

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования

**«Южно-Уральский государственный университет
(национальный исследовательский университет)»**

**Высшая школа электроники и компьютерных наук
Кафедра системного программирования**

РАБОТА ПРОВЕРЕНА

Рецензент
ИТ директор ООО «Програмлаб»

_____ А.Г. Криушичев

«___» _____ 2024 г.

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой, д.ф.-м.н.,
профессор

_____ Л.Б. Соколинский

«___» _____ 2024 г.

**Разработка приложения для сравнения нейросетевых моделей
отслеживания взгляда и фиксации объектов**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
ЮУрГУ – 09.04.04.2024.308-1472.ВКР**

Научный руководитель,
профессор кафедры СП, д.г.н.,
к.ф.-м.н.

_____ С.М. Абдуллаев

Автор работы,
студент группы КЭ-228

_____ В.С. Кадомцев

Ученый секретарь
(нормоконтролер)

_____ И.Д. Володченко

«___» _____ 2024 г.

Челябинск, 2024 г.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
**«Южно-Уральский государственный университет
(национальный исследовательский университет)»**
Высшая школа электроники и компьютерных наук
Кафедра системного программирования

УТВЕРЖДАЮ

Зав. кафедрой СП

_____ Л.Б. Соколинский

29.01.2024 г.

ЗАДАНИЕ

на выполнение выпускной квалификационной работы магистранта

студенту группы КЭ-228

Кадомцеву Владлену Сергеевичу,

обучающемуся по направлению

09.04.04 «Программная инженерия»

(магистерская программа «Искусственный интеллект и инженерия данных»)

1. Тема работы (утверждена приказом ректора от 22.04.2024 г. № 764-13/12)

Разработка приложения для сравнения нейросетевых моделей отслеживания взгляда и фиксации объектов.

2. Срок сдачи студентом законченной работы: 20.05.2024 г.

3. Исходные данные к работе

3.1. Dalveren G.G.M., Cagiltay N.E. Evaluation of Ten Open-Source Eye-Movement Classification Algorithms in Simulated Surgical Scenarios. // IEEE Access, vol. 7, 2019. –161794–161804 pp.

3.2. Komogortsev O.V., Gobert D.V., Jayarathna S., Koh D. H., Gowda S. M. Standardization of Automated Analyses of Oculomotor Fixation and Saccadic Behaviors. // IEEE Transactions on Biomedical Engineering, 2010. – 2635–2645 pp.

3.3. Holmqvist K. Eye tracking: a comprehensive guide to methods and measures. // Oxford University Press, 2011. –101–103 pp.

4. Перечень подлежащих разработке вопросов

- 4.1. Выполнить анализ предметной области.
- 4.2. Исследовать нейросетевые модели отслеживания взгляда.
- 4.3. Создать приложение для работы с нейросетевой моделью отслеживания взгляда.

5. Дата выдачи задания: 29.01.2024 г.

Научный руководитель,
профессор кафедры СП, д.г.н., к.ф.-м.н.

С.М. Абдуллаев

Задание принял к исполнению

В.С. Кадомцев

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ	7
1.1. Описание предметной области	7
1.2. Степень разработанности темы	8
1.3. Состояние вопроса исследования.....	11
1.4. Современные подходы	17
2. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ.....	20
2.1. Описание набора данных	20
2.2. Определение порогового значения дисперсии (I-DT)	21
2.3. Определение порогового значения скорости (I-VT)	22
2.4. Недостатки алгоритмов I-DT и I-VT.....	24
2.5. Случайный лес (RF).....	28
2.6. Сверточная нейронная сеть (CNN)	30
2.7. Анализ результатов.....	32
3. РАЗРАБОТКА ПРИЛОЖЕНИЯ	37
3.1. Выбор средств реализации.....	37
3.2. Общее описание архитектуры системы.....	38
3.3. Представление пользовательского интерфейса	39
ЗАКЛЮЧЕНИЕ	42
ЛИТЕРАТУРА.....	43
ПРИЛОЖЕНИЯ.....	46
Приложение А. Спецификация вариантов использования.....	46
Приложение Б. Сравнительная таблица алгоритмов	48

ВВЕДЕНИЕ

Актуальность

Отслеживание движения глаз используется в разнообразных сферах исследований, таких как когнитивная наука, психология, неврология, инженерия, медицина и маркетинг. Еще одним примером применения является взаимодействие человека и компьютера, что полезно для людей с ограниченной подвижностью. Отслеживание движения глаз также используется для управления автомобилем. Это делает данную технологию крайне междисциплинарной и используемой в различных сферах. Это также отражается в том, как с течением времени разрабатывались аппаратное и программное обеспечение для трекинга глаз.

Процедура обнаружения событий в трекинге взгляда связана с рядом проблем. Одна из них заключается в том, что записанный сигнал может содержать множество различных типов шума и помех, которые возникают из-за индивидуальных различий между пользователями и оборудованием для трекинга взгляда. Эта изменчивость между индивидами и качеством сигнала может создавать сложные для анализа сигналы. Следовательно, цель состоит в разработке надежных алгоритмов обнаружения событий (АОС), которые будут достаточно гибкими для работы с сигналами различных типов событий и помех, которые могут взаимодействовать с различными типами оборудования для трекинга и различными людьми. Другая проблема в обнаружении событий движений глаз в сигналах трекинга – это оценка и сравнение различных АОС.

Постановка задачи

Целью выпускной квалификационной работы является разработка приложения для сравнения нейросетевых моделей отслеживания взгляда и фиксации объектов. Для достижения поставленной, цели необходимо решить следующие задачи:

- 1) выполнить анализ предметной области;
- 2) исследовать нейросетевые модели отслеживания взгляда;

3) создать приложение для работы с нейросетевой моделью отслеживания взгляда.

Структура и содержание работы

Работа состоит из введения, трех глав, заключения и списка литературы. Объем работы составляет 48 страниц, объем списка литературы – 24 источника.

В первой главе проводится анализ исследуемой области. Даны основные определения по исследуемой области. Проведен анализ существующих решений.

Вторая глава посвящена получению экспериментальных результатов. Проведено тестирование и сравнение различных АОС. Даны теоретические и экспериментальные обоснования полученным результатам. По результатам проведенных экспериментов был выбран наилучший для наших задач АОС.

В третьей главе описаны средства реализации для создания приложения. Описаны преимущества и недостатки выбранных решений. Решены возникшие технические проблемы. Спроектирована архитектура, разработан макет приложения исходя из требуемых задач. Создано приложение, соответствующее функциональным требованиям.

В приложении А содержатся таблицы, описывающие основные варианты использования (ВИ) разрабатываемого приложения.

В приложении Б содержится сравнительная таблица преимуществ и недостатков исследуемых алгоритмов.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Данный раздел содержит описание основных АОС и подходов, применяемых при обработке данных отслеживания взгляда. Даны определения основным событиям и классам, а также их особенности.

1.1. Описание предметной области

Отслеживание глаз – это технология, предназначенная для определения направления взгляда человека. Обнаружение событий – это процесс обнаружения и классификации движений глаз, которые подразделяются на несколько типов. На данный момент обнаружение событий происходит главным образом с помощью АОС, применяемого к необработанным данным отслеживания глаз, записанным на носитель. Однако, из-за отсутствия стандартного метода оценки, оценка и сравнение разных АОС на основе данных отслеживания глаз представляет собой сложную задачу.

Сегодня существует множество различных алгоритмов, оценить их все сложно по нескольким причинам. Во-первых, некоторые из них являются коммерческими продуктами с закрытым исходным кодом, предоставленным производителем системы айтрекинга. Их невозможно воссоздать полностью. Несмотря на возможность получения исходных данных и распознанных событий движения глаз из системы с закрытым кодом, исходные данные сохраняются в двоичной форме, уникальной для данной коммерческой системы. Классификаторы коммерческих событий могут обрабатывать только свои собственные двоичные данные. Следовательно, нам нужно использовать алгоритмы, позволяющие вводить данные в стандартном формате. Во-вторых, не все алгоритмы доступны в виде готовых реализаций. Оценка таких алгоритмов требует их адаптации или доработки, что может увеличить количество ошибок и искажений. Таким образом, алгоритм должен быть представлен в виде официальной реализации. В-третьих, не все из них эффективно работают на реальных данных без дополнительной настройки.

Таким образом объектом исследования будут являться различные алгоритмы, будет приведена оценка их эффективности.

Исходя из вышеописанного, объектом диссертационного исследования является оценка эффективности только полностью воспроизводимых алгоритмов обнаружения событий возникающих при движении глаз.

1.2. Степень разработанности темы

Как уже упоминалось, необработанные данные о движениях глаз обычно подразделяются на события. Данные трекинга взгляда состоят не только из разных типов событий движения глаз, но также из шума из различных источников и моргания. Алгоритм обнаружения событий, следовательно, должен учитывать эти проблемы. Основные типы событий, которые используются, обсуждаются далее.

Фиксация

Фиксация – это движение, при котором глаз более или менее неподвижно фокусируется на объекте. Цель фиксирующего движения – стабилизировать объект в ямке, чтобы обеспечить четкое зрение. События фиксации включают в себя три разных типа незначительных движений: тремор, медленный дрейф и микросаккады.

Тремор – это небольшое волнообразное движение глаза с частотой менее 150 Гц и амплитудой около $0,01^\circ$. Это естественный двигательный фон окуломоторной активности, который неподвластен произвольному контролю.

Медленный дрейф – это медленное, непрерывное перемещение глаза, которое прерывается микроскачками. На дрейф при фиксации приходится 97% времени. Предполагается, что дрейф создает наиболее благоприятные условия для окуломоторного восприятия и обработки оптической информации.

Микросаккады – это самое быстрое из фиксирующих движений глаза, продолжительность которых составляет около 25 мс. Функция микросаккад – быстрое возвращение глаза в исходное положение.

Саккада

Саккады – это быстрое перемещение глаз от одной точки фиксации к другой. Типичная саккада имеет продолжительность от 30 до 80 мс и скорость от 30 до 500 градусов в секунду. Существует взаимосвязь между продолжительностью, амплитудой и скоростью саккады. Эта взаимосвязь предполагает, что более крупные саккады имеют большую скорость и длятся дольше, чем более короткие. Время от появления стимула до начала движения глаз (так называемая саккадическая латентность) составляет около 200 мс. Это включает в себя время, необходимое для центральной нервной системы на определение, следует ли начинать саккаду или нет, и в этом случае вычисляет расстояние, на которое должен переместиться глаз, и передает нервные импульсы мышцам, которые помогают двигать глазами. Правильное обнаружение саккад имеет важное значение, поскольку считается, что человеческий мозг «не видит» изображение в период саккады. Это явление называется саккадическим подавлением.

Плавное преследование (ПП)

Плавное преследующее движение выполняется, когда глаза следят за медленно движущимся объектом. Это может произойти только при наличии движущейся цели, за которой нужно следить. Латентность плавного следования составляет около 100 мс, и она немного меньше, чем латентность саккадического движения. Это относится ко времени, которое требуется глазам, чтобы начать двигаться, с момента определения положения целевого объекта.

