# Расширение метода окулографического пространственного мониторинга с использованием мультимодальной CNN с геометрическими признаками

И.А. Колидов, М.А. Бакаев

***Аннотация*—В статье представлен экономичный и воспроизводимый подход к отслеживанию взгляда пользователя с использованием стандартной веб-камеры и мультимодальной сверточной нейронной сети (CNN). В отличие от традиционных систем окулографии (eye-tracking), требующих дорогостоящего оборудования (EyeLink, Tobii), предложенное решение опирается на комбинацию визуальных и геометрических признаков, а также специально разработанную нейросетевую архитектуру, объединяющую четыре сверточные ветви для обработки изображений и три полносвязные ветви для обработки числовых данных. Метод также позволяет смягчить ограничения, связанные с освещением, положением головы и индивидуальными особенностями пользователей. Для обучения модели был создан мультимодальный датасет размером 10 ГБ, включающий 2 млн. изображений и записи геометрии глаз. После подбора гиперпараметров с использованием Ray Tune и алгоритма ASHA, наша модель достигла средней ошибки RMSE 30.23 пикселя, что на 61% лучше по сравнению с предыдущей версией метода. При этом время инференса составило 15.2 мс (≈65 FPS), что делает систему пригодной для работы в режиме реального времени. Сравнение с профессиональными трекерами (EyeLink 1000 Plus, AdHawk MindLink) показало, что предложенная в нашей работе модель занимает промежуточное положение по точности, требуя при этом оборудование кратно меньшей стоимости. Полученные результаты демонстрируют потенциал применения мультимодальных сетей для окулографии в юзабилити-тестировании, UX-аналитике и исследованиях в области человеко-компьютерного взаимодействия.**

***Ключевые слова*—отслеживание взгляда; мультимодальная нейронная сеть; веб-камера; юзабилити-тестирование; геометрические признаки.**

## I. Введение

Юзабилити-тестирование (ЮТ) – это процесс оценки интерфейсов с участием реальных пользователей, выполняющих заранее подготовленные задачи. Целью ЮТ является выявление проблем в конструкции интерфейса и улучшение пользовательского опыта (UX), обеспечивая простоту, интуитивность и соответствие продукта потребностям аудитории [1]. Несмотря на то, что традиционные методы, такие как анкеты и интервью, позволяют оценивать UX, они во многом субъективны [2].

За последние несколько лет значительно вырос интерес к объективному мониторингу впечатлений пользователей во время ЮТ, особенно в области проектирования пользовательского опыта и взаимодействия человека и компьютера (HCI). Одним из наиболее информативных методов объективного мониторинга является отслеживание глаз (eye-tracking).

Технология отслеживания взгляда фиксирует положение зрачков и направления взгляда на экране и вычисляет точки фиксации и саккады. Она позволяет анализировать внимание, когнитическую нагрузку и визуальные предпочтения, выявляя элементы интерфейса, привлекающие или, напротив, остающиеся без внимания [3, 4]. На основе таких данных строятся тепловые карты, визуализирующие концентрацию взгляда и помогающие оценивать эффективность расположения элементов интерфейса.

В традиционных методах для отслеживания взгляда используются промышленные системы, такие как EyeLink (SR Research) [5] и Tobii Pro (Tobii AB) [6]. Эти устройства фиксируют движения глаз с помощью инфракрасных датчиков или технологии микрозеркал MEMS, обеспечивая высокую точность (0.1–0.5° визуального угла) и частоту регистрации до 2,000 Гц. Однако их стоимость может достигать $10,000-$30,000, что делает подобные системы недоступными для небольших организаций и независимых исследователей.

Для решения проблемы, исследователи начали разрабатывать альтернативные, более доступные методы отслеживания глаз, основанные на компьютерном зрении и нейронных сетях. Так, ранее в нашей работе [7] был предложен подход на основе сверточной нейронной сети (CNN) с использованием данных, собранных при контролируемых условиях, который показал среднюю ошибку RMSE 77.9 пикселей – приемлемую для анализа пользовательского взаимодействия, где элементы интерфейса обычно имеют размер порядка 50×50

Статья получена 10 ноября 2025. Данная работа представляет собой результат магистерской диссертации.

Колидов И.А. аспирант кафедры Систем сбора и обработки данных, Новосибирский государственный технический университет, Новосибирск, Россия (e-mail: kolidovivan@yandex.ru).

Бакаев М.А. к.т.н., доцент, зав. кафедрой Систем сбора и обработки данных, Новосибирский государственный технический университет, Новосибирск, Россия (e-mail: bakaev@corp.nstu.ru).

пикселей. Несмотря на достигнутые результаты, этот метод демонстрирует ряд ограничений, включая зависимость от освещения, низкую способность к обобщению между пользователями и необходимостью строгой фиксации положения головы.

Реализация юзабилити-тестирования в реальных условиях требует большей гибкости: пользователи должны чувствовать себя комфортно, не ограничивая естественные движения и позу. Кроме того, модели машинного обучения (ML) должны корректно работать с пользователями, не участвовавшими в обучении, без необходимости индивидуальной калибровки. Важную роль в этом играет репрезентативность и разнообразие данных, поскольку именно они определяют способность модели к обобщению [8, 9]. Однако сбор большого, разнородного и качественного набора данных с участием множества испытуемых является сложной задачей.

В данной работе рассматривается экономичный подход к отслеживанию глаз на основе веб-камеры и сверточной нейронной сети, направленный на преодоление указанных ограничений. Ранее, предварительные результаты были представлены на конференциях PIERE-2024 и IMS-2025. В данной статье мы опираемся на методику, ранее предложенную нами в работе [7], в которой была достигнута среднеквадратическая ошибка (RMSE) 78 пикселей, и развиваем её, интегрируя геометрические признаки в архитектуру CNN. Предполагается, что объединение визуальной информации и геометрических параметров позволит повысить устойчивость модели к изменениям освещения, положению головы и индивидуальным особенностям пользователей.

