG (от англ. Histogram of oriented gradients) — дескрипторы особых точек, которые используются в компьютерном зрении и обработке изображений с целью распознавания объектов..
- SVM (от англ. Support vector machine) — метод машинного обучения, применяемый в задачах классификации, основанный на построении гиперплоскости и ее анализе.
- CART (от англ. Classification and regression trees) — это аббревиатура, обозначающая методы классификации и регрессии с использованием дерева решений.
- SIFT (англ. scale-invariant feature transform) — алгоритмом выявления и описания локальных признаков в изображениях.

ВВЕДЕНИЕ

Распознавание видов физической активности человека является одним из актуальных направлений исследования в области машинного обучения, так как результаты распознавания необходимы при решении многих практических задач.

Студенты МГТУ им Н. Э. Баумана на первом курсе университета сталкиваются с дневниками самоподготовки на парах по физической культуре. Начиная с 2020 года эта система стала автоматизированной и студенты загружают видео со своими занятиями на сайт кафедры ФВ (физического воспитания), а преподаватели отсматривают эти видео. Если бы существовала система способная распознавать вид физических упражнений, выполняемых студентом, это бы ускорило процесс проверки дневников самоподготовки.

Также после некоторых операций пациентам необходимо во время реабилитации выполнять физические упражнения для более быстрого восстановления. В статье [1] производится выбор признаков, необходимых для классификации физической активности в телереабилитационной системе и рамках этой системы разработан модуль классификации видов физической активности на основе показаний датчиков смартфона: акселерометра и гироскопа. Телереабилитация направлена на постоянный мониторинг физической активности человека.

Цель работы – разработать метод распознавания спортивных действий человека на видео с использованием локального дескриптора.

Для достижения поставленной цели необходимо решить следующие задачи.

- Рассмотреть алгоритмы, с помощью которых можно реализовать метод распознавания спортивных действий человека.
- Провести обзор существующих решений распознавания действий человека.
- Разработать метод распознавания спортивных действий человека на видео с использованием локального дескриптора.
- Спроектировать и реализовать ПО, демонстрирующее работу метода.

- Исследовать метрики разработанного метода в зависимости от разметки данных.
- Исследовать разработанный метод на применимость при работе с видео различной контрастности и качества.

1 Аналитический раздел

В данном разделе рассматриваются сферы применения методов распознавания действий человека, описываются алгоритмы детектирования объектов с помощью дескрипторов, алгоритмы классификаторов, их достоинства и недостатки. Проводится классификация уже существующих методов распознавания действий человека, представлены результаты сравнения рассмотренных методов. Описываются требования к набору данных для обучения модели и его структура.

1.1 Обзор областей применения методов распознавания физической активности человека

Распознавание видов физической активности человека является одной из актуальных задач в области машинного обучения из-за сложности и разнообразия видов физической активности, выполняемых человеком.

Целью распознавания является определение деятельности человека на основе данных датчиков для последующего анализа системой с учетом практической задачи.

Сложный и изменчивый характер данных об активности создает многочисленные проблемы, которые влияют на производительность систем, используемых для решения практических задач. Ниже рассмотрим некоторые области применения технологии распознавания физической активности человека.

1.1.1 Безопасность

Системы видеонаблюдения являются одним из основных средств обеспечения безопасности на объекте информатизации. Видео с дорожных и уличных камер могут содержать как моменты рядовых событий, так и моменты правонарушений, которые требуют своевременной фиксации и передачи в правоохранительные органы. Автоматическая идентификация, при помощи камер наружного наблюдения, преступников и террористов по характерным жестам, позволит существенно упростить работу правоохранительных органов и, потенциально, спасти множество жизней.

Задача автоматизации процесса анализа видеопотока с целью выявления инцидентов, возникающих на контролируемом объекте. В настоящее время уже существует множество разработок, направленных на предварительный анализ происходящего на видеоизображении: забытые предметы, оружие, проход людей в запретную зону (например, выход на железнодорожные пути), драки и т.п.

В статье [2] предложено применение подхода переноса знаний для распознавания агрессии на видеоизображении с использованием трех моделей трехмерных сверточных искусственных нейронных сетей (ИНС) в качестве экстрактора признаков.

1.1.2 Медицина

В области медицины приложения для распознавания физической активности также играют важную роль, так как позволяют дистанционно контролировать корректность выполнения рекомендаций или плана, назначенного врачом, тем самым сокращая время пребывания пациента в больнице.

После перенесенной болезни пациенту необходимо пройти реабилитацию. Физическая телереабилитация подразумевает дистанционный мониторинг выполнения пациентом физических упражнений с целью восстановления здоровья, физического состояния и его трудоспособности.

1.1.3 Производство и образование

Факт нахождения человека на рабочем месте не подразумевает того, что работник эффективно выполняет свои обязанности. Распознавание видов активности в сфере производства и услуг позволит контролировать эффективности работы сотрудников, выявлять нелояльных сотрудников, некачественное выполнение работы и мошеннические схемы. На основе полученных данных руководители организаций и предприятий смогут своевременно принимать меры по оптимизации работы персонала и повысить эффективность производственных процессов.

Высокие технологии глубоко проникли в сферу образования. Системы видеонаблюдения существуют во всех учебных заведениях.

В статье [3] описан метод распознавания действий учащихся на видеопоследовательности с экзамена.

Аномальными ситуациями в данном случае понимаются нарушения правил проведения экзамена: списывание, использование запрещенных предметов, нарушения со стороны организаторов экзамена.

1.2 Алгоритмы классификации

Для решения задачи распознавания действий необходимо рассмотреть алгоритмы классификации. Алгоритмы решают задачу распределение объектов по группам (классам) на основании каких-либо признаков. Класс – это множество объектов, имеющих определенный общий признак, отличающий эту совокупность от других объектов. Рассмотрим следующие методы классификации: метод Байеса, деревья решений, метод k -ближайших соседей, метод опорных векторов, случайный лес.

1.2.1 Метод Байеса

Метод относится к вероятностным методам классификации. Для каждого наблюдения их множества X высчитывается вероятность его возникновения в соответствующем классе по формуле:

$$P(x_i|y_i) = \frac{\text{count}(x_i|y_i)}{\text{count}(X|y_i)}, \quad (1.1)$$

где $\text{count}(x_i|y_i)$ – количество элементов x_i в категории y_i . $\text{count}(X|y_i)$ – количество наблюдений из множества X принадлежащих категории y_i . Для того что бы учитывать категории с небольшим количеством наблюдений используется нормализация. Тогда вероятность вычисляется по формуле:

$$P(x_i|y_i) = \frac{\frac{\text{count}(x_i|y_i)}{\text{count}(X|y_i)}}{\sum_j^n \frac{\text{count}(x_i|y_n)}{\text{count}(X|y_n)}}, \quad (1.2)$$

При вычислении вероятности по формуле (1.3) требуется так же априорная вероятность возникновения категории y_i равна отношению числа наблюдений в y_i к общему числу наблюдений.

$$P(x_i|y_i) = \frac{\text{count}(X|(y_i))}{\text{count}(X)}, \quad (1.3)$$

где $\text{count}(X|(y_i))$ – число наблюдений принадлежащих категории y_i ,

$count(X)$ – общее число наблюдений.

Классификация набора наблюдений происходит после вычисления вероятности по формуле:

$$P(X|y_i) = P(y_i) \prod_j^n p(x_j|y_i), \quad (1.4)$$

где $P(y_i)$ – априорная вероятность возникновения y_i , $p(x_j|y_i)$ – вероятность возникновения наблюдения x_j в категории y_i [4].

В статье [5] перечислены достоинства и недостатки метода Байеса.

Преимущества метода:

- высокая скорость работы;
- относительно простая программная реализация алгоритма;
- легкая интерпретируемость результатов работы алгоритма.

Недостатки метода: относительно низкое качество классификации и неспособность учитывать зависимость результата классификации от сочетания признаков.

1.2.2 Метод опорных векторов

Метод опорных векторов является широко распространенным методом, который используется в методах оптимизации, а также и в машинном обучении.

Суть задачи классификации заключается в построении алгоритма классификации по обучаемой выборке. Идея линейных классификаторов заключается в построении линейных функций, являющимися разделяющими поверхностями, по обе стороны которой лежат объекты разных классов. При использовании метода опорных векторов используется функционал особого вида:

$$\sum_{i=1}^l (1 - M_i(w, w_0)) + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}, \quad (1.5)$$

где $M_i(w, w_0) = y_i((w, x_i) - w_0)$ – величина отступа.

Пусть имеем объекты, принадлежащие двум классам. Если выборка является линейно разделимой, то оптимальную разделяющую поверхность можно найти из решения системы:

$$\begin{cases} \|w\|^2 \rightarrow \min, \\ M_i(w, w_0) \geq 1, i = 1, \dots, l. \end{cases} \quad (1.6)$$

С учетом того, что выборка на практике чаще всего не является линейно разделимой, ограничение в системе становится менее жестким:

$$\begin{cases} \frac{1}{2}\|w\|^2 + C \sum_{i=1}^l \xi_i \rightarrow \min_{w, w_0, \xi}, \\ M_i(w, w_0) \geq 1 - \xi_i, i = 1, \dots, l \quad \xi_i \geq 0, i = 1, \dots, l. \end{cases} \quad (1.7)$$

При решении системы в зависимости от параметров получаются 3 типа объектов [6]:

- периферийные, неинформативные объекты;
- опорные;
- объекты, которые оказались по другую сторону от разделяющей поверхности.

В статье [5] перечислены достоинства и недостатки метода опорных векторов.

Достоинства: за счет применения гиперплоскостей работает при малых объемах обучающей выборки. За счет использования ядра описывающего связь между элементами выборки, можно использовать гиперплоскости разной сложности.

Недостатки: большое время обучение, необходимость ядра для конкретного случая.

1.2.3 Метод k-ближайших соседей

Метод ближайших соседей – один из самых простых классификаторов, основанный на оценивании сходства признаков объектов. Классифицируемый объект относится к тому классу, к которому принадлежат ближайшие к нему

объекты обучающей выборки. Первоначально задаем обучающую выборку пар «объект-класс»:

$$X^m = (x_1, y_1), \dots, (x_m, y_m). \quad (1.8)$$

Для произвольного объекта u расположим объекты обучающей выборки x_i в порядке возрастания расстояния до u :

$$p(u, x_{1;u}) \leq p(u, x_{2;u}) \leq \dots \leq p(u, x_{m;u}), \quad (1.9)$$

где через $x_{1;u}$ обозначается тот объект обучающей выборки, который является i соседом объекта u . Так же перенумеруем классы, к которым относятся $x_{1;u}$ — обозначим их $y_{i;u}$. Алгоритм ближайших соседей в общем виде можно выразить формулой:

$$a(u) = \operatorname{argmax} \sum_{i=1}^m [y(x_{1;u}) = y] w(i, u). \quad (1.10)$$

Здесь w весовая функция, показывающая, важен ли для объекта элемент под номером i . Для метода k ближайших соседей $w(i, u) = [i \leq k]$. Таким образом, алгоритм классификации методом ближайших k соседей очень прост. У нас есть выборка из данных — обучающее множество, для каждого элемента которого известно, к какому классу он принадлежит. Для каждого следующего элемента неизвестного класса берется выборка из k ближайших элементов, класс которых уже известен. Новому элементу присваивается тот класс, элементов которого оказалось больше всего в выборке [7].

В статье [5] перечислены достоинства и недостатки метода k -ближайших соседей.

Преимущества метода:

- возможность обновления обучающей выборки без переобучения классификатора;
- устойчивость алгоритма к аномальным выбросам в исходных данных;
- хорошее обучение в случае с линейно неразделимыми выборками.

Недостатки метода:

- репрезентативность набора данных, используемого для алгоритма;
- высокая зависимость результатов классификации от выбранной метрики;
- большая длительность работы из-за необходимости полного перебора обучающей выборки;
- невозможность решения задач большой размерности по количеству классов и документов.

1.2.4 Деревья решений

Классификация является одной из задач, где используются деревья принятия решений. Они создают иерархическую структуру классифицирующих правил типа «если...то», имеющую вид дерева. Для принятия решения, к какому классу отнести некоторый объект или ситуацию, требуется ответить на вопросы, стоящий в узлах этого дерева, начиная с его корня. Основа такой структуры – ответы «Да» или «Нет» на ряд вопросов.

Рассмотрим два наиболее известных алгоритма построения деревьев решений CART и C4.5 для задачи классификации. Основным отличием данных алгоритмов является устойчивость к шумам и выбросам данных. Алгоритм CART решает задачи классификации и является самым распространенным способом выявления, структурирование и графического представления логических закономерностей в данных. Его преимущество заключается в следующем [8]:

- быстрый процесс обнаружения знаний;
- генерация правил в предметных областях, в которых трудно формализуются знания;
- извлечение правил на естественном языке;
- создание интуитивно понятной классификационной модели предметной области;
- прогноз с высокой точностью, сопоставимой с другими методами;
- построение непараметрических моделей.

В данном алгоритме можно выделить три основные операции: сортировка источника данных при формировании множества условий для атрибутов числового типа, вычисление критерия Gini при разбиении узлов бинарного дерева, перемещение в таблице значительных объемов информации при делении узла.

Отбор наилучшего варианта разбиения узла дерева проводится наибольшей классифицирующей силой, вычисляемой по критерию Gini:

$$GINI = \frac{1}{|L|} \sum_{i=1}^{N_{cp}} (l_i)^2 + \frac{1}{|R|} \sum_{i=1}^{N_{cp}} (r_i)^2. \quad (1.11)$$

Алгоритм C4.5 строит дерево решений с неограниченным количеством ветвей у узла. Данный алгоритм может работать только с дискретным зависимым атрибутом и поэтому может решать только задачи классификации. Его преимущества заключаются в следующем [8]:

- C4.5 использует относительную энтропию при генерировании деревьев решений;
- C4.5 использует однопроходное отсечение ветвей, чтобы упростить переобучение. Это существенно улучшает работу алгоритма;
- C4.5 может работать и с непрерывными и с дискретными данными.

В статье [5] перечислены достоинства и недостатки метода дерева решений.

Преимущества метода:

- высокая скорость работы;
- относительно простая программная реализация алгоритма.

Недостатки метода: неустойчивость алгоритма по отношению к выбросам в исходных данных и большой объем данных для получения точных результатов.