Событие плавного движения при преследовании, как правило, можно разделить на два этапа: этапы с разомкнутой цепью и этапы с замкнутой цепью. Стадия инициирования плавного преследования – это заранее запрограммированный этап разомкнутой цепи, на котором глаз ускоряется

для того, чтобы догнать движущийся объект. Стадия замкнутой цепи начинается, когда глаз захватывает цель и следит за ней со скоростью, подобной скорости объекта-цели.

Чтобы иметь возможность следовать за движущимся объектом на стадии замкнутой цепи, скорость движущегося объекта оценивается и сравнивается со скоростью глаза. Если скорости движущегося объекта и глаз различны (например, глаз отстает от движущегося объекта), происходит догоняющее движение саккадой, выполняемое для того, чтобы опять догнать объект.

Постсаккадические колебания (ПСК)

Быстрые колебательные движения, которые могут возникнуть сразу после саккады, называются постсаккадическими колебаниями (ПСК). Они могут быть описаны как колебательные или нестабильные движения, возникающие в конце саккады.

Постсаккадические колебания характеризуются небольшим колебательным движением, которое вызывает фиксацию после саккады. Причина ПСК до сих пор остается неясной. Некоторые исследователи считают, что они вызваны записывающим устройством, другие полагают, что глаз сам по себе естественным образом колеблется после саккады.

ПСК – это тип движения глаз, относительно которого обычно существуют наиболее значительные разногласия между экспертами, проводящими ручную оценку. Тем не менее, это события, происходящие во время записи движений глаз и способные повлиять на характеристики фиксационных и саккадических событий.

Обычно ПСК представляют собой очень кратковременные события продолжительностью около 10–40 миллисекунд с амплитудой 0,5–2 и скоростями 20–140 градусов в секунду.

Глиссады

Другой, в значительной степени неизученной, причиной различий в результатах определения событий является поведение глаз в конце многих

саккад. Оно указывает на то, что глаза иногда не фиксируются непосредственно на объекте, пропуская его, и затем нуждаются в дополнительной корректирующей короткой саккаде. Такой тип движения называется глиссадой.

Глиссады возникают примерно после 50% саккад, следовательно, они оказывают значительное влияние на точность измерения смещения саккады и момент начала последующей фиксации. Поэтому часто глиссадное движение рассматривают как отдельный тип движения глаз. Это движение известно также как быстрое постсаккадное движение (динамическое превышение скорости) или более медленное постсаккадное (глиссадное) движение. Исследователи замечали, что глиссадные движения редко возникают одновременно в обоих глазах. Хотя это часто упоминается в литературе, алгоритмы определения событий лишь иногда учитывают это явно. Таким образом, глиссадные движения обрабатываются бессистемно и по-разному в различных алгоритмах и даже в рамках одного и того же алгоритма: одно глиссадное может быть назначено саккаде, в то время как следующее объединяется с фиксацией.

1.3. Состояние вопроса исследования

В различных областях обработки сигналов сравнительная оценка эффективности АОС производится при подаче в систему обнаружения сигналов, имитирующих реальные события. Этим достигается сравнимость АОС и возможность калибровки системы. Тем не менее, поскольку задача состоит в создании имитируемых сигналов отслеживания глаз, которые могут уловить помехи и изменения в необработанных сигналах до такой степени, что они являются полезными и достоверными для оценивания эффективности работы. Кроме того, из-за отсутствия стандартного метода оценки различных алгоритмов обнаружения событий, нелегко сравнивать характеристики определения различных алгоритмов из разных источников.

При исследовании движения глаз цель обнаружения событий – надежное извлечение таких событий, как фиксации и саккады, из необработанного потока выборок данных от устройства отслеживания глаз, на основе набора основных правил и критериев, подходящих для записанного сигнала. До недавнего времени исследователи, которые решались записывать движения глаз, должны были выполнять трудоемкую ручную обработку событий. Например, как заметил Монтегю: «На это нередко приходилось тратить дни на анализ данных, сбор которых занимал всего несколько минут».

Было сделано много работы по разработке алгоритмов для обнаружения событий, связанных с движениями глаз. Эффективность и адаптируемость этих алгоритмов зависит от различных факторов, в том числе от типа стимула (статического или динамического), качества данных (которые могут быть зашумлены), устройства слежения (частоты дискретизации, монокулярного или бинокулярного, стационарного или мобильного с жесткими или гибкими окулярами).

Эти различия усложняют прямое сравнение разных методов и исследований. Различные методы обнаруживают различные типы событий. Некоторые методы определяют только фиксацию или саккаду; некоторые определяют только фиксацию; некоторые обнаруживают фиксацию, саккаду и постсаккадические колебания.

Из-за различий в типах событий, которые определяют алгоритмы, сравнение алгоритмов, которые обнаруживают однотипные, двоичные события, становится затруднительным. Это делает сравнение эффективности алгоритмов сложной задачей.

Процесс классификации движений глаз в отдельные события, на базовом уровне, обусловлен необходимостью выделить различные интервалы в потоке данных, которые сильно коррелируют с определенными глазомоторными или когнитивными особенностями.

Например, зрение, как правило, сильно ограничено во время саккады, что, наряду с общим желанием сократить объем данных, по всей видимости, послужило мотивацией для ранних детекторов фиксаций и саккад.

Аналогичным образом, плавное преследование запускается воспринимаемым движением и указывает на визуальное восприятие. Однако эти движения могут включать в себя множество интересующих областей (АОИ) и нарушать стандартные меры АОИ и фиксаций, если движение рассматривать как фиксацию.

Моргание – это еще один тип движения, хоть и не глазного яблока, но оно обычно обнаруживается, чтобы его исключить из потока данных и предотвратить его влияние на дальнейшую классификацию движений глаз или использовать его как тренировочный пример для удаления связанных с глазами артефактов в электроокулографических данных.

Моргание также связано с ограниченной видимостью во время закрытия век и после их повторного открытия. Это особенно важно, так как исходные данные о закрытии и открытии век могут отображаться в некоторых системах слежения за взглядом аналогично саккадам вниз и вверх [1].

Кроме того, считается, что колебания хрусталика во время саккад вызывают деформацию радужки (и, соответственно, зрачка), которая продолжается в течение некоторого времени после саккады. Это вызывает так называемые постсаккадические колебания в сигнале отслеживания взгляда [2]. ПСК обнаруживаются реже, но приобретают все большее значение, поскольку новые исследования и технологии слежения за глазом повышают временное и пространственное разрешение.

Соответствующие выборки данных для ПСК обычно не группируются ни по фиксациям, ни по саккадам [3]. Так как все еще существуют неопределенности относительно точного характера зрительного восприятия во время этих колебаний, то, как это событие классифицируется, поэтому является критически важным для исследователя, который использует алго-

ритмы классификации движений глаза для выбора моментов максимального зрительного восприятия.

Одним из наиболее распространенных алгоритмов для обнаружения фиксаций является IDT (Identification by Dispersion Threshold). IDT работает с данными по осям X и Y и двумя фиксированными порогами: порог максимальной дисперсии для фиксации и порог минимальной длительности фиксации. Для того чтобы точка считалась фиксацией, она должна оставаться в пределах области, ограниченной порогом дисперсии, в течение времени, достаточного для выполнения порога длительности. Если эти критерии выполнены, точка помечается как фиксация. Особенностью этой конкретной реализации является то, что она входит в состав пакета, который также включает объединение коротких близко расположенных фиксаций и связан с детектором саккад.

Реальные данные часто бывают зашумленными и могут страдать от потери данных. Разработанный для преодоления этой проблемы алгоритм должен быть многообещающим.

Фильтр Калмана – это рекурсивный фильтр, который обеспечивает оптимальное, т. е. минимизирующее ошибку, сочетание текущего измерения и предсказанного измерения с учетом предыдущих данных. Строго говоря, в алгоритме идентификации фильтр Калмана (ЕКФ (Extended Kalman filter)) не классифицирует сигналы трекера взгляда на события. Однако в этой конкретной реализации это делается с использованием χ^2 -теста. Этот тест классифицирует все точки в пределах заданного окна как относящиеся к фиксации, если значение χ^2 меньше установленного порога и соответствует порогу минимальной длительности, или как относящиеся к саккаде, если это значение больше порогового.

Эта конкретная реализация, так же как и другие реализации, выполненные Комогорцевым и Каном [4], используют ту же постобработку, которую применяют и другие алгоритмы. Таким образом, они группируют близкие фиксации.

Другой подход к обнаружению событий – это MST (Minimum Spanning Tree) идентификация. Этот алгоритм строит «дерево» данных, который разветвляется на точки данных. Алгоритм пытается захватить все данные с минимальным количеством ветвей, чтобы точки из разных кластеров захватывались ветвями, ведущими к двум отдельным узлам, а не принуждали к обширному разветвлению к одному узлу более низкого уровня. Установив определенные пороги для точек на краях кластера, саккады идентифицируются и исключаются из процесса обнаружения фиксаций. Реализации, о которых идет речь, созданы Комогорцевым и Гобертом [5].

Учитывая, что наиболее распространенным типом различий между событиями является разница между неподвижной фиксацией и быстрой саккадой, интуитивно понятно, что алгоритм, который осуществляет классификацию на основе вероятностного перехода между двумя состояниями. Такой алгоритм, основанный на скрытой Марковской модели (НММ), описан в работе Комогорцева и Гоберта [5] и основан на скорости алгоритма. Затем он обернут еще двумя дополнительными алгоритмами: первый переклассифицирует фиксации и саккады на основе вероятностных параметров, таких как начальное состояние и вероятность перехода между состояниями, а второй обновляет эти параметры.

Очень распространенным основанием для разделения образцов, относящихся к фиксациям, и образцов, относящихся к саккадам, является определение скорости этих образцов. Идентификация на основе порога скорости (VT) – это простой алгоритм, который работает следующим образом [6]. Он использует фиксированную пороговую скорость для идентификации фиксаций и саккад: фиксации – это участки образцов с точечной скоростью ниже установленного порогового значения, а саккады – это участки образца со скоростями, превышающими этот порог. Этот основной критерий скорости часто является основой других алгоритмов. Данная конкретная реализация была взята у Комогорцева, Гоберта [5].

Алгоритм, представленный Нистремом и Холмквистом [7], (далее NH), стал первым, который явным образом идентифицировал постсаккадические колебания («глиссады») наряду с фиксациями и саккадами. Этот алгоритм адаптивен в том смысле, что регулирует порог скорости в зависимости от уровня шума в данных.

Обнаружение маленьких саккад из-за шума – сложная задача, и здесь поможет тот факт, что глаза обычно смотрят на один и тот же предмет. Если левый глаз движется к определенному предмету, то и правому глазу нужно это сделать. Это упрощает определение того, вызвано ли повышение скорости реальным движением или только шумом, потому что оба глаза должны показать это повышение на кривой скорости одновременно. Эту идею использовал человек с биноклем – пороговый (бинарный) алгоритм, который разработали Van der Lans, Wedel и Pieters [8]. Как и некоторые другие алгоритмы, этот адаптивно устанавливает пороги.