## II. Методология

В области компьютерного зрения данные играют ключевую роль, выступая основой для обучения, валидации и тестирования моделей ML. Репрезентативность, качество и объем данных напрямую определяют способность алгоритмов к обобщению, а также к их эффективности в реальных сценариях [10]. Современные подходы, такие как глубокое обучение, требуют значительных размеченных данных для выявления сложных паттернов в изображениях [11].

Постановка задачи по разработке усовершенствованного метода сбора данных была следующей: необходимо было разработать специализированное приложение по сбору данных, которое позволит собирать данные со стандартной веб-камеры. Перед проектирование приложения, необходимо было определиться со структурой данных.

### A. *Структура данных*

Для повышения точности определения направления взгляда и уменьшения ошибки, вызванной изменением положения головы, в работе используется мультимодальный подход. Он предполагает подачу на вход модели не только изображений с ключевыми областями лица, но и числовых признаков, характеризующих геометрию глаз и ориентацию головы в пространстве. Такой подход позволяет повысить устойчивость модели к поворотам и наклонам головы, что особенно важно при проведении юзабилити-тестирования в естественных условиях, когда участники не ограничены в движениях.

Вся совокупность данных была разделена на две группы.

*1) Визуальные данные*

Первая группа данных включает изображения лица, положения головы относительно монитора, а также изображения левого и правого глаза.

Выбор размеров изображений обусловлен балансом между сохранением информативности и вычислительной эффективностью. В классических сверточных архитектурах, таких как ResNet [12], VGG [13] и YOLO [14], типичные размеры входных изображений составляют 224×224×3 или 448×448×3. Это связано с особенностями архитектуры: последовательными слоями свертки и пуллинга, которые уменьшают пространственные размеры карт признаков кратно 2. Кроме того, использование изображений меньшего разрешения снижает нагрузку на вычислительные ресурсы при обучении и инференсе модели.

В данной работе использовались следующие размеры изображений (рис. 1):

Изображение лица (224×224×3) – содержит ключевые элементы (глаза, нос, рот), требующие высокого разрешения для корректного распознавания. Выбранный размер кратен 32, что упрощает обработку свёрточными слоями и позволяет использовать transfer learning.

Изображение головы (121×121×3) – обеспечивает анализ общей ориентации и положения головы без необходимости детализировать мелкие элементы. Такое разрешение снижает вычислительные затраты, сохраняя при этом ключевую информацию о позе головы.

Изображения глаз (160×120×3) – сохраняют естественное соотношение сторон глаз, что предотвращает искажения и потерю информации о направлении взгляда. Данное разрешение оптимально для анализа положения зрачков, век и макродвижений глаз при умеренных вычислительных затратах.

Рис. 1. Пример визуальных данных

*2) Геометрические признаки*

Вторая группа данных включает в себя необходимые данные для определения геометрии глаз и положении головы (углы поворота). Для определения геометрии глаза использовались ключевые точки (рис. 2), такие как уголки глаз, верхний и нижний центр век, центр глаза и зрачка, угол ротации.

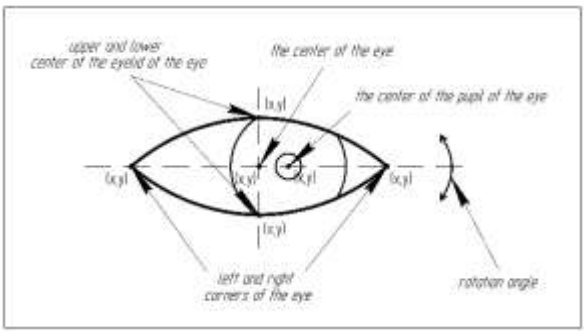


Рис. 2. Геометрические признаки глаза

Для определения ориентации головы была применена задача перспективной n-точки (PnP) [15]. Задача состоит в том, чтобы оценить матрицу поворота $R$ и вектор перемещения $t$ по набору 3D опорных лицевых точек (1) и их 2D проекциям на изображении (2), удовлетворяющих проекционному уравнению (3).

$$P_i = (X_i, Y_i, Z_i) \quad (1)$$

$$p_i = (x_i, y_i) \quad (2)$$

$$s\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = K(RP_i + t) \quad (3)$$

где K – калибровочная матрица камеры, включая фокусное расстояние и координаты основной точки.

Как только матрица вращения R получена, она разлагается на три угла Эйлера:

- рыскание (ψ): поворот головы влево – вправо.
- тангаж (θ): наклон головы вверх – вниз.
- крен (ϕ): боковой наклон.

Полученные значения преобразуются в градусы для упрощения интерпретации. Такой подход позволяет оценить пространственную ориентацию головы по изображениям лица с веб-камеры.

### *B. Сбор и обработка данных*

Для мультимодального подхода была выбран метод «сеточного» сбора данных. Суть метода заключается в том, что экран 1920×1080 делится на 300 ячеек по 64×54 пикселя каждая (рис. 3). Размер ячейки обоснован соответствием физиологии человеческого зрения. Предполагая, что среднее расстояние обзора составляет 50-70 см, ширина ячейки в 64 пикселя соответствует углу обзора примерно 0.6–0.8°. Это соответствует минимальному углу фиксации человеческого глаза [16], что помогает избежать перекрытия. Чрезмерно маленькие размеры ячейки увеличивают разрешение данных, но усложняют обучение CNN из-за избыточной детализации, в то время как чрезмерно большие ячейки снижают способность модели различать близлежащие точки фиксации. Также необходимо учитывать общее количество классов, которое зависит от размера ячейки. Использование 300 классов обеспечивает достаточную емкость для регрессии CNN и снижает риск запоминания моделью определенных позиций.