1.2.5 Метод случайного леса

Метод случайного леса основан на построение так называемого ансамбля деревьев решений. Алгоритм получился сложением двух основных идей: метод

бэггинга и метод случайных подпространств. Данный метод используется для решения задач классификации, регрессии и кластеризации.

Деревом решений обычно называют представление правил в иерархической, последовательной структуре, где каждому объекту соответствует узел, дающий решение. Под правилом понимается логическая структура «если... то».

Удобство применения построения дерева решений состоит в удобном представлении знаний в экспертных системах, а именно после написания совокупности правил их легко можно изобразить в виде графа, это и удобно, и более наглядно, чем перечисление правил. Пусть имеется некоторая выборка из m на n элементов.

Можно произвольно взять некоторые данные из этой выборки и на ее основе построить дерево решений, пример построения дерева приведен на рисунке 1.1. Таким же образом проделать и с другими данными. Вся особенность этого алгоритма состоит в универсальности, скорости и случайности построения деревьев решений. Дерево строится без отсечения ветвей (как это можно увидеть при построении деревьев алгоритмов типа CART или C4.5, т.е. происходит построение полных деревьев).

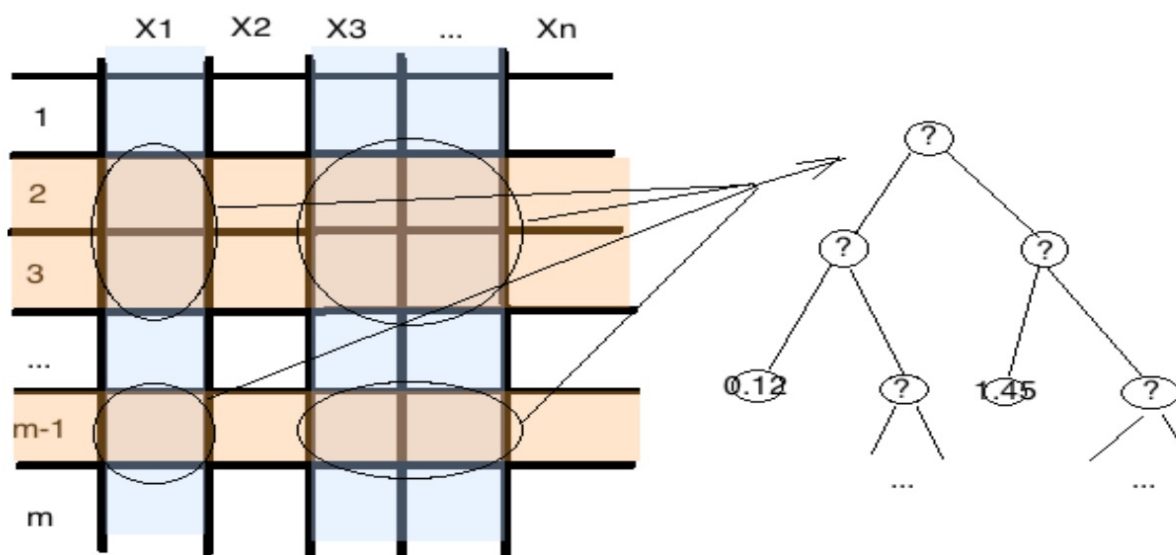


Рисунок 1.1 – Выбор случайных данных и построение на их основе дерева решений

Далее результат будет либо усредняться (в случае регрессии) или путем голосования (в случае классификации). На каждом шаге построения дерева

решения нужно выбирать признак и значение порога, который будет соответствовать наилучшему результату по некоторому заданному критерию. Для решения прикладных задач используют критерий Джини.

Существуют ряд преимуществ данного метода [9]:

- имеется встроенная проверка качества;
- повышение точности;
- простота применения, т.к. основными параметрами является количество деревьев в ансамбле и количество признаков, по которым эти деревья будут расщепляться;
- обработка данных с большим количеством признаков и классов;
- высокая параллелизуемость.

Помимо отмеченных достоинств также существует и недостатки применения:

- высокая алгоритмическая сложность;
- большое количество получившихся деревьев.

Выводы

Для реализации метода распознавания спортивных действий на видео был выбран алгоритм случайного леса, поскольку он хорошо справляется с большими наборами данных, алгоритм легко распараллелить при программной реализации, устойчив к выбросам, после обучения модели можно определить какая из функций была важнее, а также алгоритм применим в сочетании с методом градиентного бустинга.

1.3 Обзор алгоритмов детектирования объектов с помощью дескрипторов

При классификации изображений их упрощение происходит за счет извлечения важной информации. Оригинальное изображение содержит слишком много дополнительной информации, которая не требуется для классификации. Этот шаг называется извлечением объекта. Существуют специальные

алгоритмы, решающие эту задачу. Существует довольно большое количество методов, используемых в компьютерном зрении для решения этой задачи, среди них: HOG (гистограмма направленных градиентов), SIFT (масштабно-инвариантное преобразование признаков), SURF (ускоренная надежная функция) и RIFF.

Для рассмотрения перечисленных алгоритмов также необходимо упомянуть понятие дескриптор. Дескриптором (вектором признаков) называется набор численных параметров, описывающих характеристики объекта (или его части), например цвет, форму и т.д. Векторы признаков принимают значения в пространстве признаков.

Дескрипторы разделяют на глобальные, описывающие объект целиком, и локальные, описывающие значимые части объекта или изображения. Основной областью применения локальных дескрипторов является анализ изображений при моделировании компьютерного зрения [10].

Ниже рассмотрим упомянутые алгоритмы.

1.3.1 Алгоритм SIFT (масштабно-инвариантное преобразование признаков)

SIFT является одним из наиболее часто используемых алгоритмов описания дескрипторов особых точек на изображении. Дескрипторы, полученные с помощью этого алгоритма инвариантны к масштабированию и поворотам изображения, устойчивы к изменениям освещения, шумам и изменениям позиции наблюдателя [11].

Характерной особенностью алгоритма SIFT является применяемый в них метод определения позиций структурных элементов: ключевые точки (центры локальных регионов, описываемых SIFT-дескрипторами) выбираются там, где результаты фильтрации изображения с использованием детекторов Харриса или Гессе имеют экстремумы.

Анализ изображения производится последовательно для разных коэффициентов его масштабирования, а содержание локального региона, описываемого SIFT-дескриптором, нормализуется относительно вращения. Такой способ описания структурных элементов делает их инвариантными к смещениям, масштабированию и вращению в плоскости изображения [12].

В статье [13] выделяются следующие этапы метода SIFT:

- определение локальных особенностей (ключевых точек);
- локализация особенностей;
- вычисление ориентаций особенностей;
- описание локальных особенностей через дескриптор;
- сопоставление дескрипторов.

1.3.2 Алгоритм SURF (ускоренная надежная функция)

Метод SURF ищет ключевые точки и строит описание найденных ключевых точек через дескрипторы особенностей и является аналогом метода SIFT. Ключевой точкой является локальный экстремум детерминанта матрицы Гессе. Для двумерного случая детерминант матрицы Гессе определяется следующим образом:

$$\det(H) = \frac{\partial^2 f}{\partial x^2} \frac{\partial^2 f}{\partial y^2} - \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2, \quad (1.12)$$

где

матрица Гессе:

$$H(f(x, y)) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix}, \quad (1.13)$$

$f(x, y)$ – функция изменения градиента яркости.

Гессиан инвариантен к сдвигу яркости изображения и повороту, но не инвариантен к масштабу. Данную проблему решают с помощью последовательного перебора различных масштабов и фильтров и поочередного применения их к одному пикселю. В качестве опорных точек выбирают локальные максимумы гессианов, соответствующие локальным максимумам изменения градиента яркости.

После нахождения точек локальных максимумов определяют точка истинного максимума гессиана. Данная методика гарантирует, что в окрестности ключевой точки расположены участки с разными градиентами. Дескриптор

представляет собой массив чисел, определяющих опорную точку. Инвариантность дескриптора относительно поворота обеспечивается дисперсией (различием) дескрипторов для разных особых точек.

Для определения особых точек в методе SURF используется целочисленная аппроксимация детерминанта blob-детектора гессиана, который вычисляется с помощью трех операций с использованием предварительно вычисленного интегрального изображения. Blob-детектирование – выявление областей в цифровом изображении, которые отличаются по своим свойствам, таким как яркость или цвет, от прилегающих областей [13].

Также в статье [13] выделяются следующие этапы алгоритма работы SURF:

- масштабно-пространственное представление;
- расчет значений гессиана;
- поиск точек локальных максимумов;
- определение точки истинного максимума;
- определение ориентации опорной точки;
- формирование дескриптора опорной точки.

1.3.3 Метод RIFF (Rotation Invariant Fast Features)

В основу метода RIFF положено радиальное и тангенциальное разложение гистограмм градиента и последующая обработка по кольцам. Дескриптор также инвариантен к масштабированию, вращению и изменению освещенности.

Дескрипторы RIFF формируются достаточно быстро для обеспечения отслеживания объектов со скоростью следования кадров или близко к скорости следования кадров и достаточно устойчиво для решения задач крупномасштабного распознавания. Формирование дескриптора RIFF может начинаться с процедуры формирования дескриптора сжатой гистограммы градиентов (CHoG, compressed histogram of gradient) [11].

1.3.4 HOG (Гистограмма направленных градиентов)

Описание объектов составляется из признаков, выбор признаков определяется результативность работа системы. Одним из видов признаков являются признаки формы. Чаще всего при детектирование человека и его движения используются признаки формы и текстуры. Признаки формы описывают расположение и направление переходов на изображении, оценивают градиент. Более простые признаки стремятся найти простые геометрические фигуры на изображении и определить, являются ли наборы этих фигур человеком.

Следующим методом является метод под названием гистограмма направленных градиентов. Данный метод был предложен в далеком 2005 г., это был один из самых первых алгоритмов, который мог довольно быстро и качественно решать задачу по распознаванию образов на изображениях. Главной идеей данного алгоритма было то, что любое изображение можно было описать распределением градиентов интенсивности и направления краев [14].

Алгоритм преобразовывает изображение формата $w \cdot h \cdot 3$ в вектор значений, характеризующихся величинами, получаемых в процессе вычисления вертикальных и горизонтальных градиентов. Это достигается путем фильтрации изображения с помощью представленных ядер, показанных на рисунке 1.2.

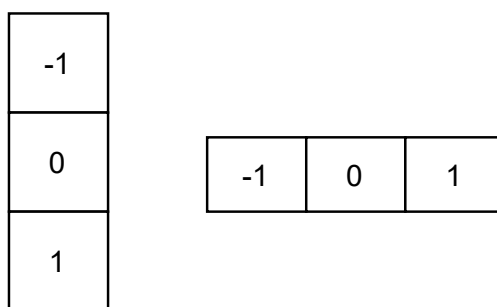


Рисунок 1.2 – Представление ядер

Величина и направление градиентов:

$$M(magnitude) = \sqrt{G_x^2 + G_y^2}, \quad (1.14)$$

$$D(direction) = \arctan \frac{G_x}{G_y}, \quad (1.15)$$

где

$$G_x = \lim_{\Delta x \rightarrow 1} \frac{f(x_0 + \Delta x) - f(x_0)}{\Delta x} = f(x_0 + 1) - f(x_0), \quad (1.16)$$

$$G_y = \lim_{\Delta y \rightarrow 1} \frac{f(y_0 + \Delta y) - f(y_0)}{\Delta y} = f(y_0 + 1) - f(y_0). \quad (1.17)$$

Величина градиента показывает, насколько контрастен переход от цвета к цвету в определенной области. При анализе каждого пикселя изображения на выходе значение величины – это максимальная среди всех возможных разностей между значениями цвета [14].

Обычно их построение происходило путем разбиения изображения на множество ячеек, каждой из которых присваивались гистограммы направлений градиентов для пикселей внутри ячейки. Среднестатистический размер такой ячейки – 8×8 пикселей. Каждая такая ячейка преобразовывается к вектору 9×1 , в каждой из компонент которого лежит некоторое значение, показывающее порядок величины в определенном направлении, то есть каждый элемент вектора соответствует градусу – $0, 20, \dots, 160$ (рассматриваются беззнаковые градиенты), представлена на рисунке 1.3 [15].

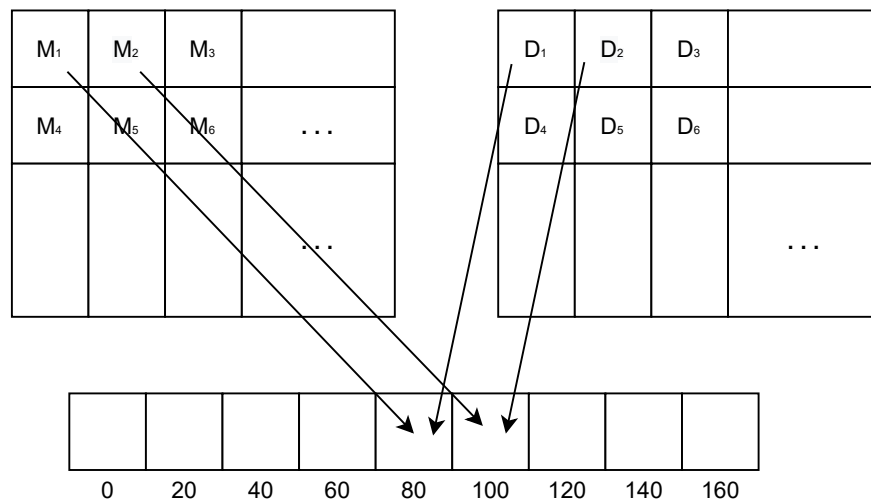


Рисунок 1.3 – Гистограмма преобразования

Для того чтобы получить наилучший и более точный результат, изображение, которое бралось для обработки, приводили к черно-белому виду, а локальные гистограммы нормализовали по контрасту относительно интенсивности. Нормализация по контрасту является очень важным шагом, так как с

помощью него можно добиться наибольшей инвариантности к освещению.

Таким образом, перебираются все возможные ячейки и для изображения получаем на выходе вектор $9N$, где N – количество таких ячеек. Смещение по индексам всегда составляет единицу и в заключении проводится классификация с помощью системы, которая построена на методе обучения с учителем, а именно, методе опорных векторов [14].

Выводы

Для реализации метода распознавания спортивных действий на видео был выбран алгоритм HOG, поскольку он работает локально, метод поддерживает инвариантность геометрических и фотометрических преобразований, за исключением ориентации объекта. Засчет грубого разбиение пространства, точного вычисление направлений и сильной локальной фотометрической нормализации получается обнаружить людей, даже при их движении. Дескриптор HOG, таким образом, является хорошим средством нахождения людей на изображениях. Также алгоритмы SIFT и SURF не позволяет точно выделить элементы для объектов с гладкими краями, а тело человека представляет собой сглаженный контур.