Окончательный алгоритм, который будет рассмотрен – недавняя разработка Ларссон, Нюстрема и Стрида [9]. Этот алгоритм – второй, способный обнаруживать постсаккадические колебания; но он также определяет саккады. Адаптивный алгоритм нов в том, что он был создан для определения саккад и постсаккадических колебаний, даже при присутствии плавных слежений глаза. Плавное преследование генерирует скорости, которые непоследовательно обрабатывались алгоритмами со стандартными порогами скоростей. Как правило, не существует четкого разделения этих движений на фиксации или саккады, а это больше зависит от конкретного плавного преследования и текущих порогов алгоритма.

Исходя из вышеописанного, можно говорить о том, что исследования АОС имеют долгую историю, но в последние годы интерес к АОС возобновился в связи с применением к проблеме более совершенных методов машинного обучения и появлению стандартизированных баз данных [10, 11, 12].

1.4. Современные подходы

Отсутствие единого стандартного алгоритма для всех систем и множество алгоритмов для решения одной и той же проблемы говорит о том, что классификация движений глаз – это нетривиальная задача, и оценка эффективности различных алгоритмов может быть непростой задачей. Важно учесть, что выбор лучшего алгоритма требует разработки соответствующего метода оценки. Так как это исследование не было первой попыткой оценки эффективности алгоритма, было бы полезно рассмотреть преимущества и недостатки уже существующих методов.

Самым базовым методом оценки для обнаружения событий в ранних исследованиях движений глаз был простой ручной анализ исследователем [10]. В настоящее время цель состоит в том, чтобы определить конкретные события (саккады), а не оценить алгоритмы. Тем не менее современные оценки используют этот ручной метод вместе с автоматизированными способами оценки классификаций. К примеру, образцы необработанных данных объединяются в фиксацию, а алгоритм обнаружения фиксаций должен обнаружить все образцы, относящиеся к этим группам, и исключить те, что находятся вне групп. К сожалению, эти ручные аспекты оценки часто упоминаются кратко, например, Виг, Дорр и Барт [11] корректировали параметры вручную, пока они не выглядели приемлемо.

Однако такая ручная оценка может использоваться более строго, когда человеческие оценщики систематически кодируют одни и те же данные с использованием одних и тех же категорий и алгоритмов. Затем можно напрямую сравнивать эффективность алгоритмов с экспертами-людьми. Это тоже часто делается при оценке новых алгоритмов. Сравнение человеческого и алгоритма, однако, зачастую предполагает, что люди действуют совершенно рационально, и поэтому любое отклонение от идеального согласия связано с ошибками в алгоритме. Таким образом, вопрос, тесно связанный с таким подходом, заключается в том, насколько надежны человеческие эксперты. Во многих сферах исследований это анализировалось с

использованием специальных мер для оценки надежности, таких как коэффициент Коэна [12], отношение несогласованных выборок или вычисленные корреляций между экспертами.

Другой подход заключается в предположении об оптимальном или рациональном соотношении между стимулами и наблюдаемым поведением индивида. Использовался подход, в котором участнику эксперимента давалось указание смотреть на анимацию одиночной мишени, которая совершала серию прыжков. Если известно количество прыжков, известное положение, известная амплитуда и так далее, можно рассчитать идеальное поведение взгляда. Данные о взглядах, проанализированные с помощью алгоритмов, затем сравнивались с этим идеальным взглядом, и чем больше сходство, тем лучше алгоритм. Несмотря на то что такой подход может иметь некоторую интуитивную привлекательность, в нем также есть несколько потенциальных проблем [13].

Прежде всего, участники эксперимента не всегда действуют абсолютно рационально и не могут полностью контролировать свой взгляд. На самом деле, часто они недооценивают предполагаемую цель и совершают корректирующие саккады. Участник может отвлекаться и забывать о цели или пытаться предугадать будущую цель, подвергая тем самым риску даже идеальный алгоритм для классификации движений глаз [14, 15].

Наконец, есть два аспекта данных, которые могут быть оценены. Первый – это событие, которое присваивается каждой выборке данных, то есть метка, независимо от фактических значений этих выборок. Второй – это фактические значения выборки, которые определяют свойства события, частью которого они являются. В ходе оценки, описанной в данной статье, мы решили сосредоточиться на маркировке, то есть классификации выборок как относящихся к конкретным глазодвижительным феноменам [16].

Одна из причин для принятия такого решения состоит в том, что прямая задача классификации довольно сложна: при приписывании пра-

вильной метки требуется либо попадание, либо промах. Вторая причина в том, что такая классификация выборок «бесплатно» предоставляет три параметра события: количество событий, продолжительность (то есть число последовательных выборок) и дисперсию их продолжительности [17, 18]. Таким образом, решение одной задачи оценки дает три возможных меры качества для оценки.

Кроме того, саккады в среднем подчиняются основному порядку, то есть между длительностью и амплитудой саккад существует систематическая связь. Мы получаем косвенную меру амплитуды саккад [19].

Можно отметить, что современные технологии не позволяют точно предсказывать когнитивные процессы, основываясь на паттернах движений глаз. По этой причине в исследованиях с отслеживанием движений глаз иногда используются дополнительные вербальные данные. Кроме того, существует множество доказательств того, что лица и тексты в первую очередь привлекают зрительное внимание. Визуальные паттерны сканирования обычно полезны для анализа когнитивных намерений и значимости. Прочие биологические факторы (возраст, пол, раса) также влияют на данные сканирования [20, 21].

Вывод по первой главе

В результате анализа предметной области были даны определения основным событиям, которые будут классифицированы при анализе данных отслеживания взгляда. Рассмотрены основные сложности при классификации таких событий, применение и степень использования АОС для решения данной задачи.

2. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

Данный раздел содержит информацию о используемом наборе данных. Содержит теоретические сведения о работе алгоритмов обнаружения событий и анализ результатов их работы.

2.1. Описание набора данных

Для проверки эффективности АОС будет использован общедоступный набор данных, записанный с помощью высокоскоростного глазного трекера 1250, фирмы «SensoMotoric Instruments» (Teltow, Германия) на частоте 500 Гц.

При создании данного датасета испытуемым предъявлялись статические изображения, тексты, видеоклипы и простые движущиеся точечные раздражители. Данные были помечены вручную двумя братьями Маркусом Нюстремом и Ричардом Андерсоном. Они были разделены на следующие категории: фиксации, саккады, пост-саккадические колебания, плавное следование, моргания и неопределенные. Для данного исследования использовались данные о просмотре изображений, помеченные фиксациями, саккадами и постсаккадическими колебаниями. Данные датасета отражены на рисунке 1 и представляют собой «сырые» данные. На рисунке 2 данные разделены на фиксации.

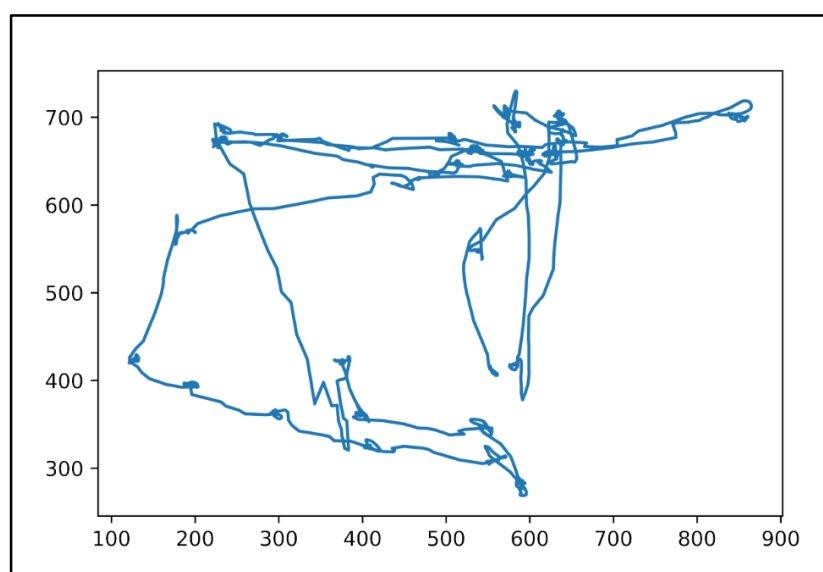


Рисунок 1 – Визуализация данных датасета

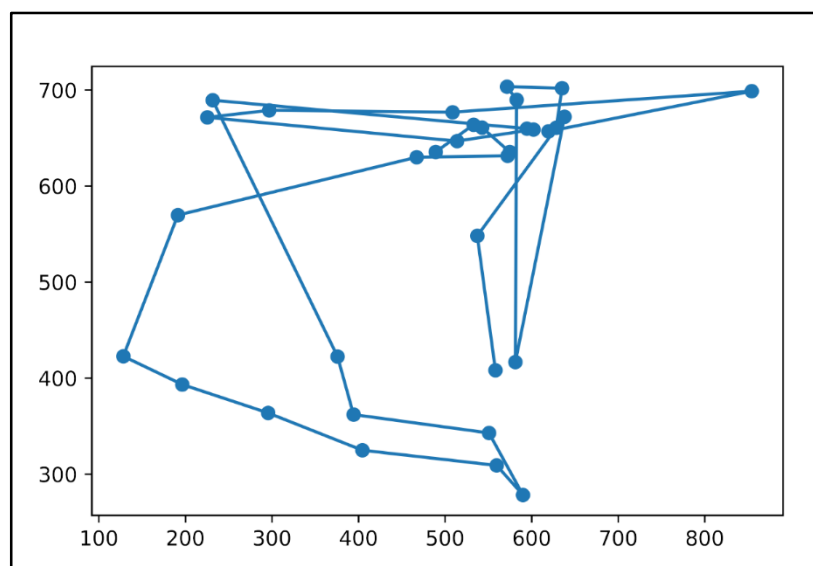


Рисунок 2 – Размеченные данные, на которых отражены фиксации

2.2. Определение порогового значения дисперсии (I-DT)

Будет рассмотрен алгоритм определения пороговых значений дисперсии (dispersion threshold). Особенностью данного алгоритма является то, что он демонстрирует низкую эффективность при обнаружении фиксаций и саккад, когда сигнал сильно зашумлен. В связи с этим выбор пороговых значений является наиболее важным и будет напрямую влиять на эффективность классификации.

Для изучения влияния порогового значения на эффективность классификации были произведены запуски алгоритма с разным пороговым значением. Входные выборки были преобразованы в последовательности, содержащих точку и четыре точки вокруг классифицируемой точки. После чего полученные результаты разметки были сопоставлены со значениями ручной классификации.

Полученные результаты отражены на рисунке 3. По ним можно говорить о том, что увеличение порогового значения дисперсии увеличивает точность определения фиксаций, но уменьшает точность определения саккады. При этом уменьшение порогового значения уменьшает точность обнаружения фиксации и увеличивает обнаружение саккады.

При значении пороговой дисперсии, равном 7-ми, точность обнаружения фиксации составляет 99%, при этом точность определения саккады составляет 82%. При пороговом значении, равном 1-му, уже точность определения саккады равна 99%, но точность обнаружения фиксации всего 39%.

Было выбрано оптимальное значение порогового значения, равного 3,5. При данном значении можно получить точность определения фиксации в 95% и точность обнаружения саккады в 93%.