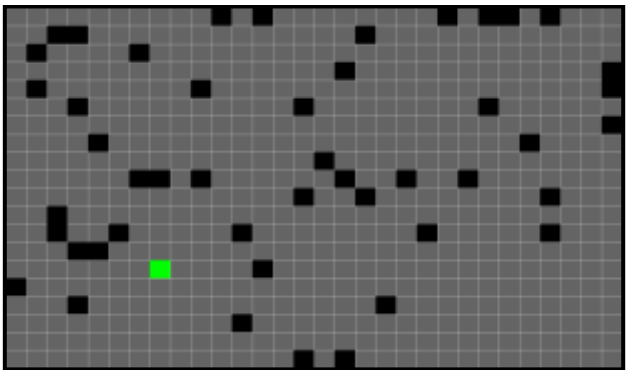

Рис. 3. Сеточный метод сбора данных

Для захвата видеопотока с веб-камеры и извлечения кадров использовалась библиотека OpenCV. Во время сбора данных поток с камеры должен непрерывно считываться и помещаться в очередь. Затем каждый кадр извлекается из очереди и обрабатывается для извлечения всех необходимых признаков. Для обнаружения лица в каждом кадре использовалась кроссплатформенная библиотека от Google, состоящая из нескольких предварительно обученных моделей машинного обучения. Для определения лицевых ориентиров использовался MediaPipe Face Landmarker [17], который позволяет распознавать лица в кадре, выделять области вокруг глаз, а также оценивать геометрию глаз и углы поворота головы.

Для сбора данных использовалась стандартная веб-камера с разрешением 720p (HD). Для дополнительного освещения использовалась кольцевая лампа с холодным спектром (холодный белый LED, 6,000 Кельвинов).

В результате был собран мультимодальный набор данных, содержащий 217,164 изображений (54,291 для каждого типа изображения), а также 54,291 записей геометрии глаз и положения головы в формате JSON. Общий размер набора данных составляет 1.5 ГБ, который охватывает 300 координат экрана.

Собранный мультимодальный датасет формировался в условиях постоянного освещения, что обеспечивает стабильность яркостных характеристик изображений, однако снижает их разнообразие. Кроме того, объём исходных данных (54,291 изображений для каждой категории) оказался недостаточным для эффективного обучения модели и предотвращения переобучения. Для решения этих проблем были применены методы аугментации и нормализации данных, направленные на расширение выборки и повышение устойчивости модели к различным условиям съёмки.

Аугментация использовалась для синтетического увеличения количества изображений и расширения их вариативности по освещению и цветовым характеристикам. Так как при сборе данных на каждую точку фиксации выделялось около трёх секунд, за это время формировалась серия из 85-89 изображений. Для каждого изображения дополнительно применялись фотометрические преобразования, имитирующие различные условия освещения и съемки.

Под фотометрической аугментацией подразумеваются операции изменения яркости, контрастности, насыщенности и оттенка, а также добавление случайного

шума. Эти преобразования повышают обобщающую способность модели, позволяя ей корректно работать в условиях изменяемого освещения и фона.

Параметры аугментации подбирались экспериментально. Примеры изображений после аугментации показаны на рис. 4.

Рис. 4. Пример аугментации данных

Нормализация выполнялась для стандартизации диапазонов значений пикселей и ускорения сходимости модели во время обучения. Она включала несколько этапов:

Изменение размера. Для каждого типа изображений выполнялась проверка соответствия целевым размерам (224×224×3, 121×121×3, 160×120×3).

Преобразование в тензор. Изображения конвертировались из формата PIL в тензоры PyTorch с плавающей точкой, при этом значения пикселей масштабировались из диапазона [0, 255] в [0, 1].

Статистическая нормализация. Для каждого цветового канала вычиталось среднее значение и делилось на стандартное отклонение, рассчитанное по всему набору данных. Это обеспечивало единообразие распределений признаков и способствовало стабильности градиентного спуска при обучении нейронной сети.

### *C. Мультимодальная модель с геометрическими признаками*

В рамках исследования была разработана гибридная мультимодальная нейронная сеть. Для обработки изображений предполагалось использовать 4 отдельные ветви CNN (для изображений лица, положения головы, левого и правого глаз) для извлечения признаков из изображений. Для обработки числовых данных – 3 отдельные ветви полносвязной сети с активацией ReLU (для геометрии левого и правого глаз, положения головы). Для слияния модальностей использовалась конкатенация векторов от данных изображений и геометрических данных (объединенный вектор). Также, использовались дополнительные полносвязанные слои после объединения признаков и выходной слой с двумя нейронами.

#### *1) Подбор гиперпараметров*

Гиперпараметры [18], [19] – это параметры модели, которые определяют ее общую структуру и способность к обучению. Эти параметры устанавливаются перед началом обучения модели и остаются фиксированными во время обучения. Гиперпараметры влияют на то, как модель обучается, какие функции учитываются и какие ограничения накладываются в процессе обучения.

В рамках данной работы, был выполнен подбор гиперпараметров модели с целью оптимизации её производительности. Для этого использовалась библиотека Ray Tune [19], [20] с использованием алгоритма ASHA (Asynchronous Successive Halving Algorithm) [21], который позволяет эффективно искать оптимальные значения гиперпараметров, минимизируя время на выполнение.

Был исследован широкий диапазон значений гиперпараметров, чтобы лучше понять пространство поиска:

**Обучение**

- `seed`: от 1 до 1000.
- `lr`: от $10^{-5}$ до $10^{-3}$.
- `bs`: 16, 32, 64.

**CNN-ветви**

*Изображение лица*

- Количество сверточных слоев (`n_conv_face`): 2, 3, 4, 5.
- Начальное количество фильтров (`n_filters_face`): 16, 32, 64.
- Размер фильтра (`kernel_face`): 3, 5, 7.