1.4 Обзор методов распознавания действий человека

Был проведен анализ предметной области, вследствие чего были рассмотрены области применения методов распознавания действий человека, виды человеческих действий, учитывающихся при распознавании, алгоритмы детектирования объектов на изображении.

В этом разделе будет проведен обзор существующих методов распознавания физической активности человека, выделены критерии их сравнения и представлена таблица с классификацией.

Область исследований распознавания физической активности человека не стоит на месте. На данный момент опубликовано множество статей с описанием различных методов распознавания физической активности человека, использующие разные технологии, подходы, алгоритмы машинного обучения. В некоторых работах авторы предлагают новые методы, некоторые основываются на уже опубликованных расширяя или оптимизируя их. Ниже будут проведен обзор работ по данной теме.

1.4.1 Пространственно-временной метод

Особенность пространственно-временных методов заключается в том, что они учитывают пространственно-временные корреляции между локальными объектами и принимают во внимание потенциально ценную информацию о глобальном пространственно-временном распределении точек интереса.

Пространственно-временные модели имеют четыре основных компонента: детектор пространственновременной точки (STIP), дескриптор функции, конструктор словарей и классификатор [16].

Пространственно-временное обнаружение действий и событий в видео является сложной задачей. Помимо трудностей, связанных с распознаванием, основная проблема заключается в том, что в каждом кадре видео необходимо оценить ограничивающую рамку интересующего действия, которые вместе образуют пространственно-временную трубу, которая определяет местоположение действия в пространстве и времени.

В статье [17] авторы предлагают улучшить уже существующий подход в методах распознавания 2D-объектов. Эти методы создают зависящие от изображения, но неконтролируемые и независимые от класса предварительные рамки, ограничивающие объект, которые затем оцениваются детектором.

Идея представляет новый иерархический метод супервокселей, который начинается с иерархической кластеризации извлеченных суперпикселей для каждого кадра. Авторы экспериментально оценивают предложения по обнаружению в сочетании с новым методом супервокселя, а также с существующими. Эта оценка показывает, что супервоксели приводят к более точным предложениям по сравнению с использованием существующих современных методов супервокселирования.

Еще один метод распознавания действий человека, использующий эффективный подход к локализации действий путем изучения контекстуальных отношений в виде относительных местоположений между различными регионами видео описан в статье [18]. Локализация начинается с чрезмерной сегментации видео на супервоксели, которые позволяют сохранить границы действия, а также снизить сложность проблемы. Контекстные отношения формируются во время обучения, в процессе которого происходит фиксирование перемещения от всех супервокселей в видео к тем, которые принадлежат действиям переднего плана. Затем выбирается супервоксель случайным образом и

используется контекстная информация, полученная во время тренировки для оценки вероятности принадлежности каждого супервокселя действию переднего плана. Происходит переход к новому супервокселю, и процесс повторяется в течение нескольких шагов. Этот «обход контекста» генерирует условное распределение действия по всем супервокселям. Затем условное случайное поле используется для поиска предложений действий в видео, достоверность которых получена с помощью метода опорных векторов.

В работе также происходит проверка предложенного метода на нескольких наборах данных (UCF-Sports, Sub-JHMDB, THUMOS13) и результаты показали, что контекст в виде относительных перемещений между супервокселями может быть чрезвычайно полезен для локализации действий. Это также приводит к значительно меньшему количеству оценок классификатора, что резко контрастирует с альтернативными подходами, использующими скользящее окно.

В статье [19] рассматривается проблема локализации неконтролируемых действий в видеороликах. Предлагается новый подход, учитывая немаркированные данные без аннотаций в виде ограничивающих рамок, который: обнаруживает метки классов действий и пространственно-временную локализацию действий в видео.

Метод начинается с вычисления локальных характеристик видео для применения спектральной кластеризации к набору немаркированных обучающих видеороликов. Для каждого кластера видео строится неориентированный граф для извлечения доминирующего набора, который известен высокой внутренней однородностью и неоднородностью между вершины за его пределами. Затем применяется подход дискриминационной кластеризации путем обучения классификатора для каждого кластера, чтобы итеративно выбирать видео из недоминирующего набора и получать полные классы действий с видео. Как только классы обнаружены, обучающие видеоролики в каждом кластере выбираются для выполнения автоматических пространственно-временных аннотаций, сначала чрезмерно сегментируя видеоролики в каждом обнаруженном классе в супервоксели и строя ориентированный граф, чтобы применить вариант задачи о рюкзаке с временными ограничениями. Оптимизация рюкзака совместно собирает подмножество супервокселей, заставляя аннотированное действие быть пространственно-временным, а его объем —

размером с субъекта. Во время тестирования действия локализуются с использованием аналогичного подхода к рюкзаку, где супервоксели группируются вместе, а метод опорных векторов, изученный с использованием видеороликов из обнаруженных классов действий, используется для распознавания этих действий.

Метод тестировался на UCF-Sports, Sub-JHMDB, JHMDB, THUMOS13 и Наборы данных UCF101. Эксперименты показывают, что, несмотря на неиспользование ярлыков классов действий и аннотации к ограничивающим рамкам, получены результаты, сопоставимые с самыми современными контролируемыми методами

1.4.2 Стохастический метод

Особое внимание уделялось видам деятельности, в которых объект, подлежащий детектированию, может рассматриваться как стохастически предсказуемая последовательность состояний. Исследователи разработали и использовали множество стохастических методов, таких как скрытая марковская модель и скрытые условные случайные поля (HCRF), чтобы вывести полезные результаты для распознавания человеческой деятельности.

Несмотря на значительные достижения в области распознавания действий человека, существующие методы проявляют недостатки при работе в реальном пространстве, характеризующемся моделью фона, которая представляет собой текстуру. Важной проблемой, связанной с разработкой интеллектуальных систем видеонаблюдения, является улучшение методов распознавания действий человека на сложноструктурированных изображениях и фоне в виде стохастических текстур. Текстура играет важную роль при анализе визуального содержания изображения. Ряд задач компьютерного зрения, в том числе и распознавание действий используют информацию о текстуре.

Так в статье [20] авторы предлагают новый метод, основанный на математическом аппарате стохастической геометрии, используя трехмерные дискретные преобразования Фурье, что позволяет сформировать большое число новых, конструктивных признаков и максимально полно охарактеризовать подобные изображения и повысить эффективность распознавания действий человека на сложноструктурированных изображениях и фоне в виде стохастических текстур.

В статье [21] предлагается метод распознавания повседневной жизнедеятельности человека. Для улучшения распознавания и классификации физической активности человека (например, ходьба, питье и бег), представлена модель, которая объединяет методы предварительной обработки данных (такие как шумоподавление) наряду с основными характеристиками предметной области (такими как время, частота, частотно-временные характеристики). После этого используется стохастический градиентный спуск (SGD) для улучшения производительности извлеченных объектов. Выбранные функции обрабатываются классификатором случайных лесов для обнаружения и мониторинга физической активности человека.

В работе предлагаемая система была оценена на пяти контрольных наборах данных, а именно IM-WSHA, PAMAP-2, UCI HAR, MobiAct и Базы данных MOTIONSENSE. Результаты эксперимента показали, что система превзошла современные методы распознавания по скорости 90,18%, 91,25%, 91,83%, 90,46%, и 92,16% из наборов данных IM-WSHA, PAMAP-2, UCI HAR, MobiAct и MOTIONSENSE соответственно. Предлагаемая модель имеет потенциальные применения в здравоохранении, играх, умных домах, безопасности и видеонаблюдении.

В статье [3] представлен подход к распознаванию аномальных действий людей в виде последовательностей промежуточных состояний. Предлагается разложить каждое действие в последовательность дискретных промежуточных состояний и представить переходы между состояниями как стохастический процесс. Каждое состояние описывается положением суставов человека. Действия описываются с помощью скрытой Марковской модели, основанной на найденных состояниях и переходах между ними. Полученная в результате модель является комбинацией стохастической модели действий человека и скелетной модели, описывающей промежуточные состояния. Для распознавания промежуточных состояний применяется сверточная нейронная сеть. Для поиска параметров Марковской модели используется алгоритм Витерби. Предложенный метод был реализован и протестирован на двух выборках: MPII Human Pose Dataset и на видеозаписях проведения экзаменов. Точность распознавания составила 75%.

1.4.3 Методы, основанные на правилах

Подходы, основанные на правилах, определяют текущие события путем моделирования деятельности с использованием правил или наборов атрибутов, описывающих событие.

Каждая деятельность рассматривается как набор примитивных правил или атрибутов, что позволяет построить описательную модель для распознавания человеческой деятельности.

В статье [22] было предложено распознавание действий сложных сцен с несколькими объектами. Каждый субъект должен следовать набору определенных правил при выполнении действия. Процесс распознавания осуществлялся по видеозаписям баскетбольных матчей, где игроки были впервые обнаружены и отслежены, генерируя набор траекторий, которые используются для создания набора пространственно-временных событий. Основываясь на логике первого порядка и вероятностных подходах, таких как сети Маркова, авторы смогли определить, какое событие произошло.

В статье [23] предлагается метод, который создает иерархическую структуру для представления составной деятельности посредством композиции действий и жестов более низкого уровня в соответствии с ее семантическим значением. Затем эта иерархическая структура преобразуется в формальные синтаксические логические формулы и правила, на основе которых применяется автоматическое рассуждение, основанное на разрешении, для распознавания составного действия с учетом распознанных действий более низкого уровня с использованием методов машинного обучения, основанных на данных.

В данной статье [24] предлагается метод распознавания человеческой активности в видеопотоке. Чтобы достичь высокой точности в результатах распознавания действий, метод на своем начальном этапе использует сопоставление временных шаблонов для распознавания действий. Поскольку временные шаблоны подвержены влиянию скорости, стиля и характера выполнения деятельности, становится трудно точно различать очень похожие виды деятельности (например, ходьбу, бег и пробежку трусцой). Путаница в распознавании действий устраняется последующим различением действий на основе правил.

Предлагаемый метод распознает действия человека в видео на различных наборах эталонных данных, включая набор данных КТН и набор данных Вейцмана. Экспериментальные результаты демонстрируют высокую распознаваемость действий. Средняя точность метода составляет 97,20% при стандартных условиях.

В статье [25] описывается классификатор, способный распознавать статические позы человеческого тела и жесты тела. Этот метод называется языком описания жестов (GDL). Предлагаемая методика интуитивно понятна, легко продумывается и может быть использована для любого вида телодвижений. В основе данного подхода лежит модуль автоматического рассуждения. Он выполняет рассуждения с прямой цепочкой (подобно классической экспертной системе) с помощью своего механизма вывода каждый раз, когда поступает новая порция данных из библиотеки извлечения объектов. Все правила базы знаний организованы в скриптах GDL, имеющих форму текстовых файлов, которые анализируются с помощью грамматики LALR-1. Уровень распознавания изученных действий находится в диапазоне 80,5-98,5%.

1.4.4 Методы, основанные на форме

Хорошо известно, что алгоритмы распознавания активности, основанные на силуэте человека, играют важную роль в распознавании действий человека. Поскольку человеческий силуэт состоит из соединенных друг с другом конечностей, важно получить точные части человеческого тела из видео. Эта проблема рассматривается как часть процесса распознавания действий.

Подходы, основанные на деталях, учитывают информацию о движении как всего человеческого тела, так и отдельных частей тела. Преимущество этой линии подходов заключается в том, что она по своей сути отражает геометрические взаимосвязи между частями тела, что является важным сигналом для различения человеческих действий.

Икизлер и Дуйгулу [26] смоделировали человеческое тело как последовательность ориентированных прямоугольных участков. Авторы описали вариацию метода BoW (мешок слов), называемую мешком прямоугольников.

Пространственно ориентированные гистограммы были сформированы для описания действий человека, в то время как классификация действий выполнялась с использованием четырех различных методов, таких как голосование по кадрам, глобальное гистограммирование, классификация методом опорных векторов и динамическое искажение времени (DTW). Метод был протестирован на наборе данных UCI HAR и показал точность результатов 95%.

В статье [27] Ванг и Мори представляют основанный на возможности различать детали подход к распознаванию действий человека по видеопоследовательностям с использованием признаков движения. Модель, представленная в этом методе, основана на скрытом условном случайном поле (hCRF) для распознавания объектов. Подобно hCRF для распознавания объектов, происходит моделирование действия человека с помощью гибкой совокупности деталей, обусловленных наблюдениями за изображением. В отличие от распознавания объектов, эта модель сочетает в себе как крупномасштабные глобальные функции, так и локальные функции исправления для различения различных действий.

Предлагаемый метод распознавания действий человека в видео тестировался на различных наборах эталонных данных, включая набор данных КТН и набор данных Вейцмана. Экспериментальные результаты демонстрируют точность распознавания 73% и 88% соответственно.

Несмотря на широкое развитие алгоритмов оценки позы, проблема все еще остается сложной для приложений реального времени. В статье [28] представлен метод быстрой оценки позы человека с частотой 1000 кадров в секунду. Для достижения такой высокой вычислительной скорости авторы использовали методы выборочной выборки случайного блуждания. Части человеческого тела обрабатывались как представления с направленной древовидной структурой, и для каждого сустава человеческого скелета было подготовлено дерево регрессии. Однако этот метод зависит от инициализации процесса случайного блуждания.

1.5 Сравнение методов

Было выделено 4 категории методов, описанных в главе выше. Определены два критерия, применимых к категориям, такие как необходимость

детектирования человека и устойчивость методов к шуму на изображении.

Далее проводилось сравнение конкретных рассмотренных методов из каждой категории по следующим критериям: использованный метод машинного обучения, модальность данных, тестируемый набор данных, точность распознавания.

Сравнения методов распознавания физической активности человека представлено ниже в таблице 1.1.