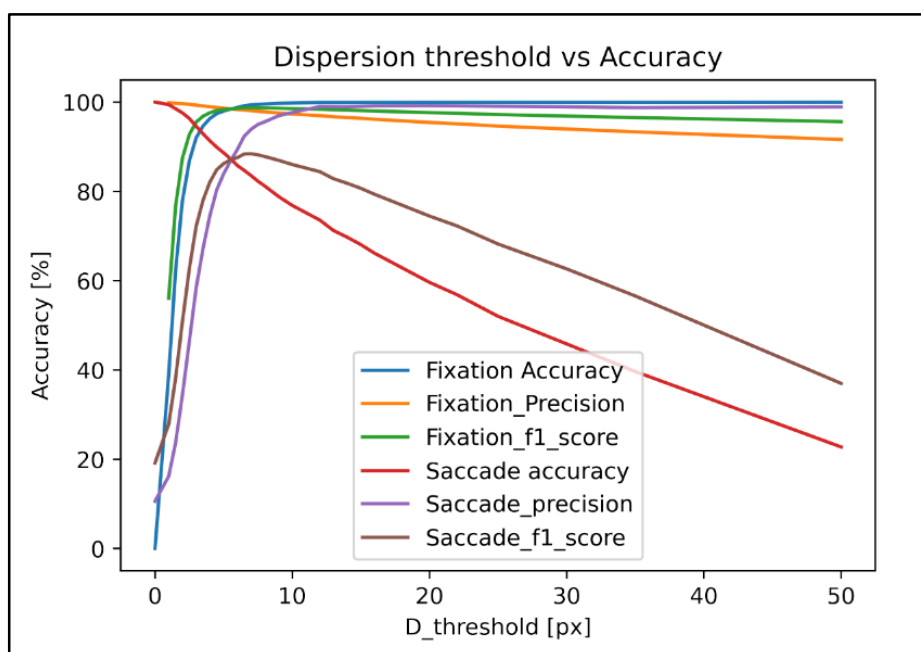


Рисунок 3 – Зависимость точности определения фиксации и саккад для алгоритма I-DT от порогового значения дисперсии

2.3. Определение порогового значения скорости (I-VT)

Алгоритм определения порогового значения скорости (velocity threshold algorithm) представляет собой еще один алгоритм и основу для автоматизированного и объективного стандартного алгоритма обнаружения событий. Этот подход использовался во многих исследованиях. Он основан на том факте, что саккадические движения глаз характеризуются более высокими значениями скорости по сравнению с движениями при фиксации. Профили скоростей движений глаз демонстрируют два основных

распределения скоростей: низкие скорости для фиксации и высокие скорости для саккад. Метод I-VT идентифицирует события путем вычисления скорости движения от точки к точке и затем классифицирует событие как фиксацию или саккаду на основе значения этой скорости. Классический метод I-VT предназначен для классификации только фиксаций и саккад на основе входных данных, полученных при отслеживании движения глаз. Другие типы событий, такие как плавные движения, постсаккадические колебания и шумы, не учитываются. На рисунке 4 показано влияние изменения пороговых значений скорости на эффективность классификации алгоритма I-VT. Точности классификации каждого класса измеряются с использованием коэффициентов запоминания, точности и F1-баллов, рассчитанных на основе матрицы ошибок. Аналогично, в алгоритме I-DT увеличение порога увеличивает полноту для определения фиксаций (fixation recall) и точность для распознавания саккад (saccade precision), но при этом снижает точность распознавания фиксаций (fixation precision) и полноту для определения саккад (saccade recall).

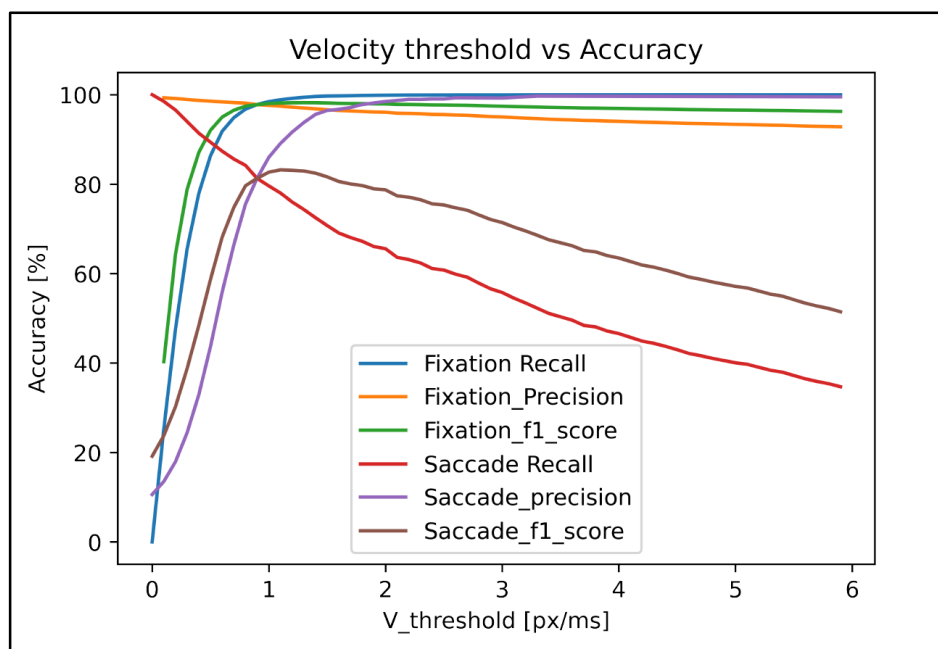


Рисунок 4 – Зависимость точности определения фиксаций и саккад алгоритма I-VT от пороговых значений

В представленном примере I-VT обеспечивает максимальную точность фиксации на уровне 99% при пороговой скорости 3,5 пикселей в секунду, а точность воспроизведения саккады немного снижается при увеличении порогового значения скорости. Повторение саккад достигает 98 %, а повторение фиксации – 25% при наименьшем пороговом значении скорости, равном 0,1 пикселя в секунду, так как при этом пороговом значении большинство точек классифицируются как саккады. Из-за влияния порогового значения на точность классификации алгоритма I-VT важно определить оптимальное пороговое значение для обоих методов фиксации и точности выполнения саккады. Таким образом, в данном случае оптимальное пороговое значение скорости для I-VT составляет 0,5 пикселей в секунду.

На текущий момент значение повторения фиксации составляет 92%, повторения саккады – 87%, точность фиксации – 96%, точность саккады – 46%, показатель фиксации F1 – 94% и показатель саккады F1 – 60%. Значение коэффициента каппы Коэна при оптимальном пороговом значении 0,6 пикселей/мс составляет всего 0,5, что указывает на умеренное согласие между людьми-программистами и алгоритмом классификации I-VT. Основным недостатком алгоритма является то, что он использует только скорость взгляда без учета других факторов, таких как ускорение сигнала, направление движения взгляда, расстояние между глазом и камерой и т. д. Это может привести к неправильной классификации событий, так как диапазоны скоростей самых быстрых движений глаз и самых медленных частей саккад могут пересекаться. Таким образом, представляется, что использование других параметров движения глаз, таких как ускорение, амплитуда и положение движения глаз, могло бы улучшить результаты.

2.4. Недостатки алгоритмов I-DT и I-VT

Стандартное оптимальное пороговое значение скорости отсутствует, и изменение пороговых значений влияет на результативность алгоритмов обнаружения событий. По этой причине разные специалисты применяют

разные пороговые значения для разработки и оценки эффективности алгоритмов I-VT. Из-за этих различий сложно сравнивать различные исследования алгоритмов обнаружения событий, основанных на пороговых значениях.

Обнаружение фиксаций и саккад при наличии плавного преследования

Одна из ключевых проблем, связанных с упомянутыми алгоритмами I-DT и I-VT, состоит в том, что они не принимают во внимание плавные события преследования. В отличие от этого, автоматизированные методы классификации, представленные в работе (ссылка), разделяют данные о движении глаз на фиксации, саккады и плавные движения. Эти методы улучшают существующие методы обнаружения событий, I-VT и I-DT, объединяя их и добавляя дополнительную пороговую скорость. Существуют модификации данных алгоритмов, такие как: IVVT, IVDT и IVMT.

Алгоритм IVVT определяет фиксации, саккады и плавные движения (smooth pursuits (SP-s)). Сначала он классифицирует фиксации и саккады с использованием существующего алгоритма I-VT, а затем идентифицирует плавные движения из фиксаций, добавляя дополнительную пороговую скорость.

Метод IVMP разделяет фиксации и саккады, применяя алгоритм I-VT, а затем отличает плавное движение от фиксации, используя схему движения. Согласно методу классификации, основанному на I-VT, измеренная скорость может использоваться для классификации образцов пристального взгляда как фиксаций или саккад. Однако, поскольку плавные движения преследования могут иметь скорость, аналогичную скорости фиксации, метод простой скорости не может быть использован для различения плавного преследования и фиксации. Чтобы определить, выполняет ли глаз фиксирующее движение или плавное отслеживание, направление движения анализируется во временном окне размером T_w . В этом окне величина движения вычисляется путем анализа углов, создаваемых каждой

парой соседних позиционных точек, и по горизонтальной оси координат. Затем величина перемещения сравнивается с пороговым значением перемещения T_m . Если величина перемещения превышает пороговое значение, оно помечается как плавное преследование, а если оно ниже порогового значения, оно помечается как фиксация.

Алгоритм IVDT использует методы I-VT и I-DT для классификации фиксаций, саккад и плавных действий. Вначале применяется порог скорости V_t для разделения данных на саккады и фиксации, и точка считается саккадой, если ее скорость превышает V_t . Затем используется порог дисперсии V_t для разделения оставшихся данных на фиксированные и плавные последовательности.

Принцип работы алгоритма IVMP аналогичен описанному выше. Он использует V_t и схему движения для классификации событий как фиксацию, саккаду или плавное преследование. Сначала применяется V_t для классификации всех данных по фиксациям и саккадам, а затем по шаблону движения T_m для различения плавных действий и фиксаций.

Для расчета оптимальных пороговых значений для каждого алгоритма можно использовать качественные и количественные оценки поведения. Однако эти оценки основаны на данных и могут отличаться для данных, полученных с помощью разных офтальмологических трекеров. Поэтому поиск оптимальных пороговых значений все еще остается актуальным.

Автоматизированный метод классификации событий на основе пороговых значений скорости, основанных на данных

Основная проблема предыдущих подходов заключалась в определении корректного порога, что особенно сложно при наличии зашумленного сигнала. В связи с этим был предложен автоматизированный метод классификации событий, основанный на данных о пороговой скорости. Алгоритм способен адаптивно определять порог и нивелировать влияние шума. Кроме того, он выделяет глиссады как отдельные типы событий. Этот метод разработан для преодоления проблемы чувствительности к шуму, воз-

никающей в предыдущих алгоритмах, путем определения адаптивных значений V_t с учетом различных уровней возникновения шума. Он устраняет шумы и нежелательные вариации, вычисляя профили скорости и ускорения, которые в предыдущих работах рассчитывались с помощью простого вычитания от выборки к выборке. Однако результат такого расчета является зашумленным. Автоматизированный метод, основанный на данных, устраняет шум, вычисляя скорость и ускорение, которые определяют скорость и ускорение с использованием конечной импульсной характеристики (finite impulse response – FIR) с помощью фильтров.