*Положение лица*

- Количество сверточных слоев (`n_conv_face_pos`): 1, 2.
- Первоначальное количество фильтров (`n_filters_face_pos`): 16, 32.
- Размер фильтра (`kernel_face_pos`): 3, 5.

*Изображения глаз*

- Количество сверточных слоев (`n_conv_eye`): 2, 3, 4, 5.
- Начальное количество фильтров (`n_filters_eye`): 16, 32, 64.
- Размер фильтра (`kernel_eye`): 3, 5.

**Табличные ветви**

*Геометрия глаза*

- Скрытые слои (`eye_geom_hidden`): [32, 32] или [64, 32].
- Выходной слой (`eye_geom_out`)): [32, 64].

*Положение головы*

- Скрытые слои (`head_geom_hidden`): [32] или [64].
- Выходной слой (`head_geom_out`)): [16, 32].

**Регрессор**

- Количество нейронов в плотном слое (`dense_nodes`): 128, 256, 512.
- Регуляризация отсева (dropout): от 0.1 до 0.5.

Для широкого диапазона значений было протестировано 100 различных конфигураций гиперпараметров. Для каждой конфигурации проводилось обучение модели в течении 10 эпох, после чего оценивалась функция потерь на валидационном наборе данных. На основе значений функции потерь

(val_loss) на валидационном наборе определялись наиболее успешные конфигурации гиперпараметров. Посмотрев на графики обучения и валидации (рис. 5), можно увидеть в действии алгоритм ASHA, который удаляет неэффективные конфигурации, чтобы сэкономить время.

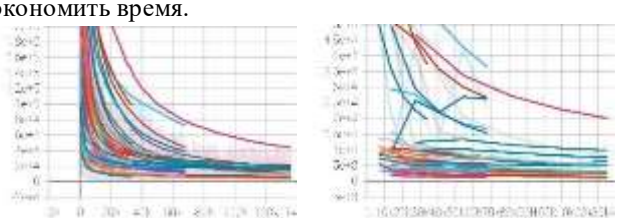

Рис. 5. Графики обучения и валидации при подборе гиперпараметров

Подбор гиперпараметров проходил с использованием GPU (NVIDIA Quadro RTX 5000). Общее время подбора составило 226 часов. По результатам подбора гиперпараметров, был подобран наилучший диапазон параметров, который использовался для обучения модели.

*2) Архитектура мультимодальной модели*

Модель глубокого обучения, используемая для предсказания точки фиксации взгляда пользователя на экране, представляет собой мультимодальную модель, объединяющую обработку визуальных данных (изображения лица, положения головы и глаз) и структурированных метаданных (геометрия глаз, положение головы). Архитектура включает в себя параллельные сверточные ветви для извлечения пространственных признаков из изображений; многослойные персептроны (multilayer perceptron, MLP) необходимые для обработки табличных данных, а также блок слияния с последующей регрессией для предсказания координат точки фиксации взгляда пользователя на экране (рис. 6).

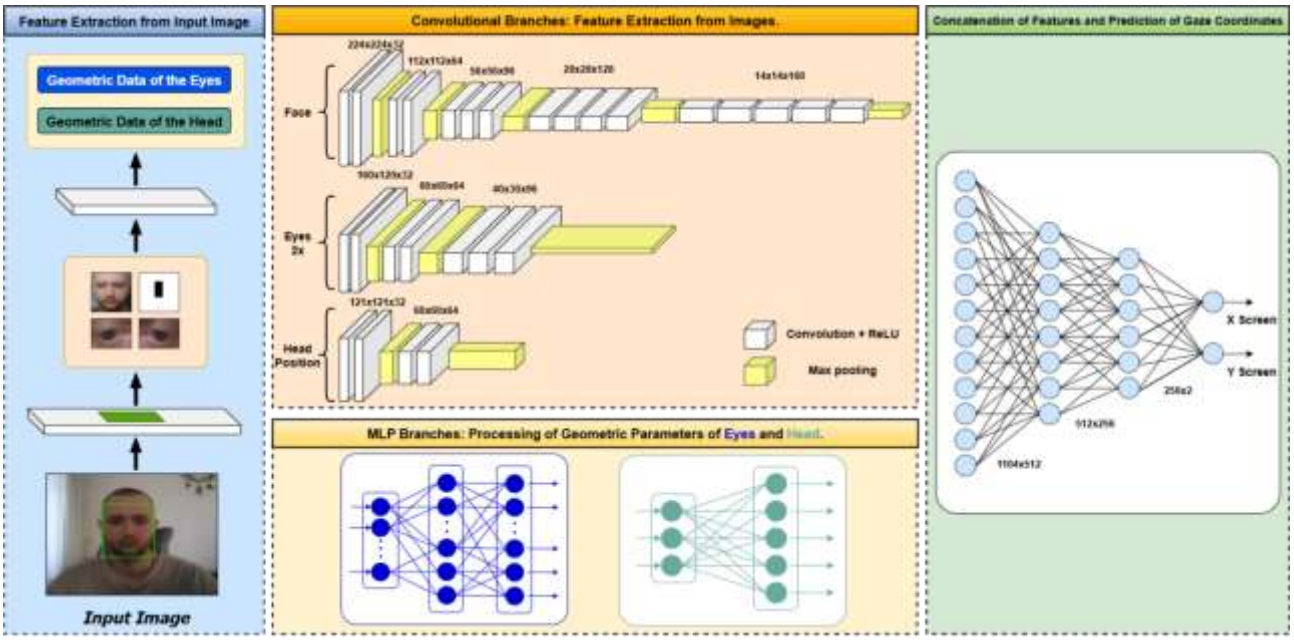


Рис. 6. Архитектура предлагаемой мультимодальной модели отслеживания движения глаз. Входные данные состоят из изображений лица, положения головы и глаз, а также геометрических параметров глаз и головы. Сверточные ветви извлекают визуальные характеристики, в то время как ветви MLP обрабатывают геометрические параметры. Все вектора объединяются и проходят через блок регрессии, чтобы предсказать координаты взгляда (x, y) на экране

- Для обработки изображений используются четыре параллельных ответвления. Все изображения обрабатываются с помощью сверточных блоков (ConvBranch), каждый из которых состоит из следующей последовательности слоев:
- Conv2d – свертка с заполнением p=k/2 для сохранения пространственного разрешения входного изображения перед объединением.
- BatchNorm2d – пакетная нормализация, которая вычисляет среднее значение и стандартное отклонение для каждого канала.
- ReLU – функция активации.
- MaxPool2d – максимальное объединение с ядром 2×2 и шагом 2, которое уменьшает размер карты объектов вдвое.