Таблица 1.1 – Сравнение методов распознавания действий человека

Категория	Рассмотренный метод	Метод машинного обучения	Модальность данных	Набор данных	Точность
Пространственно-временные	Соомро К., Шах М. [14]	Метод опорных векторов	RGB	UCF-Sports, Sub-JHMDB, JHMDB, THUMOS13	63,2% 45,9% 67,3% 54,4%
	Х. Идресс М. Шах [13]	Метод опорных векторов	RGB	UCF-Sports sub-JHMDB THUMOS13	55% 42% 48%
Стохастические	Тахир С., Догар А. [16]	Метод случайного леса	RGB	IM-WSHA, PAMAP-2, UCI HAR, MobiAct, MOTIONSENSE	90,18% 91,25% 91,83% 90,46% 92,16%
	Ю. А. Егорова, И. Г. Захарова, А. Р. Гасанов, А. А. Филиппин[3]	Скрытая Марковская модель, сверточная нейронная сеть	RGB, скелетная модель	MPII Human Pose Dataset	75%
Основанные на правилах	Sharma С. М., Kushwaha А. [19]	GMM модель, нейронная сеть	RGB	KTH набор данных Вейцмана	97,2% 91%
	Park, J. Jin, Y. Cho, S. [20]	Скрытая Марковская модель, FNN	RGB, скелетная модель	LALR-1	80,5%
Основанные на форме	Wang, Y., Mori, G.[22]	Марковские случайные поля	RGB	KTH, набор данных Вейцмана	73% 88%
	Ikizler, N., Duygulu, P.[21]	Метод опорных векторов, сверточная нейронная сеть	RGB	Набор из 8 действий	95%

1.6 Выбор данных для обучения модели

В машинном обучении самый главный и критичный компонент – сами данные, на которых проводится обучение. Качество данных напрямую влияет на эффективность и надежность получаемых моделей. Поэтому значительная часть работы аналитика – это сбор, очистка, анализ и улучшение данных. В реальных проектах на это уходит до 80% всего времени, затраченного на проект. И лишь оставшиеся 20% – это обучение и выбор моделей, диагностика и повышение их эффективности.

1.6.1 Требования

Одним из требований, которым должен удовлетворять набор данных это реалистичность.

Главное правило любого набора данных – данные должны быть максимально приближены к реальным условиям, в которых будет работать обученная модель. До начала сбора данных важно знать какие именно видео будет получать на вход модель, где будет стоять камера, разрешение камеры.

Важно понимать, что если камеры могут иметь различные разрешение, а также человек выполняющий упражнение может снимать себя с разных ракурсов, то и видео в наборе должны учитывать это и быть различных размеров, разрешения и сняты под разными углами обзора.

Пренебрежение данным шагом может понизить точность распознавания действий, даже если модель обучалась на большом наборе данных [29].

Еще одним важным требованием является достаточность. Под достаточностью понимается объем набора данных. Он должен быть достаточно большим для того, чтобы модель учитывала различные видео.

1.6.2 Структура

Первичный набор исходных данных принято называть генеральной совокупностью. Процесс формирования выборок из генеральной совокупности называется порождением данных. Выборка – это конечное подмножество элементов генеральной совокупности, изучив которое можно понять поведение исходного множества.

Вероятностная модель порождения данных предполагает, что выборка

из генеральной совокупности формируется случайным образом. Если все ее элементы одинаково случайно и независимо друг от друга распределены по исходному множеству (генеральной совокупности), выборка называется простой. Простая выборка является математической моделью серии независимых опытов и, как правило, используется для машинного обучения. При этом для каждого этапа машинного обучения необходим свой набор данных [30]:

- для непосредственного обучения модели нужна обучающая выборка (training sample), по которой производится настройка (оптимизация параметров) алгоритма;
- для оценки качества модели используется тестовая (контрольная) выборка (test sample), которая, в идеальном случае, не должна зависеть от обучающей;
- для выбора наилучшей модели машинного обучения понадобится проверочная (валидационная) выборка (validation sample), которая также не должна пересекаться с обучающей.

Вывод

Были рассмотрены сферы применения методов распознавания действий человека. Были рассмотрены алгоритмы детектирования объектов с помощью дескрипторов, алгоритмы классификаторов, их достоинства и недостатки. Была проведена классификация уже существующих методов распознавания действий человека на видео, описаны требования к набору данных для обучения и его структура.

В качестве технологий для разрабатываемого метода выбраны метод НОГ дескриптора для детектирования силуэта человека на видео и для классификации полученных дескрипторов видео выбран метод случайного леса.

2 Конструкторский раздел

В данном разделе описываются требования к разрабатываемому методу. Рассматриваются особенности разрабатываемого метода, его архитектура и описываются ключевые шаги метода в виде схем алгоритмов.

Требования к разрабатываемому методу

Метод распознавания спортивных действий человека на видео с использованием локального дескриптора должен:

- принимать на вход видео в форматах mp4;
- разбивать видеоролик на кадры;
- вычислять для каждого кадра дескриптор точек интереса;
- проводить классификацию полученных дескрипторов.

2.1 Проектирование метода распознавания

Формализуем постановку задачи в виде IDEF0-диаграмм нулевого и первого уровня и опишем входные и выходные данные метода распознавания спортивных действий человека на видео. Рассмотрим каждый из этапов реализации метода распознавания спортивных действий человека на видео. Для каждого этапа предоставляется схема алгоритма, решающая задачу, поставленную на этом этапе, а также более подробно описываются действия, представленные на схемах.

2.1.1 IDEF0

На рисунке 2.1 представлена диаграмма IDEF0 метода распознавания спортивных действий человека нулевого уровня.

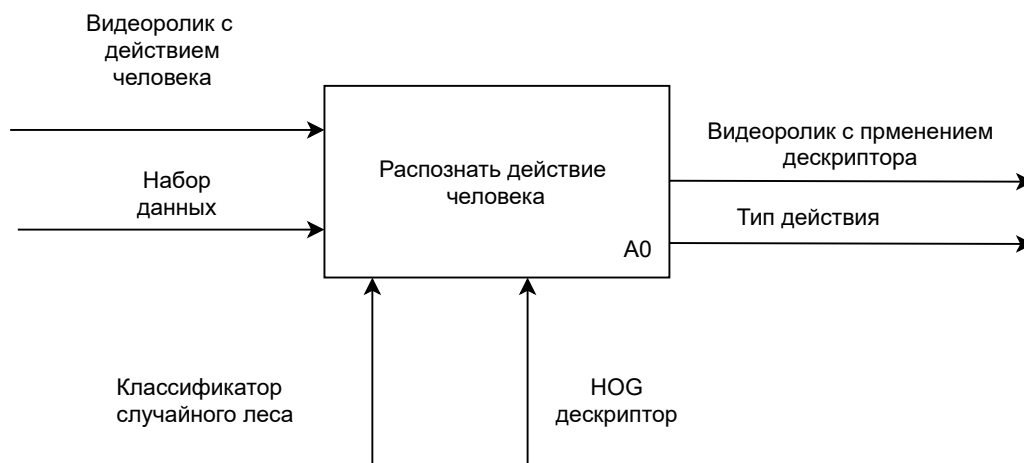


Рисунок 2.1 – IDEF0 нулевого уровня

На вход методу поступает набор данных для обучения модели и видео, которое после предобработки поступает на вход функции формирования дескриптора. Также кадры видео, полученные в результате предобработки видео поступают на вход функции формирования исходного видео с наложенными градиентами интенсивностей. Полученные дескрипторы передаются на вход классификатора. На выходе метода: предсказанный тип действия и видео с градиентами интенсивностей.

На рисунке 2.2 представлена диаграмма IDEF0 разрабатываемого метода первого уровня.



Рисунок 2.2 – IDEF0 первого уровня

2.1.2 Предобработка видеоролика

Для того, чтобы классифицировать действие человека на видео, необходимо разделить видеоролик на кадры для дальнейшей их обработки. Для получения кадров необходимо определить частоту кадров (FPS – frames per second). Частота кадров показывает сколько раз изображение появляется на экране в течение секунды. На рисунке 2.3 представлен алгоритм получение кадров видео.

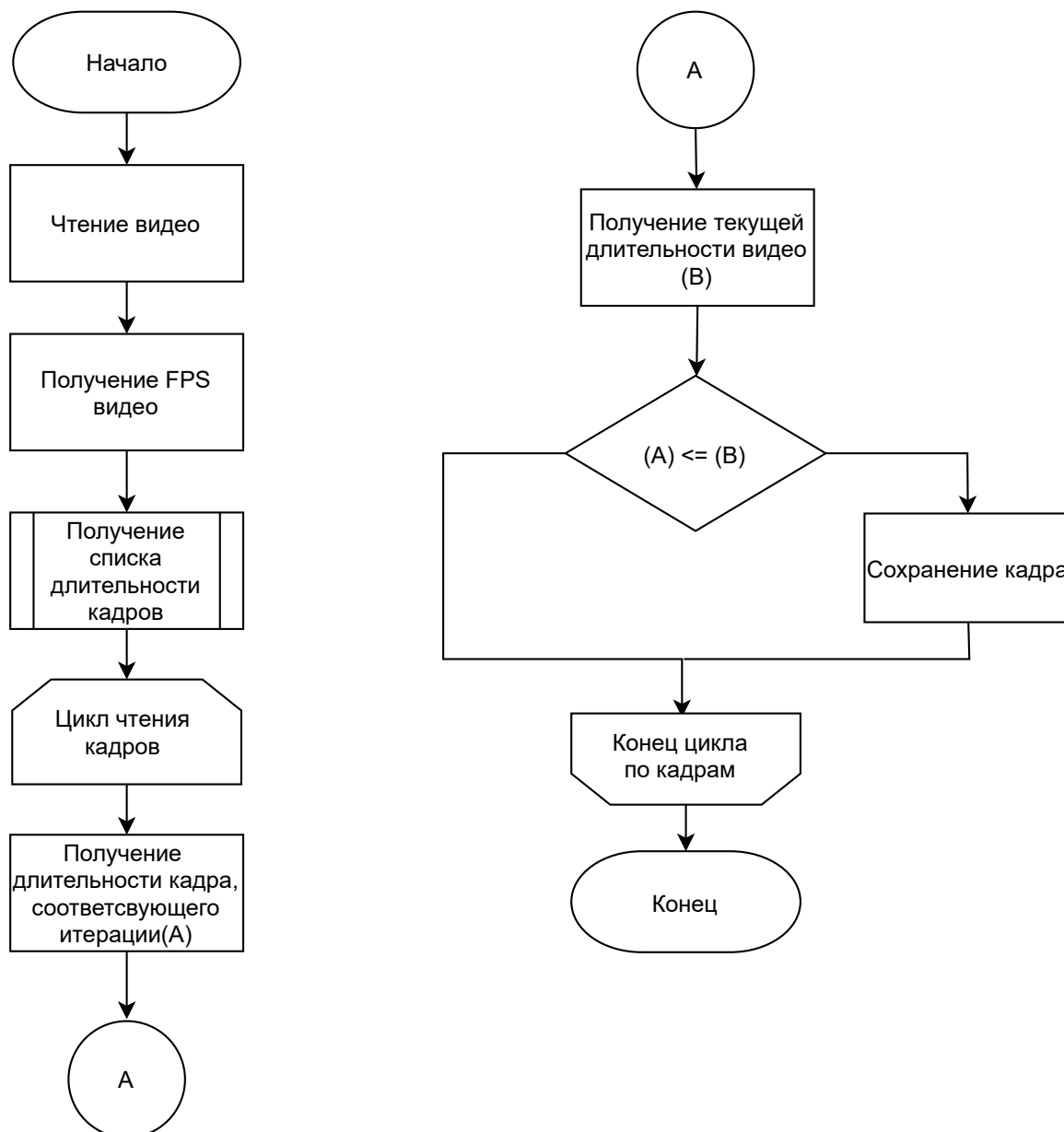


Рисунок 2.3 – Алгоритм формирования кадров видео

Для более точного распознавания необходимо получать дескриптор не всего кадра, а его части, в котором находится человек. Это обусловлено тем, что без обрезания кадра дескриптор будет больше по размеру и иметь ненужные данные, которые негативно повлияют на результат обучения модели. Так же для того, чтобы обеспечить корректную работу алгоритма НОГ в случае если размер кадра видео не кратен 16, необходимо изменить его размер на кратный 16, так как алгоритм вычисляет гистограмму ориентированных градиентов, разбивая изображения на области 16x16 пикселей. На рисунке 2.4 представлен алгоритм обрезания кадра.

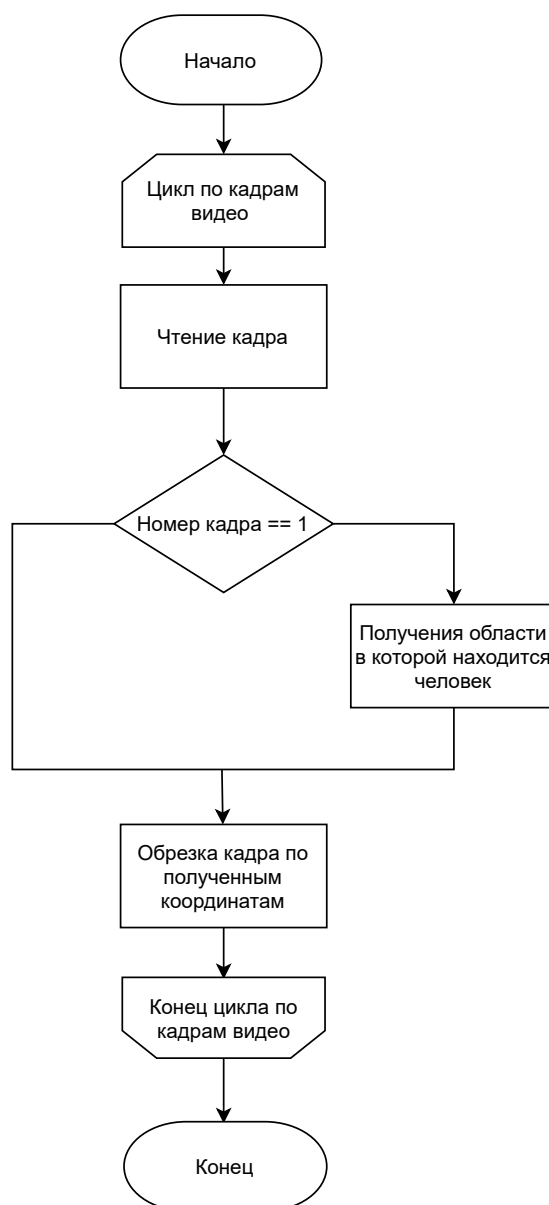


Рисунок 2.4 – Алгоритм обрезания кадра

2.1.3 Формирование НОГ дескриптора видео

Для обучения модели, способной делать предсказания типа действия, совершаемого человеком на видео, необходимо сформировать дескрипторы видео, которые состоят из дескрипторов всех кадров данного видео. Дескриптор кадра представляет собой объединение гистограмм ориентированных градиентов, полученных для каждого блока кадра размером 16×16 пикселей. Гистограмма является вектором 36×1 . Процесс формирования гистограмм подробно описан в разделе 1.3.4.