Недостаток этого метода заключается в том, что глиссада определяется исключительно по длительности. Это значит, что она происходит в половине времени саккады. Таким образом, саккада с короткой длительностью может быть ошибочно принята за глиссаду, а глиссада с большой длительностью может быть интерпретирована как саккада, так как нет других параметров для различения этих движений. Алгоритм предназначен для обнаружения глиссад только при наличии фиксации и саккады и не принимает во внимание другие события, такие как ПП и ПСК. Также он не способен обрабатывать глиссадоподобные движения, предшествующие саккаде.

Методы обнаружения событий, основанные на машинном обучении

Основной недостаток всех алгоритмов, основанных на пороговых значениях, заключается в том, что пользователю приходится самостоятельно настраивать параметры и искать оптимальные значения порогов. Еще один минус таких алгоритмов – их ориентация на решение конкретных задач, таких как фиксация и саккада.

Машинное обучение позволяет решить эти проблемы, автоматически классифицируя необработанные данные визуального отслеживания без ручной настройки параметров и поиска пороговых значений. Обучение моделей происходит на основе обучающих данных, и обычно предполага-

ется, что классификация точки наблюдения зависит от ее близости к другим точкам.

Стандартными входными данными для моделей являются наборы свойств нескольких точек наблюдения до и после классификации. Свойства могут включать координаты, скорость, ускорение, направление движения и рывок. Размер окна – один из ключевых параметров каждой модели.

2.5. Случайный лес (RF)

Полностью автоматизированная классификация событий, связанных с движением глаз, с использованием классификатора Random Forest для классификации фиксаций, саккад и постсаккадических колебаний. Эффективность классификации сравнили с современными алгоритмами и экспертами, работающими вручную. Результаты показали, что алгоритм машинного обучения превосходит современные алгоритмы и почти достигает производительности экспертов. Однако эта производительность была достигнута только для высококачественных данных с низким уровнем шума.

В этом разделе описана реализация алгоритма, использующую модель классификации случайных лесов для классификации событий. Был внедрен алгоритм классификации случайных лесов для классификации данных отслеживания глаз по фиксациям, саккадам и ПСК. Оценена эффективность классификации в отношении точности классификации фиксаций, точности определения саккад и ПСК.

Этот алгоритм может обнаруживать движения глаз в непрерывном потоке взгляда и назначать метки для всех трех типов движений глаз одновременно. Поэтому дополнительно была оценена эффективность классификации алгоритма, определяя ее отдельно для задачи обнаружения трех классов путем оценки прогнозов выборки за выборкой, матрицы ошибок и, наконец, оценена эффективность классификации каждого класса.

Для построения модели была использована скорость в качестве параметра: выполнены преобразования координатных точек eye tracker в область скоростей и создали последовательности выборок длиной 40. Также протестированы более короткие и длинные последовательности, но это существенно не повлияло на результат. Таким образом, входными данными для модели стала последовательность пристальных взглядов образцы размером 40×2 . На рисунке 5 показана матрица ошибок для оценки выборки за выборкой. Фиксации правильно помечены в 97 % случаев, в то время как для ПСК и саккад есть тенденция к тому, чтобы их помечали как фиксации. Саккада и ПСК правильно идентифицируются в 91 % и 76 % кадров соответственно. Около 7 % ПСК ошибочно классифицируются как фиксация, а 17 % ПСК классифицируются как саккада. Это происходит потому, что большинство обучающих событий были фиксациями, и модель, естественно, стремится классифицировать все неоднозначные выборки как фиксации.

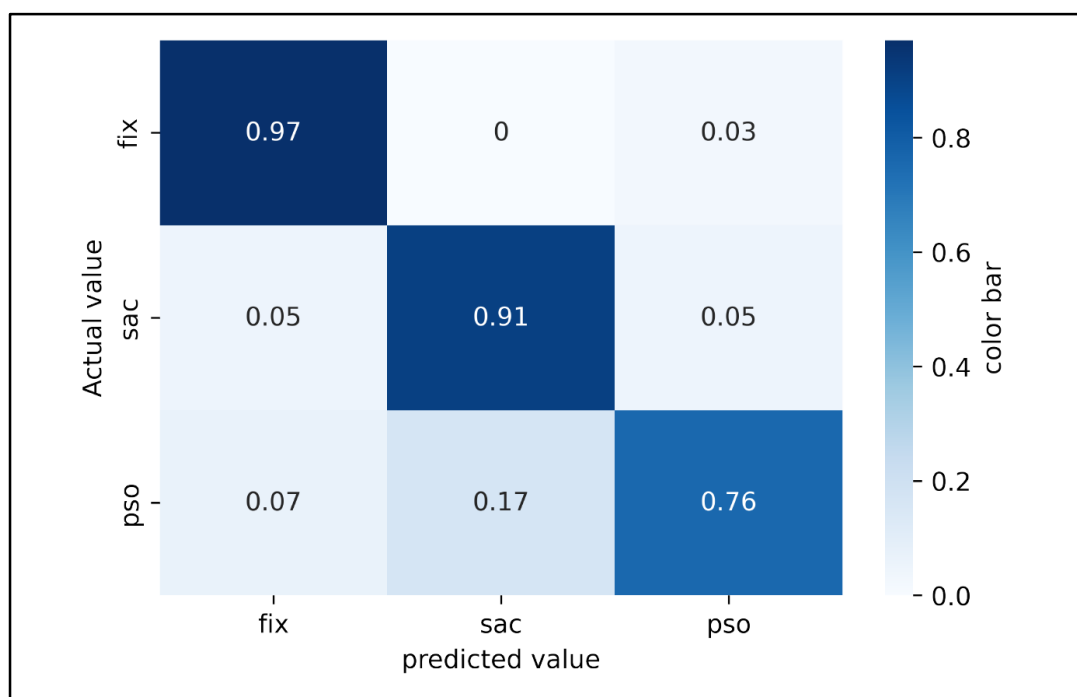


Рисунок 5 – Матрица ошибок классификатора случайный лес

В таблице 1 представлены метрики работы классификатора RF. Результаты классификации показывают, что RF хорошо справляется с клас-

сификацией фиксаций и саккад. При этом эффективность обнаружения ПСК далека от идеала.

Таблица 1 – Метрики классификатора случайный лес

	Accuracy	Precision	Recall	F1-Score
Фиксация	97%	99%	97%	98%
Саккада	92%	87%	91%	89%
ПСК	76%	64%	76%	69%

2.6. Сверточная нейронная сеть (CNN)

Сверточные нейронные сети хорошо подходят для анализа закономерностей в данных, поэтому их можно использовать для обнаружения событий, связанных с движением глаз. Один из примеров такого применения – метод, который основан на глубокой сверточной нейронной сети. Эта сеть для каждой выборки предсказывает вероятности принадлежности к фиксации, саккаде или плавному преследованию на основе последовательности выборок взгляда. Этот метод направлен на устранение недостатков предыдущих методов, которые использовали форму и амплитуду сигнала для определения или классификации движения глаз, что может быть проблематично, например, при плавном движении.

Предлагаемый метод использует частоту сигнала для классификации данных по типам событий. Это означает, что сначала необработанные данные о взгляде преобразуются в частотную область необработанного сигнала с использованием быстрого преобразования Фурье (FFT), а затем частотное представление сигнала передается в сверточную нейронную сеть, которая, в свою очередь, выдает на выходе трехмерный сигнал активации. Каждый сигнал представляет вероятность каждого типа движения глаз (фиксация, саккада и ПП). Наконец, метка с высокой вероятностью присваивается центральному образцу в окне.

Метод не является сквозным, поскольку входные данные для сети являются выходными данными ПСК. В нем используются функции, созданные вручную, – входные данные, которые необходимо преобразовать в

частотную область. Предлагаемый метод классифицирует фиксации, саккады и плавные движения без учета других событий, таких как ПСК. Метод превосходит старые алгоритмы, основанные на простом определении дисперсии и пороговой скорости.

Чтобы проверить способность сети CNN классифицировать события, связанные с движением глаз, создана простая сеть, представленная на рисунке 6. В качестве входных данных сеть использует непрерывный поток двумерных выборок взгляда. Чтобы получить прогноз для каждой выборки взгляда, окно перемещается по последовательности один за другим. Сначала идет преобразование данных о координатах x и y в горизонтальную и вертикальную составляющие скорости, вычисляя скорость от выборки к выборке. Для получения соответствующих характеристик движения глаз поток выборок взгляда анализируется в окнах по 100 выборок, что дает наилучшие результаты в экспериментах.

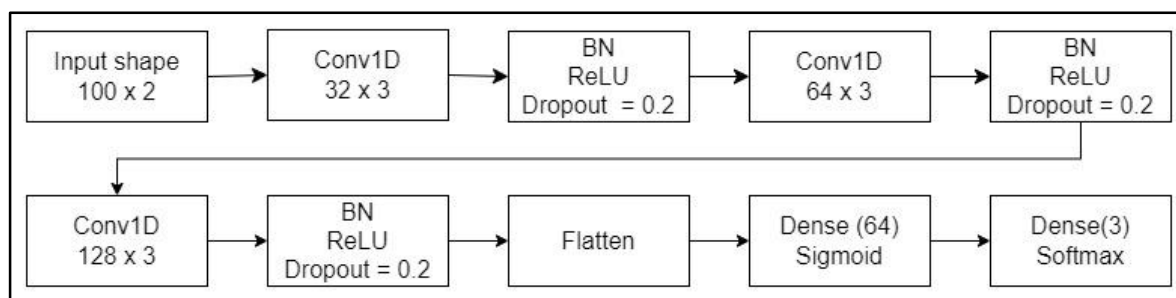


Рисунок 6 – Архитектура сверточной нейронной сети

Сеть состоит из нескольких слоев, включая три сверточных слоя с постепенно увеличивающимся количеством фильтров (32, 64 и 128) и размером ядра 3. Перед активацией используются операции пакетной нормализации. Имеется выходной слой. Входные данные для сети представляют собой последовательность выборок пристального взгляда в формате 100×2 . Архитектура сети представлена на рисунке 6.

На рисунке 7 представлена матрица ошибок для классификации CNN. Фиксации правильно классифицируются в 99% случаев, саккады – в 88%, а ПСК – в 76% случаев. В общей сложности 4% саккад ошибочно классифи-

цируются как фиксации, а 16% ПСК – как ложные фиксации. Кроме того, 8% саккад классифицируются как ПСК.