После настройки гиперпараметров ветвь изображения лица была сконфигурирована с 5 слоями с использованием ядра 5×5 с начальными количеством фильтров 32 (32 → 64 → 96 → 128 → 160). Ветвь положения головы была сконфигурирована с 2 слоями, использующими ядра 3×3, с начальным количеством фильтров 32 (32 → 64). Ветви изображений глаз были сконфигурированы из 3 слоев с использованием ядра 3×3, с начальными количеством фильтров 32 (32→ 64 → 96).

Для обработки числовых параметров используются специализированные полносвязные слои. Каждый слой спроектирован для нелинейного преобразования входных параметров в вектор признаков.

Для данных о геометрии глаз используется двухслойная сеть:

- Первый полносвязный (Linear) слой имеет на входе 13 параметров и на выходе 32 нейрона. На данном слое применяется инициализация весов HeNormal, а также функция активации ReLU.

- Второй полносвязный (Linear) слой имеет на входе 32 нейрона, и выдает 32 нейрона. Данный слой имеет функцию активации ReLU.

Для данных о положении головы используется однослойная сеть:

- Полносвязный (Linear) слой имеет на входе 3 параметра и на выходе 32 нейрона. На данном слое применяется инициализация весов HeNormal, а также функция активации ReLU.

Для объединения признаков всех ветвей используется блок слияния, в котором, с помощью Global Average Pooling преобразуются карты признаков всех ветвей в вектор. Линейные слои после GlobalAvgPool для изображений проецируют признаки в фиксированный размер 256. Далее идут полносвязные слои, которые уменьшают количество признаков до 2 нейронов на выходе.

## III. ОЦЕНКА МОДЕЛИ

### *D. Обучение мультимодальной модели*

Для создания модели, способной предсказывать координаты фиксации взгляда пользователя на экране, был использован фреймворк машинного обучения PyTorch [18].

Среднеквадратичная ошибка (MSE) [22] использовалась в качестве функции потерь. MSE измеряет среднеквадратичную разницу между фактическими и прогнозируемыми значениями (4).

$$MSE = \frac{1}{n} \times \sum_{i=1}^{n} \begin{pmatrix} \left(x_{true,i} - x_{pred,i}\right)^2 + \\ +\left(y_{true,i} - y_{pred,i}\right)^2 \end{pmatrix} \quad (4)$$

Чем меньше значение MSE, тем лучше модель предсказывает фактические данные. Для оценки модели использовалась метрика RMSE [23], представляющая собой квадратный корень из MSE (5). RMSE можно интерпретировать как попиксельное расстояние между прогнозируемыми и истинными координатами фиксации взгляда.

$$RMSE = \sqrt{\frac{1}{n} \times \sum_{i=1}^{n} \begin{pmatrix} \left(x_{true,i} - x_{pred,i}\right)^2 + \\ +\left(y_{true,i} - y_{pred,i}\right)^2 \end{pmatrix}} \quad (5)$$

После фотометрической аугментации и нормализации набор данных содержит 2,171,640 изображений. Данные были разделены по принципу 80/10/10, при котором 80% набора данных использовалось для обучения, а оставшиеся 10% – для валидации и тестирования.

Разработанная мультимодальная модель была обучена на собранном наборе данных за 150 эпох на графическом процессоре (NVIDIA Quadro RTX 5000). Обучение заняло 47 часов, а среднеквадратичная ошибка (MSE) на тестовом наборе составила 249.25, что соответствует 15.8 пикселям экрана (Рис. 7).

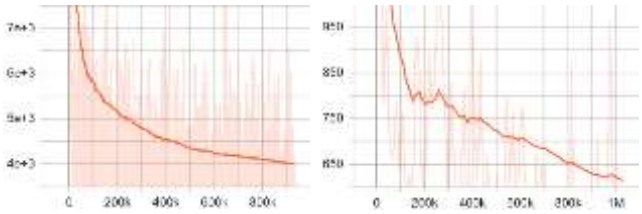

Рис. 7. Графики обучения и валидации мультимодальной модели

### *E. Тестирование и сравнение модели*

Для оценки точности модели прогнозировать координаты фиксации взгляда пользователя на экране был проведен контролируемый эксперимент. На экране, с разрешением 1920×1080, по заранее заданной траектории, охватывающей весь экран, перемещался маркер, за которым следил пользователь.

Во время эксперимента, видеопоток с веб-камеры захватывался с помощью OpenCV, а необходимые данные для модели извлекались с помощью MediaPipe. Всего за время сеанса было собрано 3,200 тестовых образцов.

Для оценки модели, каждая запись из собранного тестового набора данных, подавалась в модель. Модель была развернута на процессоре (AMD Ryzen 5 3600 / Intel Core i5-10400F). Входные данные обрабатывались моделью, которая выводила прогнозируемые координаты взгляда пользователя на экране. Прогнозируемые координаты сохранялись для дальнейшего анализа. Точность модели оценивалась с использованием метрики RMSE.