В результате для каждого видео будет сформирован дескриптор, состоящий из гистограмм ориентированных градиентов, с помощью которых можно будет определить силуэт человека и вид действия, который совершает человек. На рисунке 2.5 представлен алгоритм формирования НОГ дескриптора видео.



Рисунок 2.5 – Алгоритм формирования HOG дескриптора

2.1.4 Обучение модели

Обучающая выборка видеороликов должна была пройти этап предобработки, после которого формировались данные для обучения, а именно дескрипторы видео и метки соответствующих действий. Проинициализовав модель классификатора, ей передаются данные полученные в результате формирования дескрипторов. И после обучения файл готовой модели сохраняется. В качестве классификатора используется модель случайного леса.

Алгоритм построения случайного леса, состоящего из N деревьев, выглядит следующим образом:

Для каждого $n = 1, \dots, N$:

- сгенерировать выборку X_n с помощью бутстрэпа;
- построить решающее дерево b_n по выборке X_n :
 - по заданному критерию выбирается лучший признак, происходит разбиение дерева по этому критерию и так до исчерпания выборки;
 - дерево строится, пока в каждом листе не более n_{\min} объектов или пока не достигается определенная высота дерева;
 - при каждом разбиении сначала выбирается m случайных признаков из n исходных, и оптимальное разделение выборки ищется только среди них.

Формула итогового классификатора:

$$a(x) = \frac{1}{N} \sum_{i=1}^N b_i(x). \quad (2.1)$$

Для задачи классификации результат определяется голосованием по большинству.

Алгоритм получения обученной модели представлен на рисунке 2.6.

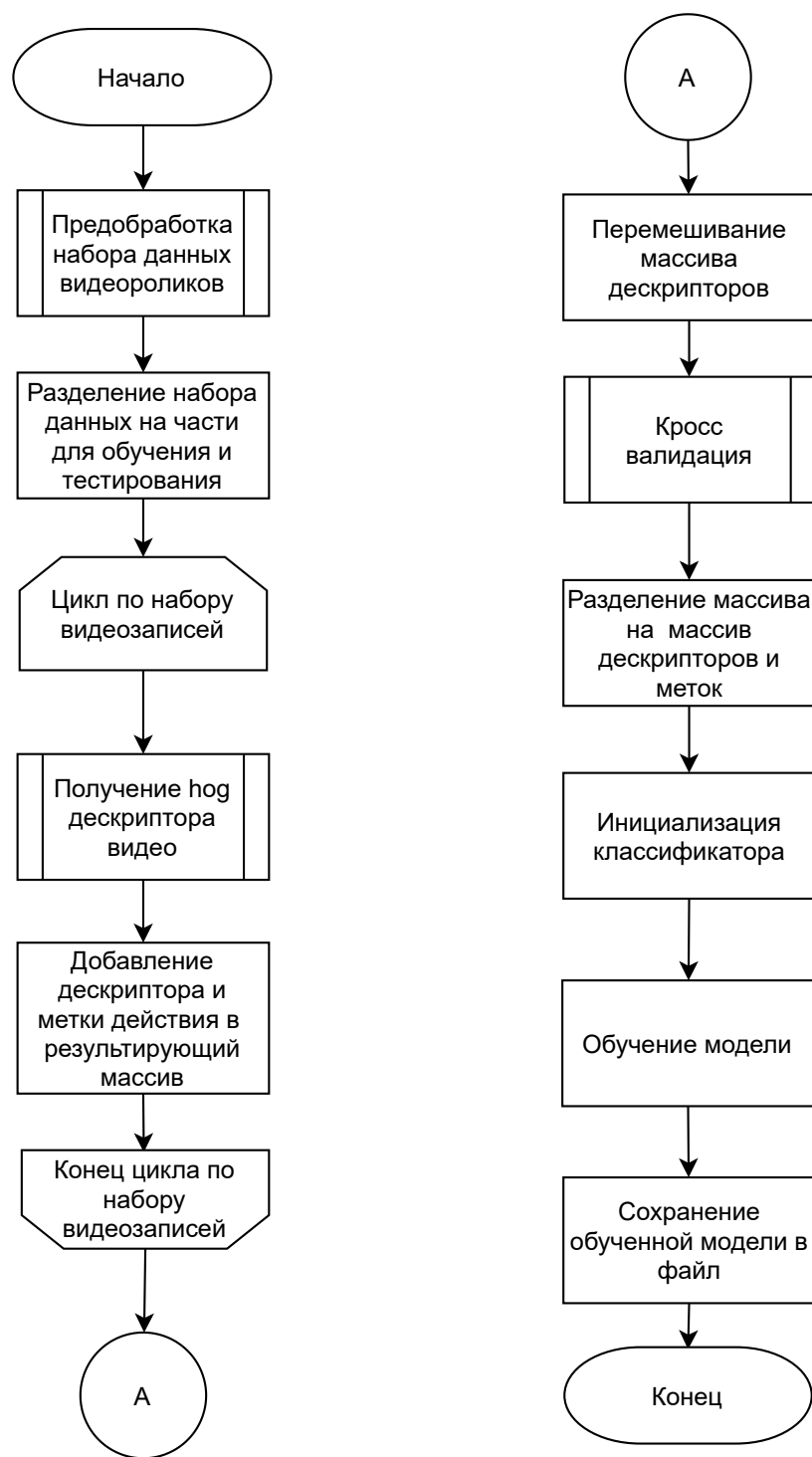


Рисунок 2.6 – Алгоритм получения обученной модели

Алгоритм кросс-валидации (перекрестной проверки) предназначен для оценки качества работы модели. На рисунке 2.7 представлен этот алгоритм.

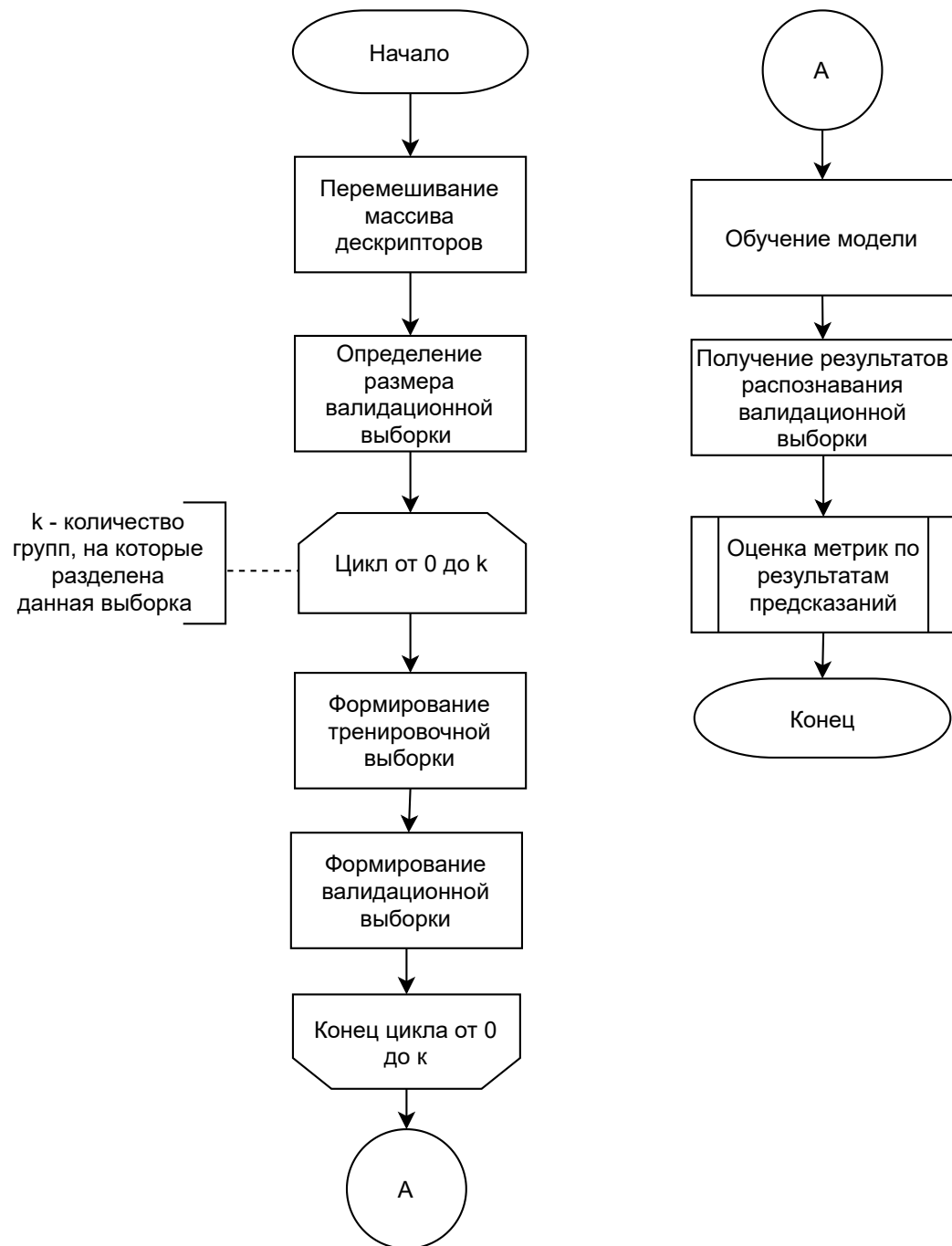


Рисунок 2.7 – Алгоритм k-fold кросс-валидации

2.2 Набор данных

Набор данных на котором будет обучаться модель был собран из трех открытых наборов данных:

- UCF101 – это набор данных для распознавания действий, состоящий из реалистичных видеороликов, собранных с YouTube и имеющих 101 категорию действий. UCF101 обеспечивает наибольшее разнообразие с точки зрения действий и с наличием больших различий в движении камеры, внешнем виде объекта и позе, масштабе объекта, точке обзора, загроможденном фоне, условиях освещения и т.д.;
- Набор данных HMDB51 представляет собой большую коллекцию реалистичных видеороликов из различных источников, включая фильмы и веб-видео. Набор данных состоит из 6766 видеоклипов из 51 категории действий, причем каждая категория содержит не менее 101 клипа;
- Kinetics400 – это набор данных для распознавания действий, состоящий из реалистичных видеороликов, собранных с YouTube. С 306 245 короткими обрезанными видеороликами из 400 категорий действий, это один из крупнейших и наиболее широко используемых наборов данных в исследовательском сообществе для сравнения современных моделей распознавания видео действий.

Так как наборы данных содержат множество разных видов действий среди них, необходимо было выбрать различные фитнес упражнения. Итоговая выборка, на которой происходило обучение модели, содержит следующие виды спортивных действий:

- мостик (108 видео);
- поднятие рук (330 видео);
- джампинг джек (123 видео);
- выпады (211 видео);
- подтягивания (178 видео);

- отжимания (139 видео);
- пресс (146 видео);
- приседания (122 видео);
- растяжка ног (154 видео);
- махи ногами (129 видео).

Всего выборка содержит 1650 видео в списке выше указано конкретное количество видео содержащий каждый класс действий. Перед обучением модели выборка была разделена в отношении 70% для обучения и 30% для тестирования.

Вывод

Были представлены требования к разрабатываемому методу распознавания спортивных действий человека на видео. Обозначены особенности разрабатываемого метода, его архитектура и описаны ключевые шаги метода в виде схем алгоритмов. Описана структура выбранного набора данных для обучения модели.

3 Технологический раздел

В данном разделе производится выбор средств программной реализации метода распознавания спортивных действий человека на видео, описывается формат входных и выходных данных. Приводятся детали реализации программных компонентов, предоставляются результаты тестирования разработанного метода, описывается взаимодействие пользователя с интерфейсом ПО, реализующим метод.

3.1 Входные и выходные данные

Входными данными является видео, на котором человек выполняет фитнес упражнения. Видео может быть любого размера, вертикальное или горизонтальное, формата mp4. Видео должно быть с нормальным освещением и одним человеком в кадре.

Выходные данные – название класса вида действия на видео, к которому оно относится с наибольшей вероятностью и видео сформированное из кадров, на которых изображены градиенты интенсивностей каждого пикселя.

3.2 Выбор языка программирования

В качестве языка реализации разрабатываемого метода был выбран Python версии 3.11. Использование данного языка обусловлено тем, что он часто используется в области машинного обучения и имеет доступную документацию и библиотеки данной предметной области.

В разработанном ПО были использованы следующие библиотеки и фреймворки:

- Scikit-learn [31] является библиотекой для работы с машинным обучением. Была использована при реализации метода случайных лесов;
- OpenCV [32] представляет собой библиотеку для работы с компьютерным зрением, обработки изображений и набором сопутствующих алгоритмов;
- Flask [33] представляет собой фреймворк для создания веб-приложений на языке программирования Python, использующий набор инструментов Werkzeug, а также шаблонизатор Jinja2;

- Scikit-image [34] представляет собой библиотеку обработки изображений с открытым исходным кодом для языка программирования Python.

3.3 Разработка ПО

Требования к разрабатываемому ПО

ПО, предоставляющие интерфейс для разрабатываемого метода должно предоставлять:

- возможность загрузить видео формата mp4;
- возможность просмотреть какие виды действий могут быть распознаны;
- воспроизводить загруженное видео и видео с градиентами интенсивностей;
- выводить предсказанный вид действия.

На рисунке 3.1 представлена UML диаграмма вариантов использования разработанного ПО, доступные пользователю действия описаны выше.



Рисунок 3.1 – UML диаграмма вариантов использования разработанного ПО

Структура разработанного ПО

ПО было поделено на следующие основные модули.

- Предобработка данных.
- Формирование дескрипторов видео.
- Обучение модели.
- Классификатор видов действий.
- Графический интерфейс.

3.3.1 Модуль предобработки данных

Исходный набор данных содержит видеоролики различного размера кадра, в среднем длительностью 10 секунд. Для получения дескриптора видео, оно должно пройти следующие этапы предобработки:

- разделение видео на кадры;
- обрезание кадра по области, в которой находится человек;
- изменение размера кадра на кратное 16;
- получение черно-белого кадра.

На рисунке 3.2 представлен пример преодобработки видео.