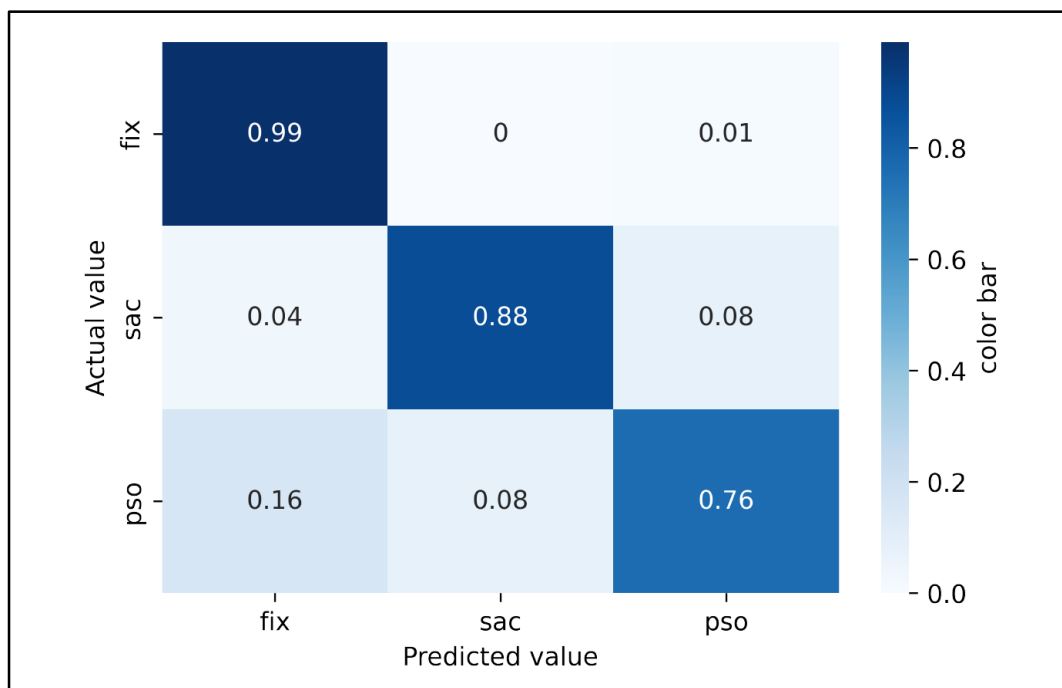


Рисунок 7 – Матрица ошибок сверточной нейронной сети

В таблице 2 представлены метрики работы сверточной нейронной сети. Результаты классификации показывают, что CNN хорошо справляется с фиксацией и классификацией саккад. Однако эффективность классификации для ПСК далека от идеала.

Таблица 2 – Метрики для сверточной нейронной сети

	Accuracy	Precision	Recall	F1-Score
Фиксация	99%	98%	99%	99%
Саккада	89%	93%	89%	91%
ПСК	75%	83%	75%	79%

2.7. Анализ результатов

Для сравнения работы различных алгоритмов, была проведена оценка производительности четырех разных алгоритмов классификации событий из областей пороговых значений, машинного обучения и глубокого анализа. Также была учтена совместная работа двух оценщиков-людей.

В таблице 3 представлены показатели эффективности для каждого класса событий, в строках указаны метрики, а в столбцах указаны алгоритмы классификации.

Таблица 3 – Сравнительная таблица эффективности классификаторов

	I-VT	I-DT	RF	CNN	Coder MN	Coder RA
Fixation Accuracy	92%	95%	97%	99%	99%	99%
Saccade Accuracy	87%	93%	92%	89%	92%	96%
PSO Accuracy	–	–	76%	75%	88%	82%
Fixation F1-score	94%	96%	99%	99%	99%	99%
Saccade F1-score	60%	66%	87%	91%	94%	94%
PSO F1-Score	–	–	64%	79%	85%	85%
Fixation Recall	92%	95%	97%	99%	99%	99%
Saccade Recall	87%	93%	92%	89%	92%	96%
PSO Recall	–	–	76%	75%	88%	82%
Fixation Precision	96%	98%	99%	98%	99%	99%
Saccade Precision	46%	51%	87%	93%	96%	92%
PSO precision	–	–	64%	83%	82%	88%
Cohen's kappa	0,5	0,6	0,83	0,88	1	0,9

Результаты показали, что I-DT работает лучше, чем I-VT, по всем показателям производительности. Однако алгоритмы RF и CNN превосходили алгоритмы на основе пороговых значений (I-VT и I-DT) по всем показателям производительности, за исключением полноты для саккады (recall метрика). В случае классификационных моделей RF и CNN существенной разницы в классификации фиксации и саккад не обнаружено. Однако CNN превосходит RF по точности ПСК, F1-баллу и коэффициенту каппы Коэна. В приложении Б представлена сравнительная таблица исследованных алгоритмов.

Чтобы подчеркнуть, что представленный результат учитывает только точечные сравнения, каждая точка обзора классифицируется как часть определенного события. На самом деле само событие занимает некоторое время. Например, средняя продолжительность фиксации составляет около 250 мс. Поэтому следующим типичным шагом при обнаружении события является преобразование последовательности точек, классифицированных как фиксации, в одну фиксацию с местоположением, вычисленным как

среднее из местоположений этих точек. Если между последовательностями фиксации есть промежутки (когда несколько точек классифицируются по-разному), две последовательности считаются отдельными фиксациями. Это существенно влияет на такие конкретные показатели, как общее количество фиксаций и средняя продолжительность фиксации. Проведено сравнение полученных результатов после объединения последующих точек.

Результаты для I-VT и RF представлены на рисунках 8 и 9 соответственно. Алгоритм I-VT обнаружил 189 фиксаций со средней продолжительностью 121 мс, в то время как алгоритм CNN обнаружил только 64 фиксации со средней продолжительностью 121 мс. Средняя длительность составила 264 мс. Это хорошо видно на рисунках 8 и 9.

Учитывая, что ручной кодировщик обнаружил 91 запись со средней продолжительностью 222 мс, можно сделать вывод, что алгоритмы, основанные на пороговых значениях, требуют дополнительного этапа объединения близлежащих записей (дополнительные пороговые параметры).

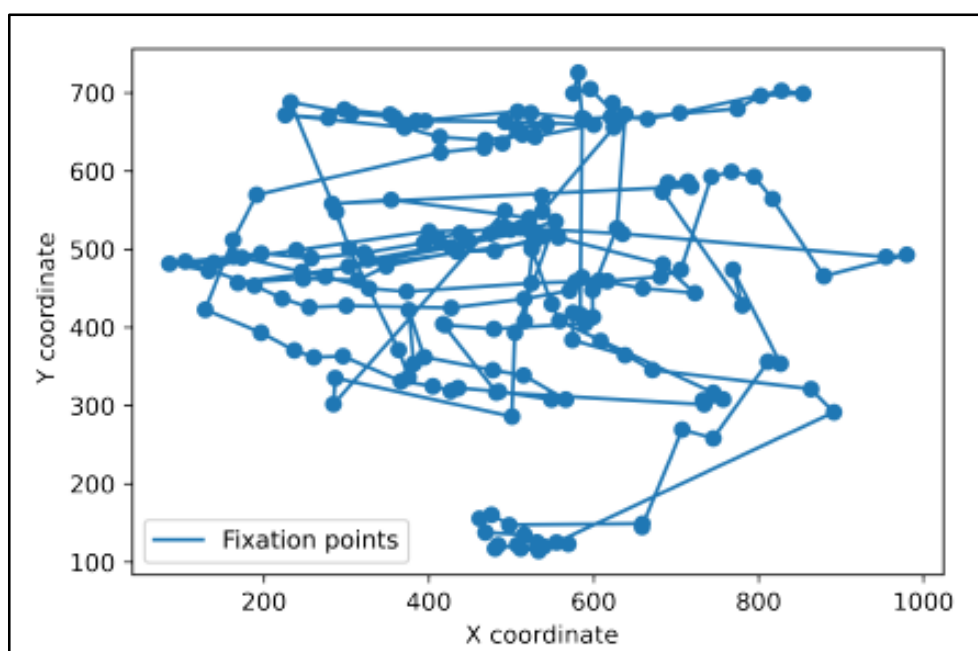


Рисунок 8 – Результат работы алгоритма I-VT
(при пороговом значении 3.5 px/ms)

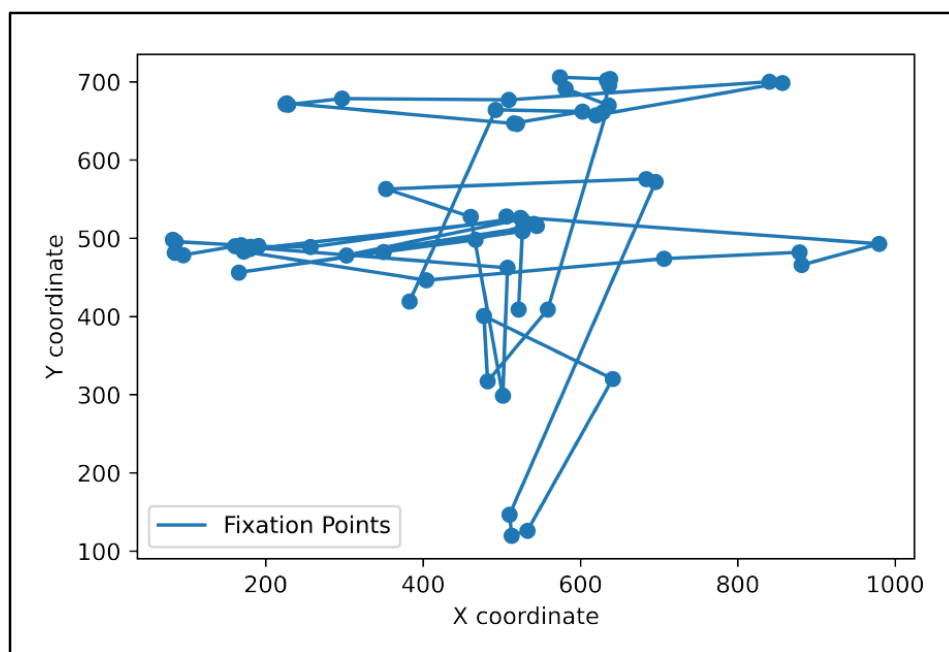


Рисунок 9 – Результат работы алгоритма CNN

В отличие от алгоритмов, основанных на пороговом значении, алгоритмы машинного обучения решают проблему объединения близлежащих записей самостоятельно.

Выбор алгоритма для приложения

Исходя из полученных экспериментальных результатов, следует, что классификатор на основе рекуррентной нейронной сети на данном датасете показал наилучший результат среди всех представленных алгоритмов. Поэтому в итоговом приложении будет использован именно этот алгоритм. Алгоритм будет выполнять задачу классификации событий на датасете отслеживания взгляда.

В ходе выполнения экспериментов можно сказать о том, что разработка приложения именно для задач сравнения и оценки эффективности работы алгоритмов не будет полезной. Поскольку существует множество различных АОС отслеживания взгляда и их модификаций. Многие из них требуют точного подбора порогового значения и могут ввести в заблуждение пользователя. К тому же алгоритмы машинного и глубокого обучения дают гораздо лучшие результаты, близкие к человеческим, при этом не

требуя подбора параметров, а также могу идентифицировать больше классов событий.

Было решено создать приложение, которое, используя один из АОС, будет осуществлять классификацию событий в датасете. На данном датасете наилучший результат, самый близкий к человеку-кодировщику, показал АОС на основе рекуррентной нейронной сети. Несмотря на то, что данный алгоритм показал чуть худшую эффективность в обнаружении саккад по сравнению с классификатором RF, он превзошел все остальные алгоритмы в других критериях и показал результаты, практически идентичные с кодировщиками-людьми.