Результаты тестирования модели показывают, что среднее время инференса модели составило 15.24 мс, что соответствует 65 кадрам в секунду (FPS = 1000/15.24). Что указывает на то, что модель способна работать в режиме реального времени.

Инициализация модели потребляет 310 МБ оперативной памяти, что очень эффективно и удобно для встраивания модели в легкие приложения. Средняя загрузка процессора составила 23.7%, что указывает на отсутствие значительной нагрузки на центральный процессор.

В ходе тестирования модель достигла среднего значения RMSE в 30.23 пикселей (Рис. 8). По сравнению с предыдущим решением (RMSE ~ 78 пикселей), это представляет собой снижение средней погрешности примерно на 61% (коэффициент улучшения 2.6×), что указывает на значительный прогресс в точности при сохранении экономичной реализации с использованием стандартной веб-камеры.

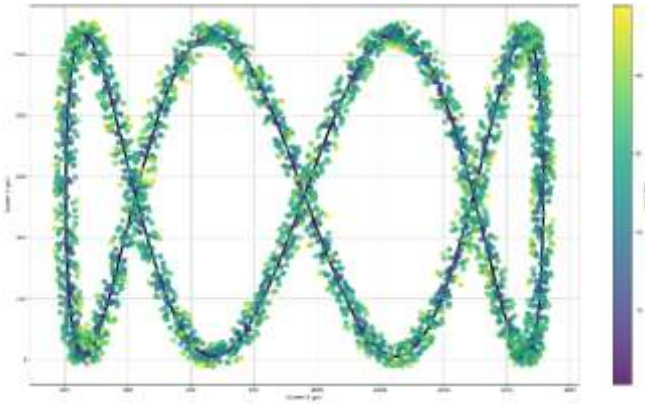

Рис.8. Результаты тестирования модели

Для сравнения, точность профессионального оборудования для отслеживания глаз составляет примерно 11.86 – 23.71 пикселей для EyeLink 1000 Plus и 23.71 – 47.43 пикселей для AdHawk MindLink [16]. Таким образом, по средней погрешности наша система занимает промежуточное положение между потребительским и профессиональным оборудованием. Это демонстрирует, что в контексте практических задач UX предлагаемое решение является конкурентоспособным и обеспечивает экономичную альтернативу дорогостоящим устройствам.

### *F. Построение тепловых карт с использованием обученной модели*

Тепловые карты являются важным инструментом для анализа взаимодействия пользователя с продуктом. Такие карты наглядно показывают какие области веб-страницы привлекают внимание пользователя, а какие остаются без него.

Был проведен эксперимент, суть которого проверить пригодность предсказаний предлагаемой мультимодальной модели для построения тепловой карты визуального внимания пользователя при естественно просмотре веб-страницы. Просмотр проводился без целевых инструкций: участник свободно изучал страницу прототипа сайта магазина одежды.

Для построение тепловой карты использовалась гистограмма плотности (оценка плотности с гауссовым ядром), далее данные нормализовались к диапазону [0, 1] после чего применялось цветовое кодирование через градиент от синего (минимальной активности) к красному максимальная активность) цвета.

Тепловая карта, построенная по предсказаниям модели (Рис. 9 и Рис. 10), продемонстрировала адекватное распределение визуального внимания пользователя по странице: плотность концентрируется вокруг ожидаемых зон интерфейса: карточка товара, навигация сайта, поиск. Полученные карты пригодны для прикладного юзабилити-тестирования и позволяют выделять AOI без применения специализированного оборудования.

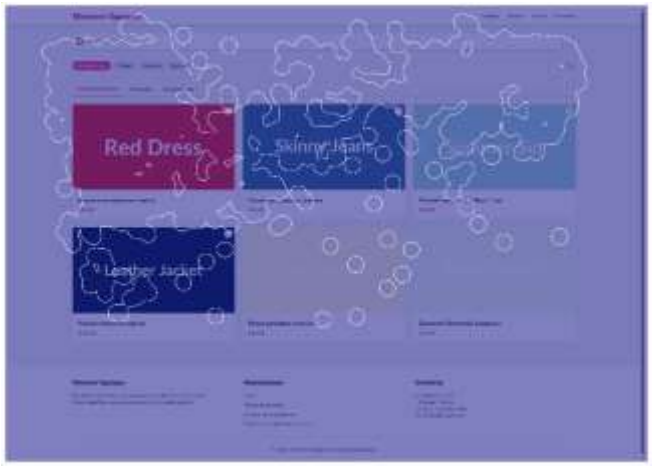


Рис.9. Визуализация распределения взгляда пользователя: контуры взгляда на странице.

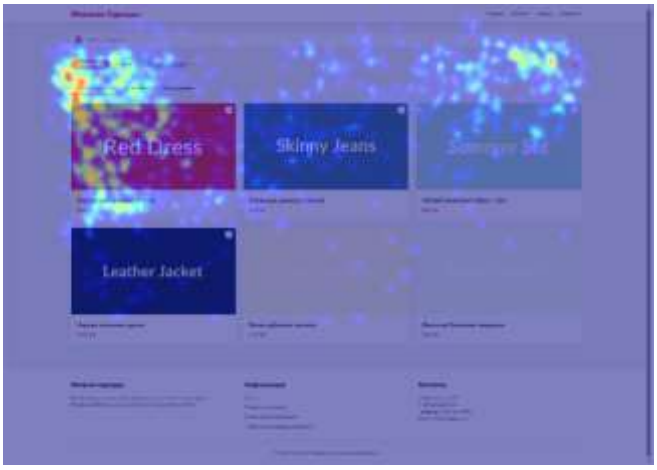


Рис. 10. Визуализация распределения взгляда пользователя: тепловая карта интенсивности фиксации взгляда.