Рисунок 3.2 – Предобработка данных

На этапе получение кадров использовалась библиотека OpenCV для чтения видео. Для каждого видео вычислялся его fps и формировался массив длительностей каждого кадра. Далее в цикле для каждого кадра вычислялось

его время от начала видео и в случае если время было меньше длительности самого видео, кадр сохранялся.

Для определения области кадра, в которой находится человек был использован метод библиотеки OpenCV, который называется selectROI. С помощью этого метода можно вручную выбрать область изображения, представляющую интерес, путем выделения области на изображении. После этого координаты ограничивающего прямоугольника сохраняются и все последующие кадры обрезаются по полученным координатам. На листинге 3.1 представлена функция обрезания кадра.

Листинг 3.1 – Обрезание кадра

```
def parserman(tPath):
    hog = cv2.HOGDescriptor()
    hog.setSVMDetector(
        cv2.HOGDescriptor_getDefaultPeopleDetector())
    path2 = tPath + "/man"
    if not os.path.exists(path2):
        os.makedirs(path2)
    count = 0
    imageFrames = getImageList(tPath)
    for imagePath in imageFrames:
        path3 = path2 + "/" + str(count) + ".jpg"
        image = cv2.imread(imagePath)
        if count == 0:
            imagedraw = cv2.selectROI(image)
            croppedimage =
                image[int(imagedraw[1]):int(imagedraw[1]+imagedraw[3]),
                    int(imagedraw[0]):int(imagedraw[0]+imagedraw[2])]
            cv2.imwrite(path3, croppedimage)
            cv2.waitKey(0)
            cv2.destroyAllWindows()
            count += 1
```

Преобразование размера и цвета кадра происходило с помощью методов библиотеки OpenCV `resize` и `cvtColor`.

3.3.2 Модуль формирования дескрипторов

Дескриптор видео формируется на основе кадров видео прошедших этап предобработки и представляет собой массив, элементом которого является

дескриптор кадра (массив чисел с плавающей точкой). Для каждого видео дескриптор формируется по 25 кадрам, так как за этот промежуток происходит одно или пару повторяющихся действий и этого достаточно для обучения модели.

В случае формирования дескриптора для обучения в начало массива дескриптора кадра вставляется метка, соответствующая данному типу действия. Формирование дескрипторов производилось многопоточно (10 потоков) и в результате дескрипторы каждого класса действий были сохранены в файлы. На листинге 3.2 представлена функция формирования дескрипторов каждого класса для их сохранения.

Листинг 3.2 – Сохранение дескрипторов тренировочной выборки

```
def get_file_features(sportsActionName):
    sportsActionFeatures = []
    firstActionFlag = 0
    sportsActionDir = sportsActionPath + "/" + sportsActionName
    sportsActionFeatures = []
    videoList = getListOfDir(sportsActionDir)
    videoCount = 1
    videoFeatures = []

    for video in videoList:
        videoPath = sportsActionDir + "/" + video
        videoFeatures = featureExtraction(videoPath ,
            sportsActionName, 'Trng')
        if firstActionFlag == 0:
            sportsActionFeatures = videoFeatures
            firstActionFlag = 1
        else:
            sportsActionFeatures = np.concatenate(
                (sportsActionFeatures, videoFeatures), axis=0)
        videoCount += 1
    with open(f'{sportsActionName}.npy', 'wb') as f:
        np.save(f, sportsActionFeatures)
```

3.3.3 Модуль обучения модели

Обучение модели проводилось с помощью встроенного классификатора случайного леса библиотеки Scikit-learn RandomForestClassifier.

Для обучения модели необходимо было определить параметры, при которых модель обеспечивает наилучшую точность, это можно было сделать используя поиск по сетке. Поиск по сетке (Grid Search) – это алгоритм оптимизации, который позволяет определить лучшие параметры для оптимизации проблемы из списка вариантов, который задан, тем самым автоматизируя метод «проб и ошибок». На листинге 3.3 представлена реализация поиска по сетке.

Листинг 3.3 – Поиск по сетке

```
classifier = RandomForestClassifier(n_estimators=300, verbose=2,
    random_state=0)
grid_param = {
    'n_estimators': (100, 300, 500, 800, 1000),
    'criterion': ['gini', 'entropy'],
    'bootstrap': [True, False]
}
gd_sr = GridSearchCV(estimator=classifier,
    param_grid=grid_param,
    scoring='accuracy',
    cv=5,
    n_jobs=-1)

clf=gd_sr.fit(feature, labels)
joblib.dump(clf, 'model.joblib')
best_parameters = gd_sr.best_params_
print(best_parameters)
best_result = gd_sr.best_score_
print(best_result)
```

Был проведен поиск по сетке в результате, которого в качестве оптимальных параметров классификатора были определены следующие значения: `bootstrap = false`, `criterion = entropy`, `n_estimators = 1000`.

Модель способна распознавать действия для каждого кадра видео и с помощью функции `mode()` определяется наиболее часто встречающееся метка действия среди 25 кадров, действие соответствующие этой метки возвращается как результат распознавания типа действия на видео.

Перед обучением производится чтение дескрипторов из файлов, полученных в результате обработки тренировочного набора данных, и их объединение. Также происходит разделение полученного массива данных на дескрипторы

и соответствующие им метки. Массив меток и дескрипторов передаются в функцию обучения модели, и уже обученная модель сохраняется в файл.

3.4 Тестирование модели

На тестовом наборе данных, составляющем 30% исходного набора, проводилось тестирование модели в результате которого была сформирована матрица ошибок. Так как в каждом классе содержалось разное количество видео, для тестирования каждого класса было выбрано 30% от соответствующего объема видеороликов этого класса.

Матрица ошибок – это показатель успешности классификации, где классов два или более. Она представляет собой таблицу с 4 различными комбинациями сочетаний, прогнозируемых и фактических значений.

Матрица ошибок формировалась для многоклассовой классификации из 10 классов. В алгоритме кросс-валидации класс, который представляет для нас интерес, называется «положительным», а оставшиеся – «отрицательными».

Поэтому для каждого объекта в выборке возможно 4 ситуации:

- предсказана положительная метка и это верно. Такие объекты относятся к истинно-положительной (TP, true positive) группе. Истинно потому что предсказание верное, а положительный, потому что предсказываем объект положительного класса;
- предсказана положительная метка, но предсказание ошибочно – ложно-положительный (FP, false positive). Ложное, потому что предсказание было неправильным;
- предсказана отрицательная метка и это верно – истинно-отрицательный (TN, true negative);
- предсказана отрицательная метка, но предсказание ошибочно – ложно-отрицательный (FN, false negative).

Так в результате сформирована матрица ошибок, где на главной диагонали показано количество объектов положительного класса (TP), которые верно предсказаны, а на пересечение видов действий количество ошибочных

предсказаний (FN). В качестве объекта предсказания принимается кадр видео. На рисунке 3.3 представлена матрица ошибок модели распознавания спортивных действий человека.

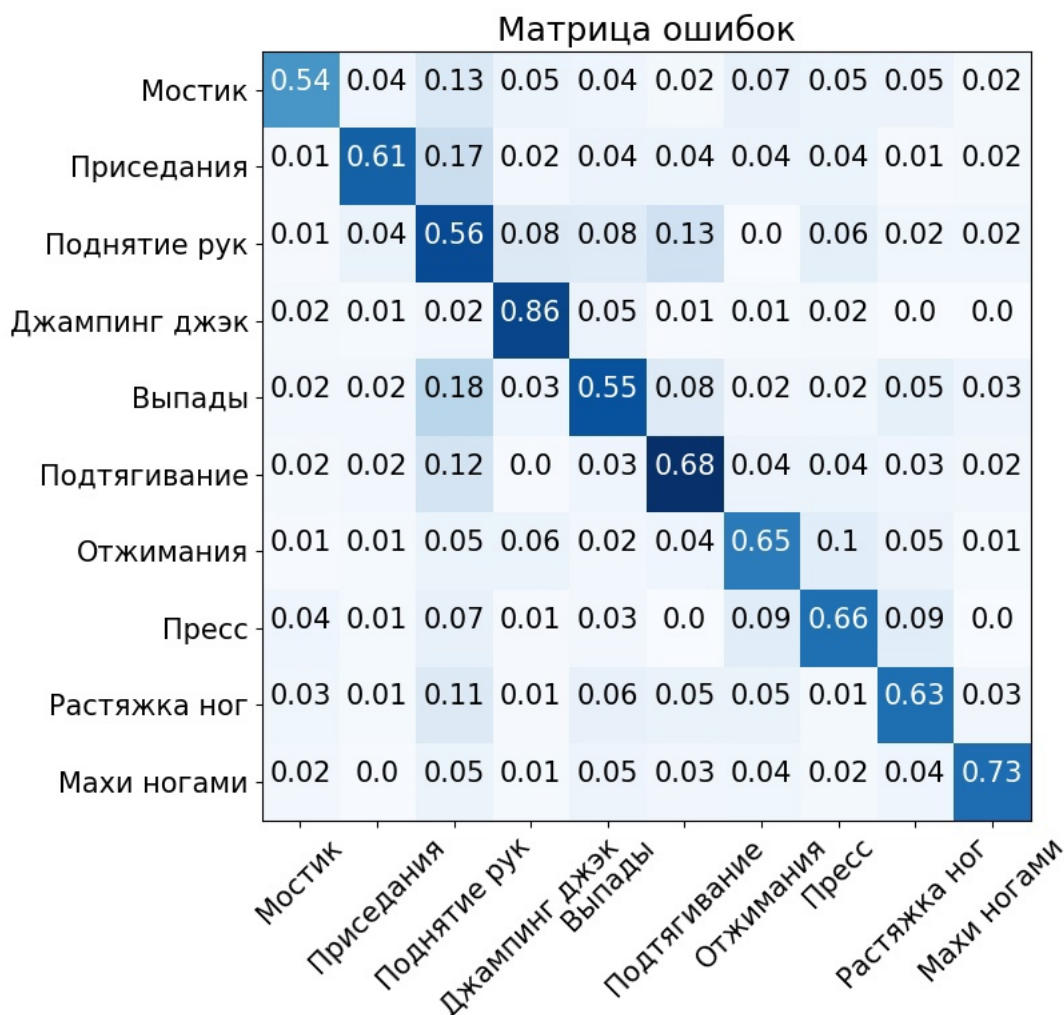


Рисунок 3.3 – Матрица ошибок

3.5 Пользовательский интерфейс

Интерфейс был реализован с помощью веб-фреймворка Flask. На сайте пользователю доступно две страницы. На странице справка описаны характеристики видео, которое нужно загрузить для успешного распознавания, а также все виды действий, которые подлежат распознаванию с видео примерами. На рисунке 3.4 представлена страница со справочной информацией.

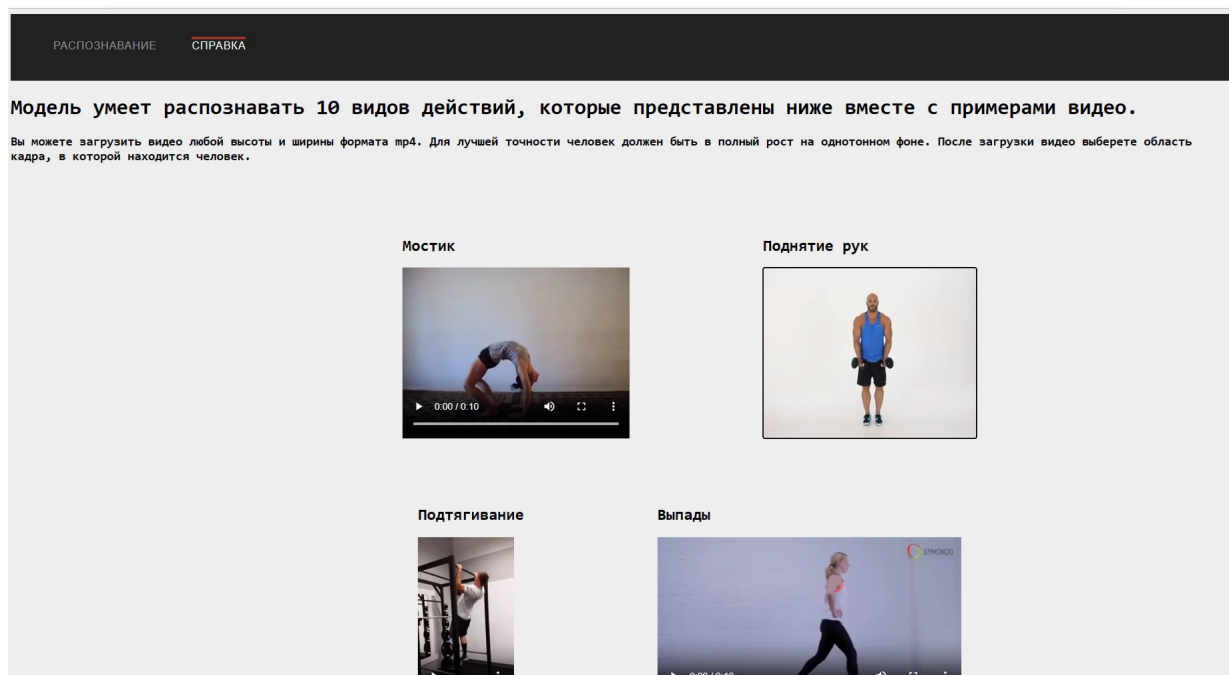


Рисунок 3.4 – Страница справка

На главной странице есть форма с помощью, которой пользователь может загрузить видео для предсказания вида действия со своего локального компьютера, нажав кнопку выбрать файл. На рисунке 3.5 представлена главная страница сайта.