Вывод по второй главе

Во второй главе были получены экспериментальные результаты работы алгоритмов обнаружения событий, используя один и тот же набор данных. Также были проведены этапы подбора наилучших параметров для алгоритмов, которые используют пороговые значения, проведен анализ результатов. Оценена согласованность между людьми-кодировщиками и АОС. Были установлены основные недостатки и преимущества алгоритмов, проведено сравнение эффективности их работы. В качестве наилучшего алгоритма был выбран классификатор на основе рекуррентной нейронной сети. Поскольку, как и RF, CNN также решает проблемы, связанные с методом обнаружения порога. Она выполняет одноэтапное сквозное обнаружение без вмешательства человека и обеспечивает обнаружение на уровне человека для идентификации фиксации.

3. РАЗРАБОТКА ПРИЛОЖЕНИЯ

Данный раздел содержит информацию о выбранных средствах реализации и их особенностях. Описывает взаимодействие пользователя с приложением, общее описание архитектуры системы.

3.1. Выбор средств реализации

Для разработки приложения будет использована кроссплатформенная среда разработки компьютерных игр – Unity. Выбор данного решения обоснован следующими преимуществами:

- 1) возможность создавать приложения;
- 2) широкая поддержка платформ;
- 3) наличие встроенного функционала по созданию пользовательского интерфейса.

Основным преимуществом будет являться собственный опыт работы в данной среде разработки. Функционала данной среды будет достаточно для создания приложения для пользовательских компьютеров под управлением операционной системы «Windows 10». Наличие функционала по созданию пользовательского интерфейса значительно ускорит и упростит создание и разработку приложения, позволит создать простое и понятное для конечного пользователя приложение.

Однако Unity не имеет возможности выполнять Python код, а также работать напрямую с такими файлами и скриптами. Что усложнит задачу, так как потребует написания средства, которое бы связало среду Unity с кодом на Python. В таком случае Unity будет выполнять роль отображения и представления информации для пользователя, а также будет предоставлять возможность работы с интерфейсом, в то время как алгоритм обработки данных будет исполняться в Python.

3.2. Общее описание архитектуры системы

Основу для функционирования системы будет составлять среда разработки Unity, алгоритм классификации событий, написанный на языке Python, а также API, которое будет связывать среду Unity и алгоритм для обработки данных. На рисунке 10 представлена диаграмма компонентов приложения.

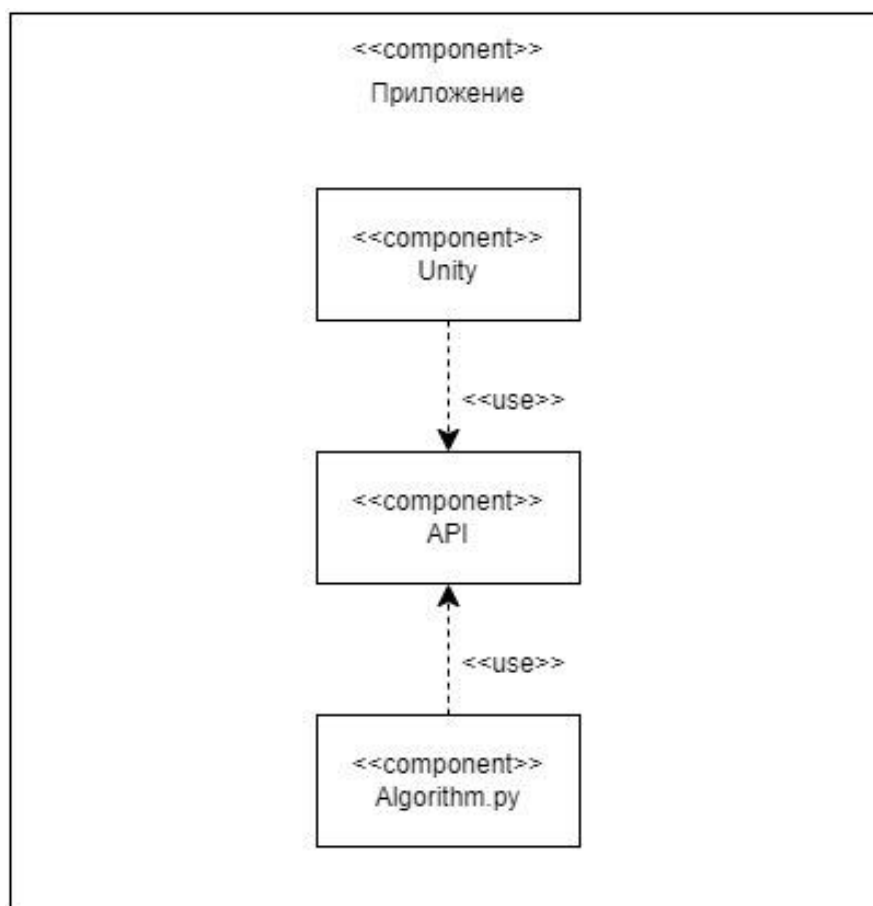


Рисунок 10 – Диаграмма компонентов

Компонент Unity – предоставляет графический интерфейс для конечного пользователя, обеспечивает взаимодействие с графическими элементами. Является исполняемым приложением.

Компонент Algorithm.py – является обученным алгоритмом, с помощью которого происходит обработка данных.

Компонент API – связывает Unity и Algorithm.py.

Диаграмма основных вариантов использования

Для проектирования системы был использован язык графического описания UML. На рисунке 11 представлена диаграмма вариантов использования, отражающая модель взаимодействия актера «Пользователь» с приложением. С системой взаимодействует только один основной актер – «Пользователь», использующий приложение. Пользователь запускает приложение, загружает датасет, после чего может скачать размеченные данные.

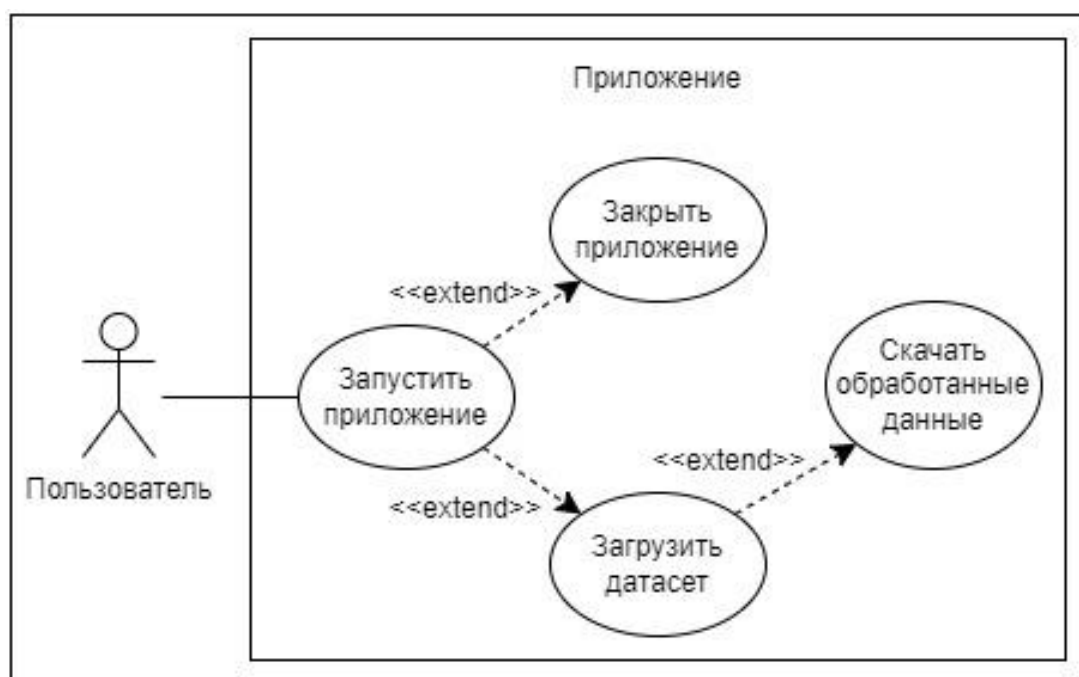


Рисунок 11 – Диаграмма вариантов использования

В ходе использования приложения пользователь может загрузить датасет, после чего скачать обработанные данные. Также у пользователя есть возможность закрыть приложение.

Спецификации основных вариантов использования представлены в приложении А.

3.3. Представление пользовательского интерфейса

В данном разделе будет представлен макет пользовательского интерфейса приложения. Для создания дизайна интерфейса необходимо учи-

тывать требования к функциональности и пользовательскому опыту. В ходе реализации или тестирования элементы пользовательского интерфейса могут быть изменены.

Для улучшения взаимодействия между пользователем и программным обеспечением, функциональные элементы будут размещены рядом, в одной области экрана, а информативные по центру. Приложение будет состоять из одного окна и содержать следующие элементы:

- 1) загрузить датасет – загрузить датасет, который необходимо обработать;
- 2) скачать обработанные данные – скачать размеченные данные;
- 3) статус – отображение текущего статуса работы;
- 4) выход – завершить работу приложения.

На рисунке 12 представлен макет приложения.

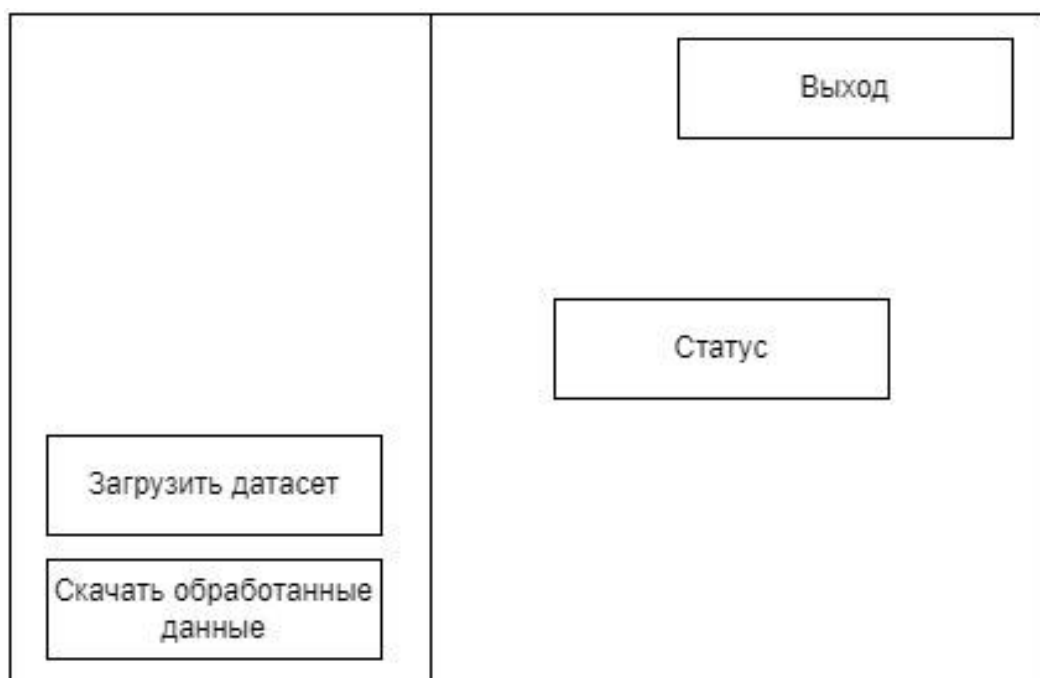


Рисунок 12 – Макет приложения

При использовании приложения может сразу выбрать загрузку датасета. После начала загрузки данных можно наблюдать текущий статус приложения и своевременно узнать о готовности к скачиванию обработан-

ных данных. По готовности к скачиванию пользователь может выбрать скачивание обработанных данных.