## IV. Заключение

В статье представлен практический и воспроизводимый подход к бюджетному отслеживанию взгляда пользователя на основе стандартной веб-камеры и мультимодальной сверточной сети. Разработанная модель совмещает как визуальные (изображение лица/глаз), так и геометрические (геометрия глаз, углы поворота головы) признаки.

Экспериментальная проверка показала, что предложенные изменения привели к заметному улучшению точности по сравнению с предыдущей версией модели.

Результаты разработки мультимодальной архитектуры модели показали среднюю RMSE 30.23 пикселя, что на 61% лучше предыдущего результата (RMSE 78 пикселя). Также, данная модель показала хорошую производительность: среднее время инференса составляет 15.23 ms, что позволяет работать модели с FPS 65 кадров в секунду.

В сравнении с профессиональными системами, по средней ошибке наша система занимает промежуточное положение между значениями профессионального оборудования:

- EyeLink 1000 Plus: Accuracy ≈ 11.86-23.71 пикселей;

- AdHawk MindLink: Accuracy ≈ 23.71-47.43 пикселей.

Данный подход отлично подходит для задач юзабилити-тестирования, построения тепловых карт и агрегированного анализа внимания, где важны пространственные паттерны, а не микро-попадания с субпиксельной точностью.

Предложенная мультимодальная система демонстрирует, что сочетание изображений и геометрии лица пользователя позволяет существенно повысить точность отслеживания глаз через веб-камеру. Что делает его практичным для задач UX-аналитики при минимальных затратах.

## Библиография


[1] Moran K. Usability (User) Testing 101. 2019. URL: https://www.nngroup.com/articles/usability-testing-101/ (дата обращения: 10.11.2025).

[2] Banuelos-Lozoya E., Gonzalez-Serna G., Gonzalez-Franco N., Fragoso-Diaz O., Castro-Sanchez N. A systematic review for cognitive state-based QoE/UX evaluation // Sensors. 2021. No. 21(10). Article 3439.

[3] Onur A., Yang Y. Using eye trackers for usability evaluation of health information technology: a systematic literature review // JMIR human factors. 2015. No. 2.1. Article e4062.

[4] Szekely D., Vert S., Rotaru O., Andone D. Usability evaluation with eye tracking: The case of a mobile augmented reality application with historical images for urban cultural heritage // Heritage. 2023. No. 6(3). P. 3256-3270.

[5] Cornelissen F. W., Peters E. M., Palmer J. The Eyelink Toolbox: eye tracking with MATLAB and the Psychophysics Toolbox // Behavior Research Methods, Instruments, & Computers. 2002. Vol. 34. No. 4. P. 613-617.

[6] Niehorster D. C., Hessels R. S., Benjamins J. S. GlassesViewer: Open-source software for viewing and analyzing data from the Tobii Pro Glasses 2 eye tracker // Behavior Research Methods. 2020. Vol. 52. No. 3. P. 1244-1253.

[7] Kolidov I. A., Bakaev M. V. Discount Eye-Tracking for Usability Testing with Webcam, CNN, and the Selected Training Data // 2024 IEEE 3rd International Conference on Problems of Informatics, Electronics and Radio Engineering (PIERE). IEEE, 2024. P. 680-685.

[8] Davila F., Paz F., Moquillaza A. Usage and application of heatmap visualizations on usability user testing: A systematic literature review // International Conference on Human-Computer Interaction. Cham: Springer, 2023. P. 3-17.

[9] García M., Cano S. Eye tracking to evaluate the user experience (UX): Literature review // International Conference on Human-Computer Interaction. Cham: Springer International Publishing, 2022. P. 134-145.

[10] LeCun Y., Bengio Y. Hinton G. Deep learning // Nature. 2015. Vol. 521. No. 7553. P. 436-444.

[11] Deng J., Dong W., Socher R., L. Li J., Li K., Fei-Fei L. Imagenet: A large-scale hierarchical image database // IEEE conference on computer vision and pattern recognition. 2009. P. 248-255.

[12] He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. P. 770-778.

[13] Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition // arXiv preprint. arXiv:1409.1556, 2014.

[14] Redmon J., Divvala S., Girshick R., Farhadi A. You only look once: Unified, real-time object detection // Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. P. 779-788.

[15] Rocca F., Mancas M., Gosselin B. Head pose estimation by perspective-n-point solution based on 2d markerless face tracking // International Conference on Intelligent Technologies for Interactive Entertainment. Cham: Springer International Publishing, 2014. P. 67-76.

[16] Holmqvist K., Örbom S. L., Hooge I. T., Niehorster D. C., Alexander R. G., Andersson R., Hessels R. S. Eye tracking: empirical foundations for minimal reporting guideline // Behavior research methods. 2023. Vol. 55. No. 1. P. 364-416.

[17] Kartynnik Y., Ablavatski A., Grishchenko I., Grundmann M. Real-time facial surface geometry from monocular video on mobile GPUs // arXiv preprint. 2019. arXiv:1907.06724.

[18] Kadhim Z. S., Abdullah H. S., Ghathwan K. I. Artificial Neural Network Hyperparameters Optimization: A Survey // Int. J. Online Biomed. Eng. 2022. Vol. 18. No. 15l .P. 59-87.

[19] Bartz-Beielstein T. PyTorch Hyperparameter Tuning-A Tutorial for spotPython // arXiv preprint. 2013. arXiv:2305.11930.

[20] Polyakov D. N., Stepanova M. M. Hyperparameter tuning of neural network for high-dimensional problems in the case of Helmholtz equation // Moscow University Physics Bulletin. 2023. Vol. 78. No. Suppl 1. P. S243-S255.

[21] Schmucker R., Donini M., Zafar M. B., Salinas D., Archambeau C. Multi-objective asynchronous successive halving // arXiv preprint. 2021. arXiv:2106.12639.

[22] Das K., Jiang J., Rao J. N. K. Mean squared error of empirical predictor. 2004. P. 818-840.