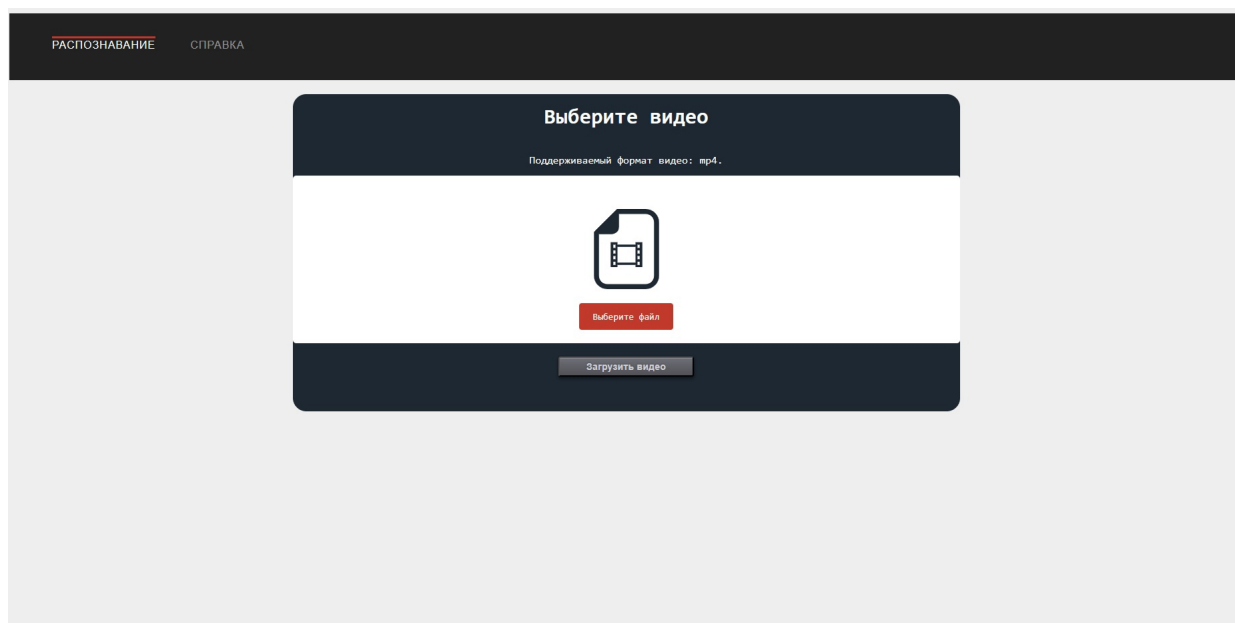


Рисунок 3.5 – Главная страница

При нажатии кнопки выбрать файл открывается диалоговое окно, в котором пользователь должен выбрать нужный ему видеоролик. Далее нажать кнопку загрузить видео и ожидать результат предобработки и предсказания вида действия. На рисунке 3.6 представлен процесс загрузки видео.

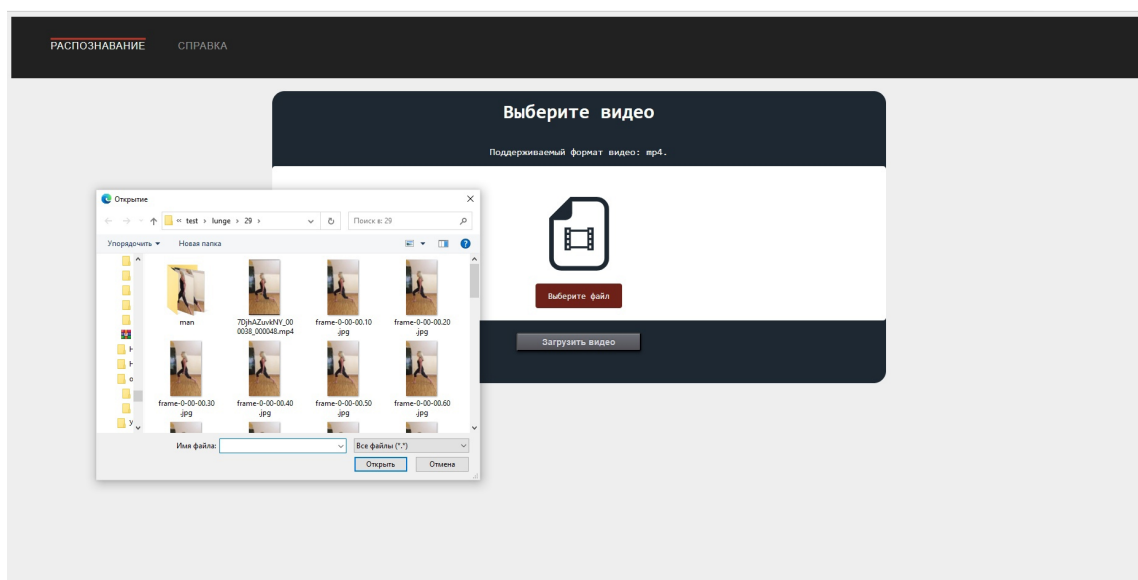


Рисунок 3.6 – Загрузка видео

В результате на странице пользователю представлено исходное видео и предсказанный вид действия. Также для визуального представления помимо основного видео формируется видео из кадров, на которых изображены направления интенсивности, используемые для формирования дескриптора. На рисунке 3.7 представлен результат работы ПО.

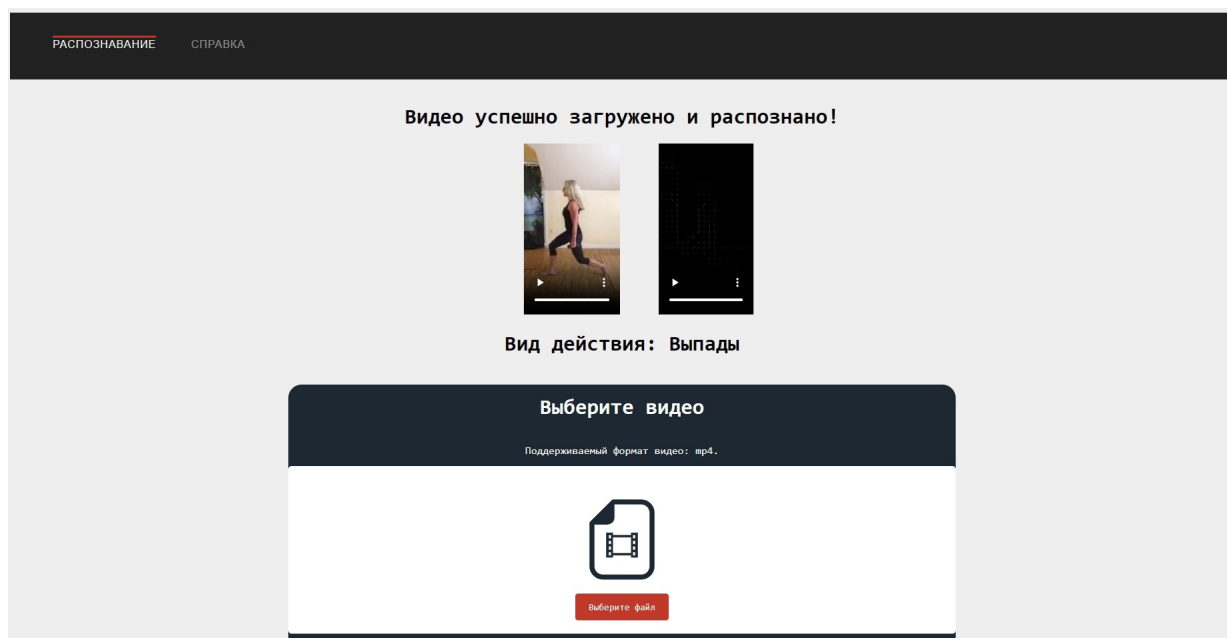


Рисунок 3.7 – Результат распознавания

Вывод

В данном разделе произведен выбор средств программной реализации метода, описан формат входных и выходных данных ПО. Приведены детали реализации программных компонентов, предоставлены результаты тестирования разработанного метода, описано взаимодействие пользователя с интерфейсом ПО, реализующим метод.

4 Исследовательский раздел

В данном разделе проводится исследование эффективности разработанного метода распознавания спортивных действий человека. Проводится исследование метрик модели в зависимости от разметки данных. Описываются исследования характеристик разработанного метода, а именно исследование результатов распознавания модели в зависимости от контрастности видео и исследование результатов распознавания модели в зависимости от качества видео.

4.1 Исследование метрик модели в зависимости от разметки данных

В качестве метрик оценки классификации были выбраны общая точность предсказания модели (accuracy), точность (precision) и полнота (recall). Величина метрик вычислялась, как среднее значение метрик на каждой итерации алгоритма кросс-валидации.

Перекрестная проверка (кросс-валидация) – это процедура повторной выборки, используемая для оценки моделей машинного обучения на ограниченной выборке данных.

Процедура имеет один параметр, называемый k , который относится к числу групп, на которые должна быть разделена данная выборка данных. Таким образом, процедура часто называется k -fold кросс-валидацией (перекрестной проверкой).

Так как основной набор данных содержал неодинаковое количество видео в каждом классе, использовался метод стратификации, а именно при разбиении на тестовый и тренировочный набор на каждой итерации алгоритма, данные распределены так, что каждый набор содержит примерно такое же соотношение классов, как и всё исходное множество. Каждый кадр видео после предсказания относится к одной из ситуаций, которые более подробно описаны в разделе 3.4:

- TP – истинно-положительное решение;
- TN – истинно-отрицательное решение;
- FP – ложно-положительное решение;

— FN – ложно-отрицательное решение.

На каждой итерации цикла кросс-валидации на основе вычисленных выше описанных переменных можно рассчитать величины метрик.

Если рассмотреть долю правильно предсказанных положительных объектов среди всех объектов, предсказанных положительным классом, то можно получить метрику, которая называется точностью (precision). Точность рассчитывается по следующей формуле:

$$Precision = \frac{TP}{TP + FP}. \quad (4.1)$$

Доля правильно найденных положительных объектов среди всех объектов положительного класса, называется полнотой (recall). Полнота рассчитывается по следующей формуле:

$$Recall = \frac{TP}{TP + FN}. \quad (4.2)$$

Ассурасу – это показатель, который описывает общую точность предсказания модели по всем классам. Это особенно полезно, когда каждый класс одинаково важен. Он рассчитывается как отношение количества правильных прогнозов к их общему количеству:

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FT}. \quad (4.3)$$

Метрики рассчитывались для двух моделей, обученных на данных с разной разметкой. В первом случае при формировании дескрипторов метка назначалась на каждое видео, в другом на дескриптор каждого кадра, в этом случае предсказанное действие выбиралось как чаще всего встречающиеся предсказанная метка среди кадров. Результаты исследования представлены на рисунке 4.1.

По столбчатым диаграммам можно заметить, что при назначении метки на каждый кадр метрика полнота выросла в два раза. Это показывает, что при назначении метки на каждое видео меньше половины объектов положительного класса были не распознаны моделью, это является плохим показателем, при разметки на каждый кадр метрика растет до 76%, что уже является оптимальным показателем. Остальные метрики при разметки на каждый кадр также выше, но не значительно.

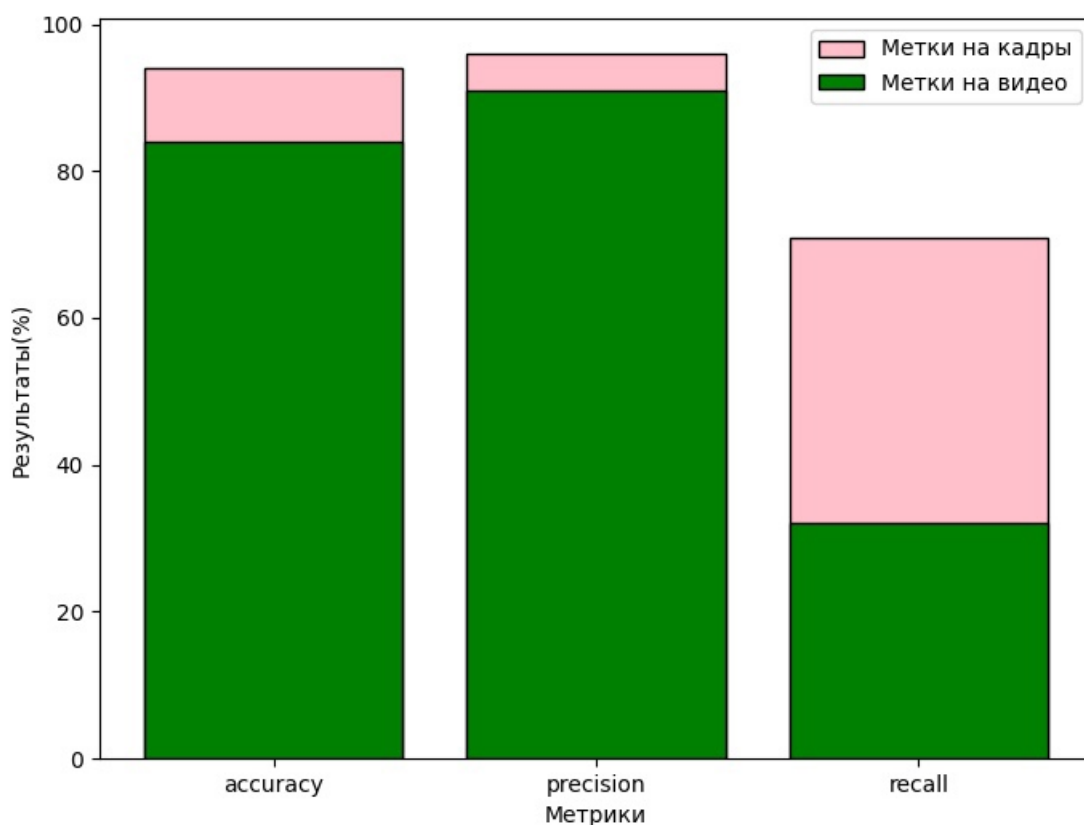


Рисунок 4.1 – Зависимость метрик модели от разметки данных

4.2 Исследование результатов распознавания модели в зависимости от контрастности видео

Для проведения этого исследования тестовая выборка каждого класса на основе экспертной оценки была разделена на видео высокой контрастности и низкой. Далее для каждого видео предсказывался вид действия и считалось количество верно распознанных действий для каждого класса. На основе рассчитанного количества верно распознанных действий рассчитывался процент верного распознавания каждого класса действий в зависимости от контрастности, как отношение верно распознанных к общему количеству видео в соответствующей выборке. Тестовые выборки, разделенные по контрастности содержали видео только высокого качества. На рисунке 4.2 представлены результаты проведенного исследования.