Итоговый вид приложения

На рисунке 13 представлен скриншот финального окна приложения. Расположение элементов было переработано относительно спроектированного шаблона.



Рисунок 13 – Окно приложения

Переработка дизайна позволила выделить верхнюю панель для интерактивных элементов в верхней части экрана. Функциональные элементы, затрагивающие основной функционал приложения, при этом остались рядом и расположены в верхнем левом углу.

Вывод по третьей главе

В данной главе были рассмотрены основные преимущества и недостатки выбранных средств реализации. Исходя из особенностей средств реализации, была спроектирована архитектура приложения. Исходя из функциональных требований, спроектирована диаграмма вариантов использования приложения. Создано приложение, соответствующее заданным функциональным требованиям.

ЗАКЛЮЧЕНИЕ

В данной работе были изучены алгоритмы обнаружения событий из различных областей: I-VT и I-DT, основанные на пороговых значениях, модель случайного леса из области машинного обучения и модель CNN из области глубокого обучения. Была проведена оценка эффективности их классификации с использованием одного и того же набора данных для всех методов. Также обсуждалась взаимосвязь между программистами и алгоритмами. Результаты показали критическое влияние пороговых значений на классификацию алгоритмов I-VT и I-DT. В связи с этим поиск оптимального порога в алгоритмах, основанных на пороговых значениях, является сложной задачей. Алгоритмы RF и CNN превосходят алгоритмы, основанные на пороговых значениях, по всем показателям производительности и способны классифицировать данные по нескольким классам.

Также в работе были изучены алгоритмы обнаружения событий для классификации данных о событиях по фиксациям, саккадам и ПСК. События с плавным преследованием не рассматривались, использовались только три типа событий, так как в алгоритмах для классификации событий использовались только параметры скорости. Для различения ПП и фиксаций требуется более высокое значение скорости из-за схожего поведения с точки зрения скорости. В будущих исследованиях планируется расширить классификацию, включить плавные движения и другие виды событий.

Было разработано приложение с использованием среды Unity, использующее АОС на основе рекуррентной сети. Приложение имеет возможность обработать загружаемый датасет и получить обработанные данные. В будущем планируется расширить функционал приложения, написать документацию по использованию.

ЛИТЕРАТУРА

1. Komogortsev O.V., Gobert D.V., Jayarathna S., Koh D. H., Gowda S. M. Standardization of Automated Analyses of Oculomotor Fixation and Saccadic Behaviors. // IEEE Transactions on Biomedical Engineering, 2010. – 2635–2645 pp.
2. Bahill A.T., Clark M.R., Stark L.W. The main sequence, a tool for studying human eye movements. // Bellman Prize in Mathematical Biosciences, 1975. – 191–204 pp.
3. Kapoula Z. Evidence for a range effect in the saccadic system. // Vision Research, 1985. – 1155–1157 pp.
4. Holmqvist K. Eye tracking: a comprehensive guide to methods and measures. // Oxford University Press, 2011. – 101–103 pp.
5. Wade N.J., Tatler B.W., Heller D. Dodge-ing the issue: Dodge, Javal, Hering, and the measurement of saccades in eye-movement research. // Perception, 2003. – 793–804 pp.
6. Захариков В.С., Клусов Д.В., Палей А.Д., Плотников А.А. Краткие результаты разработки оптико-электронной системы наведения образцов ввст по положению глаза оператора. алгоритм сокращения времени калибровки в системах айтрекинга. // Союз машиностроителей России. Национальная научно-техническая конференция, 2021. – С. 37–40.
7. Ахмедов Д.Ф., Лисицына Л.С. Автоматизация проверки уровня знаний языков программирования методом отслеживания глаз. // Вестник науки и образования, 2020. – С. 22–27.
8. Малин И.К., Крапивенко А.В. Система отслеживания направления взгляда с использованием доступной видеоаппаратуры. // Труды МАИ, 2009. – С. 17–18.
9. Бутенко В.В. Поиск границы радужки на изображении. // Известия высших учебных заведений России, 2016. – С. 36–39.

10. Dalveren G.G.M., Cagiltay N.E. Evaluation of Ten Open-Source Eye-Movement Classification Algorithms in Simulated Surgical Scenarios. // IEEE Access, vol. 7, 2019. – 161794–161804 pp.
11. Birawo B., Kasprowski P. Review and Evaluation of Eye Movement Event Detection Algorithms. // Sensors, 2022. – 1–10 pp.
12. Startsev M., Zemblys R. Evaluating Eye Movement Event Detection: A Review of the State of the Art. // Behavior research methods, 2023. – 1653–1714 pp.
13. Казакова А.С. Регистрация траектории движения глаз человека. // Решетневские чтения, 2011. – С. 616–617.
14. Орлов П.А., Лаптев В.В., & Иванов В.М. К вопросу о применении систем ай-трекинга. // Информатика, телекоммуникации и управление, 2014. С. – 82–92.
15. Куликов, И. А. Возможности применения трекинга глаз в образовании. // Гуманитарная информатика, 2013. – С. 121–125.
16. Хайдарлы, А. И., Кофанова, Т. В., & Путивцева. «Eye-tracking - технология будущего». // МНИЖ, 2014. – С. – 76–77.
17. Буденкова Е.А., Ваколюк И.А. Скорость и точность зрительно вызванных саккад у детей 2–9 лет. // Ученые записки Крымского федерального университета имени В. И. Вернадского. Биология. Химия, 2019. – С. 3–12.
18. Игнатовский Я.Р., Иванов В.Г. Айтрекинг: потенциал для применения в государственном управлении, политическом брендинге и планировании избирательных кампаний. // Вестник Российского университета дружбы народов. Серия: Государственное и муниципальное управление, 2020. – С. 155–160.
19. Грушко А.И. Использование систем регистрации движений глаз в психологической подготовке спортсменов. // Дис. канд. псих. Наук: 19.00.03, 2017. – 33 с.

20. Andrew Duchowski. Eye Tracking Methodology Theory and Practice. // Springer Science & Business Media, 2007. – 55–57 pp.
21. Holmqvist K., Nyström M., Andersson R., Dewhurst R., Jarodzka H., Weijer J. Eye Tracking. A Comprehensive Guide to Methods and Measures. // OUP Oxford, 2011. – 45–47 pp.
22. Bojko A. Eye Tracking the User Experience. A Practical Guide to Research. // Rosenfeld Media, 2013. – 89–92 pp.
23. Wedel M., Pieters R. Eye Tracking for Visual Marketing // Now Publishers, 2008. – 101–106 pp.
24. Conklin K., Pellicer A., Carrol G. Eye-Tracking A Guide for Applied Linguistics Research. // Cambridge University Press, 2018. – 143–145 pp.

ПРИЛОЖЕНИЯ

Приложение А. Спецификация вариантов использования

Спецификация вариантов использования (ВИ) приведена в таблицах 1–4.

Таблица 1 – Спецификация ВИ «Запустить приложение»

Прецедент: запустить приложение
ID: 1
Краткое описание: запуск приложения позволяет перейти к загрузке датасета
Главные актеры: пользователь
Второстепенные актеры: отсутствуют
Предусловия: успешный запуск приложения
Основной поток: вариант использования начинается, когда пользователь запускает приложение
Постусловия: отсутствуют
Альтернативные потоки: отсутствуют

Таблица 2 – Спецификация ВИ «Закрыть приложение»

Прецедент: запустить приложение
ID: 2
Краткое описание: запуск приложения позволяет перейти к загрузке датасета
Главные актеры: пользователь
Второстепенные актеры: отсутствуют
Предусловия: успешный запуск приложения
Основной поток: вариант использования начинается, когда пользователь запускает приложение
Постусловия: отсутствуют
Альтернативные потоки: отсутствуют

Таблица 3 – Спецификация ВИ «Загрузить датасет»

Прецедент: загрузить датасет
ID: 3
Краткое описание: пользователь может загрузить датасет
Главные актеры: пользователь
Второстепенные актеры: отсутствуют
Предусловия: отсутствуют
Основной поток: вариант использования начинается, когда пользователь начинает загрузку датасета
Постусловия: отсутствуют
Альтернативные потоки: отсутствуют

Таблица 4 – Спецификация ВИ «Скачать обработанные данные»

Прецедент: скачать обработанные данные
ID: 4
Краткое описание: пользователь может скачать обработанные данные
Главные актеры: пользователь
Второстепенные актеры: отсутствуют
Предусловия: успешно обработан датасет
Основной поток: вариант использования начинается, когда пользователь скачивает обработанные данные
Постусловия: отсутствуют
Альтернативные потоки: отсутствуют

Приложение Б. Сравнительная таблица алгоритмов

В таблице 1 представлены слабые и сильные стороны алгоритмов.

Таблица 1 – Преимущества и недостатки алгоритмов

Алгоритм	Преимущества	Недостатки
Ручная обработка данных	Ручное кодирование по-прежнему широко применяется для оценки алгоритмов обнаружения событий, а данные, классифицированные вручную, служат основой для обучения алгоритмов машинного обучения.	Много времени тратится на то, что разные специалисты применяют различные субъективные критерии выбора, приводящие к разным результатам, так как параметры и пороги устанавливаются в таком случае самостоятельно.
I-VT	Простая реализация и легкость в понимании. Одно пороговое значение, основанное на скорости, используется для идентификации событий на основе необработанных входных данных. Отлично подходит для фиксации и определения саккад за один этап идентификации. Требует низких вычислительных ресурсов.	Хотя метод I-VT прост, в реальных условиях он используется редко. Он чувствителен к зашумленным сигналам с большим количеством выбросов. Поиск оптимального порогового значения сложен, так как не существует стандартного оптимального значения. Метод определяет только фиксации и саккады.
I-DT	Первый автоматизированный алгоритм обнаружения событий. Выполняет идентификацию фиксаций и саккад с показателями идентификации на уровне человека. I-DT часто доступен в коммерческом программном обеспечении.	Выбор пороговых значений влияет на производительность. Выбор метода расчета дисперсии также является сложной задачей, так как разные методы расчета дисперсии влияют на ее значение. Предназначен только для идентификации фиксаций и саккад.
RF	Не требуется задавать пороговое значение. Осуществляет мультиклассовую классификацию, поэтому может использоваться для различных событий. Это полностью автоматизированный метод классификации событий.	Требует значительного количества правильно размеченных данных для обучения. В реализации, представленной в работе, для определения событий (фиксация, саккада и ПСК) использовались только показатели скорости.
CNN	Как и в случае с RF, CNN решает проблемы, связанные с методом обнаружения на основе порогов. Она выполняет одноэтапное сквозное обнаружение без участия человека и обеспечивает уровень обнаружения, сопоставимый с человеческим, для идентификации фиксаций.	Для обучения CNN требуется еще больше корректно размеченных данных, чем для алгоритма RF. В работе использовался только параметр скорости для идентификации событий на основе входных данных. Используемый в исследовании параметр скорости недостаточен для различения плавного преследования и фиксации, поскольку оба типа движения имеют низкую скорость.