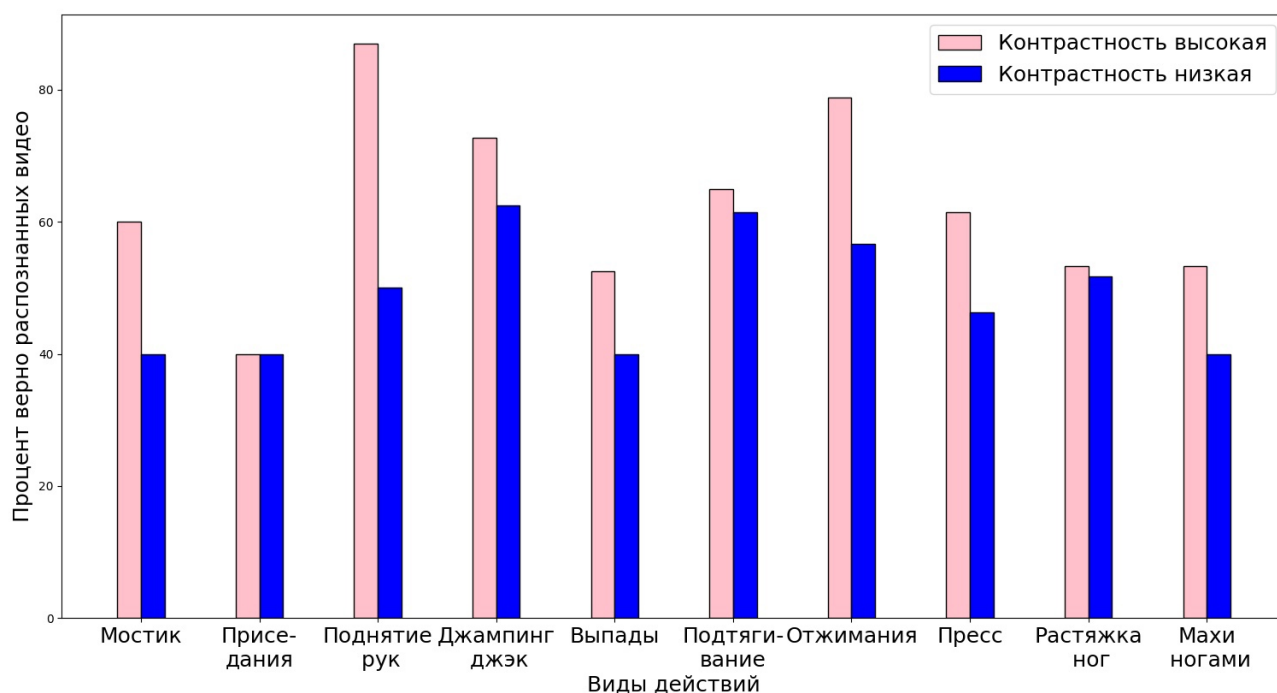


Рисунок 4.2 – Зависимость результатов распознавания от контрастности видео

В результате можно сделать вывод о том, что видео высокой контрастности модель распознает с меньшим количеством ошибок за счет того, что при резких переходах цвета на кадре градиенты интенсивностей больше.

4.3 Исследование результатов распознавания модели в зависимости от качества видео

Для проведения этого исследования тестовая выборка каждого класса была разделена на видео высокого и низкого качества. В связи с ограниченностью набора данных за высокое качество были приняты видео размером больше 256x256 пикселей, а за низкого размером меньше 256x256 пикселей. Выборка состояла из видео высокой контрастности и проводилось только на некоторых видах действий в связи с отсутствием достаточного количества видео низкого качества в исходном наборе данных. Далее для каждого видео предсказывался вид действия и считалось количество верно распознанных действий для каждого класса. На основе рассчитанного количества верно распознанных действий рассчитывался процент верного распознавания каждого класса действий в зависимости от качества, как отношение верно распознанных к общему количеству видео в соответствующей выборке. На рисунке 4.3

представлены результаты проведенного исследования.

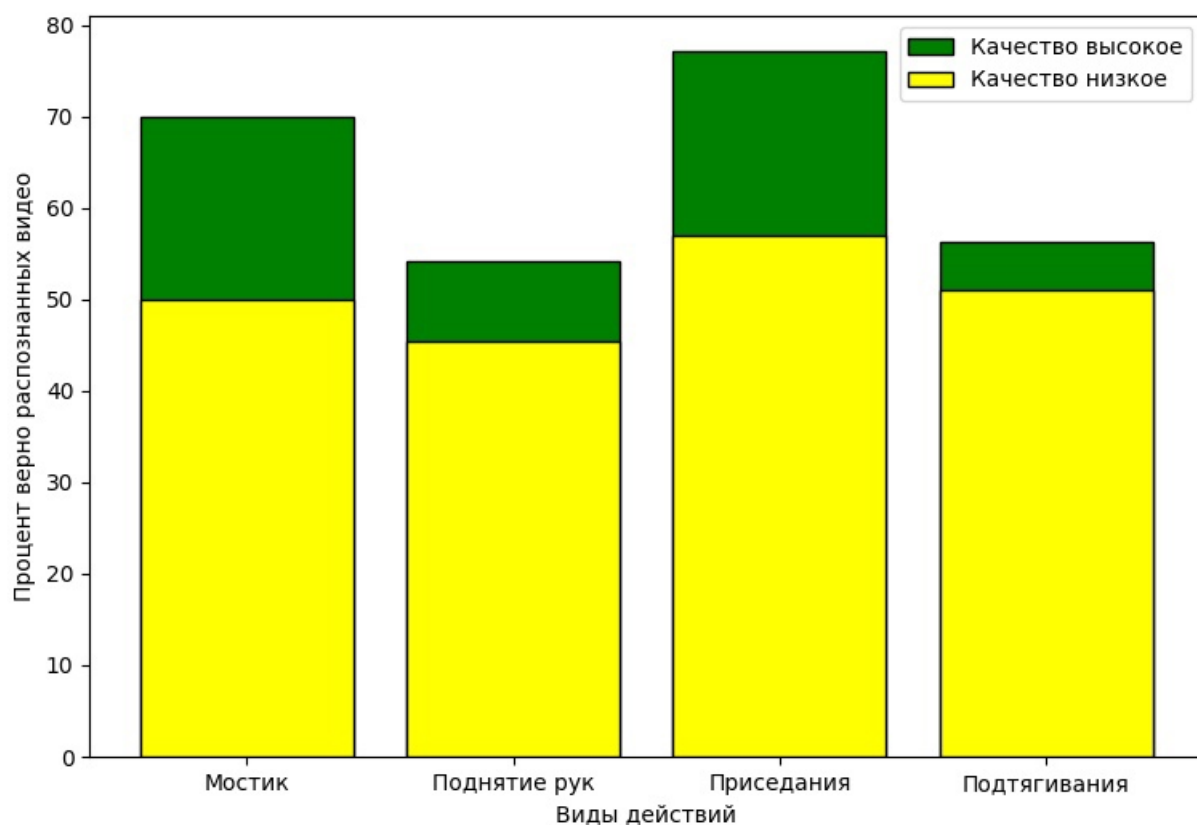


Рисунок 4.3 – Зависимость результатов распознавания от качества видео

4.4 Выводы

В результате проведенных исследований можно сделать вывод о том, что модель имеет более высокие показатели метрик при назначении метки на дескриптор каждого кадра, нежели при назначении на дескриптор видео.

Так же исследования показали, что на видео высокого качества на выборке, состоящей из контрастных видео, модель ошибается меньшее количество раз, нежели на видео низкого качества.

Модель чаще ошибается при распознавании видео с низкой контрастностью, нежели с высокой на выборке из видео высокого качества.

ЗАКЛЮЧЕНИЕ

В результате был разработан метод распознавания спортивных действий человека на видео с использованием локального дескриптора.

В ходе работы были выполнены все поставленные задачи.

- Рассмотрены алгоритмы, с помощью которых можно реализовать метод распознавания спортивных действий человека;
- Проведен обзор существующих решений распознавания действий человека.
- Разработан метод распознавания спортивных действий человека на видео с использованием локального дескриптора.
- Спроектирован и реализован ПО, демонстрирующее работу метода.
- Проведено исследование зависимости метрик разработанного метода от разметки данных.
- Проведено исследование разработанного метода на применимость при работе с видео различной контрастности и качества.

Можно выделить следующие направления дальнейшего развития.

- Распознавание действий нескольких людей.
- Автоматическое обрезание области с человеком из кадра.
- Распознавание нескольких разных видов действий на одном видео.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Тарантова Е. С., Макаров К. В., Орлов А. А.* Выбор признаков для распознавания физической активности человека в рамках телереабилитационной системы // Проблемы информационных технологий. — 2019. — № 2. — С. 49—59.
2. *Уздяев М. Ю.* Распознавание агрессивных действий с использованием нейросетевых архитектур 3D-CNN // Известия Тульского государственного университета. Технические науки. — 2020.
3. *Егорова Ю. А., Захарова И. Г., Гасанов А. Р.* Стохастический метод распознавания действий человека на базе скелетной модели // Информационные технологии и системы. — 2020. — С. 96—102.
4. *Черезов Д. С., Тюкачев Н. А.* Обзор основных методов классификации и кластеризации данных // Вестник воронежского государственного университета. Серия: системный анализ и информационные технологии. — 2009. — С. 25—29.
5. *Батура Т. В.* Методы автоматической классификации // Программные продукты и системы. — 2017. — № 1.
6. *Дмитриев Е. А.* Метод опорных векторов [Электронный ресурс]. — Режим доступа: <https://interactive-plus.ru/e-articles/389/Action389-462813.pdf> (Дата обращения: 10.04.2023).
7. *Виноградова А. А.* Обработка и анализ данных для распознавания человеческих действий [Электронный ресурс]. — Режим доступа: https://math.spbu.ru/SD_AIS/documents/2015-12-341/2015-12-b-05.pdf (Дата обращения: 10.04.2023).
8. *Чернов И. А., Федяев О. И.* Алгоритм построения деревьев решений для системы автоматизированного извлечения знаний [Электронный ресурс]. — Режим доступа: https://masters.donntu.ru/2006/fvti/ichernov/library/my_donetsk06.htm (Дата обращения: 10.04.2023).
9. *Пальмов С. В., Денискова А. О.* Случайный лес: основные особенности // Наука сегодня: теоритические и практические аспекты. — 2017. — С. 51—53.

10. *Приступа А. В., Лапатин И. Л., Замятин А. В.* Применение глобальных дескрипторов формы для идентификации 3D объектов // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. — 2019. — № 48. — С. 57—66.
11. *Финогеев А., Четвергова М.* Разработка и исследование методики распознавания изображений для систем расширенной реальности // Известия Волгоградского государственного технического университета. — 2012. — № 15. — С. 130—136.
12. *Луцив В., Малашин Р.* Объектно-независимый структурный анализ изображений: история и современные подходы // Оптический журнал. — 2014. — № 11. — С. 31—44.
13. *Пименова М. Б.* Распознавание характерных объектов на местности с использованием метода SURF // Политехнический молодежный журнал. — 2019. — № 10. — С. 3.
14. *Панкин А.* Обзор и классификация опубликованных методов детектирования человека // Известия высших учебных заведений. Геодезия и аэрофотосъемка. — 2019. — № 4. — С. 463—475.
15. *Калиберда Ю.* Применение метода Histogram of oriented gradients and support vector machines для реализации алгоритма распознавания рукописных цифр // Системы управления, информационные технологии и математическое моделирование. — 2019. — С. 55—60.
16. *Азаренко К. А., Ньейн К. М., Белов Ю. С.* Обзор методов для распознавания действий человека // E-Scio. — 2019. — 5 (32). — С. 613—624.
17. *Oneata D., Revaud J., Verbeek J.* Spatio-temporal object detection proposals // Computer Vision – ECCV 2014. — 2014. — С. 737—752.
18. *Soomro K., Idrees H., Shah M.* Action localization in videos through Context Walk // 2015 IEEE International Conference on Computer Vision (ICCV). — 2017.
19. *Soomro K., Shah M.* Unsupervised Action Discovery and Localization in Videos // 2017 IEEE International Conference on Computer Vision (ICCV). — 2017. — С. 696—705.

20. *Зеленский А. А., Письменскова М. М.* Метод распознавания действий человека на сложноструктурированных изображениях и фоне в виде стохастических текстур // Вестник МГТУ Станкин. — 2018. — № 3. — С. 116—120.
21. *Tahir S., Dogar A.* Stochastic Recognition of Human Physical Activities via Augmented Feature Descriptors and Random Forest Model // Sensors. — 2022. — С. 6632.
22. *Morariu V. I., Davis L. S.* Multi-agent event recognition in structured scenarios // CVPR. — 2011. — С. 3289—3296.
23. *Jia H., Chen S.* Integrated Data and Knowledge Driven Methodology for Human Activity Recognition // Information Sciences. — 2020.
24. *Sharma C. M., Kushwaha A.* Human Action Recognition using Rule based Fuzzy Motion Feature Templates // International Journal of Innovative Technology and Exploring Engineering. — 2019. — С. 4695—4700.
25. *Park J., Jin Y., Cho S.* Advanced Machine Learning for Gesture Learning and Recognition Based on Intelligent Big Data of Heterogeneous Sensors // Symmetry. — 2019. — С. 929.
26. *Ikizler N., Duygulu P.* Human Action Recognition Using Distribution of Oriented Rectangular Patches // Human Motion – Understanding, Modeling, Capture and Animation. — 2007. — С. 271—284.
27. *Wang Y., Mori G.* Hidden Part Models for Human Action Recognition: Probabilistic vs. Max-Margin. // IEEE transactions on pattern analysis and machine intelligence. — 2010.
28. *Jung H. Y., Lee S., Heo Y. S.* Forest Walk Methods for Localizing Body Joints from Single Depth Image // PLOS ONE. — 2015. — № 9. — С. 1—20.
29. О важности датасета и о том, как сделать его лучше. Наш опыт [Электронный ресурс]. — Режим доступа: <https://vc.ru/ml/464790-o-vazhnosti-dataset-a-i-o-tom-kak-sdelat-ego-luchshe-nash-opyt> (Дата обращения: 07.04.2023).

30. Отберем то, что нужно Data Mining: как сформировать датасет для машинного обучения [Электронный ресурс]. — Режим доступа: <https://www.bigdataschool.ru/blog/dataset-data-preparation.html> (Дата обращения: 07.04.2023).
31. Scikit-learn [Электронный ресурс]. — Режим доступа: <https://scikit-learn.org/stable/> (Дата обращения: 17.04.2023).
32. OpenCV [Электронный ресурс]. — Режим доступа: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html (Дата обращения: 17.04.2023).
33. Flask [Электронный ресурс]. — Режим доступа: <https://flask-docs.readthedocs.io/en/latest/quickstart/> (Дата обращения: 17.04.2023).
34. Scikit-image [Электронный ресурс]. — Режим доступа: <https://scikit-image.org/> (Дата обращения: 17.04.2023).

ПРИЛОЖЕНИЕ А