[23] Hodson T. O. Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not // Geoscientific Model Development Discussions. 2022. P. 1-10.



Колидов И.А. аспирант кафедры Систем сбора и обработки данных, Новосибирский государственный технический университет, Новосибирск, Россия (e-mail: kolidovivan@yandex.ru).

Бакаев М.А. к.т.н., доцент, зав. кафедрой Систем сбора и обработки данных, Новосибирский государственный технический университет, Новосибирск, Россия (e-mail: bakaev@corp.nstu.ru).

# Enhancement of the Oculographic Spatial Monitoring Method Using a Multimodal CNN with Geometric Features

Ivan Kolidov, Maxim Bakaev

***Abstract*—this paper presents a cost-efficient and reproducible approach to eye-tracking using a standard webcam and a multimodal convolutional neural network (CNN). Unlike traditional eye-tracking systems (such as EyeLink or Tobii), which require expensive hardware, the proposed solution relies on a combination of visual and geometric features, as well as a specially designed neural architecture that integrates four convolutional branches for image processing and three fully connected branches for numerical data processing. The method also mitigates common limitations related to lighting conditions, head position, and individual user characteristics.**

**A multimodal dataset of 10 GB was created for model training, containing 2 million eye images and corresponding geometric annotations. After hyperparameter optimization using Ray Tune and the ASHA algorithm, our model achieved a mean RMSE error of 30.23 pixels, representing a 61% improvement over the previous version of the method. The inference time was 15.2ms (≈65 FPS), making the system suitable for real-time applications.**

**Comparison with professional eye-trackers (EyeLink 1000 Plus, AdHawk MindLink) showed that the proposed model achieves intermediate accuracy while requiring significantly less expensive equipment. The obtained results demonstrate the potential of multimodal neural networks for oculography in usability testing, UX analytics, and human–computer interaction research.**

***Keywords*—eye-tracking; multimodal neural network; webcam; usability testing; geometric features.**

## REFERENCES

[1] K. Moran, Usability (User) Testing 101. 2019. Available: https://www.nngroup.com/articles/usability-testing-101/.

[2] E. Banuelos-Lozoya, G. Gonzalez-Sema, N. Gonzalez-Franco, O. Fragoso-Diaz and N. Castro-Sanchez, "A systematic review for cognitive state-based QoE/UX evaluation," *Sensors*, No. 21(10), article 3439, 2021.

[3] A. Onur and Y. Yang, "Using eye trackers for usability evaluation of health information technology: a systematic literature review," *JMIR human factors*, No. 2.1, article e4062, 2015.

[4] D. Szekely, S. Vert, O. Rotaru and D. Andone, "Usability evaluation with eye tracking: The case of a mobile augmented reality application with historical images for urban cultural heritage," *Heritage*, No. 6(3), pp. 3256-3270, 2023.

[5] W. Cornelissen, E. M. Peters and J. Palmer, "The Eyelink Toolbox: eye tracking with MATLAB and the Psychophysics Toolbox," *Behavior Research Methods, Instruments, & Computers*, Vol. 34, No. 4, pp. 613-617, 2002.

[6] D. C. Niehorster, R. S. Hessels and J. S. Benjamins, "Glasses Viewer: Open-source software for viewing and analyzing data from the Tobii Pro Glasses 2 eye tracker," Behavior Research Methods, Vol. 52, No. 3, pp. 1244-1253, 2020.

[7] I. Kolidov and M. Bakaev, "Discount Eye-Tracking for Usability Testing with Webcam, CNN, and the Selected Training Data," In *2024 IEEE 3rd International Conference on Problems of Informatics, Electronics and Radio Engineering (PIERE)*, IEEE, pp. 680-685, 2024.

[8] F. Davila, F. Paz and A. Moquillaza, "Usage and application of heatmap visualizations on usability user testing: A systematic literature review," In *International Conference on Human-Computer Interaction*, Cham, Springer, pp. 3-17, 2023.

[9] M. García and S. Cano, "Eye tracking to evaluate the user experience (UX): Literature review," In International Conference on Human-Computer Interaction, Cham, Springer International Publishing, pp. 134-145, 2022.

[10] Y. LeCun and Y. Bengio, "Hinton G. Deep learning," *Nature*, Vol. 521, No. 7553, pp. 436-444, 2015.

[11] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," In *IEEE conference on computer vision and pattern recognition*, pp. 248-255, 2009.

[12] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778, 2016.

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint*, arXiv:1409.1556, 2014.

[14] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You only look once: Unified, real-time object detection," In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779-788, 2016.

[15] F. Rocca, M. Mancas and B. Gosselin, "Head pose estimation by perspective-n-point solution based on 2d markerless face tracking," In *International Conference on Intelligent Technologies for Interactive Entertainment*, Cham, Springer International Publishing, pp. 67-76, 2014.

[16] Holmqvist, S. L. Örbom, I. T. Hooge, D. C. Niehorster, R. G. Alexander, R. Andersson and R. S. Hessels, "Eye tracking: empirical foundations for a minimal reporting guideline," *Behavior research methods*, Vol. 55, No. 1, pp. 364-416, 2023.

[17] Y. Kartynnik, A. Ablavatski, I. Grishchenko and M. Grundmann, "Real-time facial surface geometry from monocular video on mobile GPUs", *arXiv preprint*, arXiv:1907.06724, 2019.

[18] Z. S. Kadhim, H. S. Abdullah and K. I. Ghathwan, "Artificial Neural Network Hyperparameters Optimization: A Survey," *Int. J. Online Biomed. Eng*, Vol. 18, No. 15l, pp. 59-87, 2022.

[19] T. Bartz-Beielstein, "PyTorch Hyperparameter Tuning-A Tutorial for spotPython", *arXiv preprint*, arXiv:2305.11930, 2023.

[20] D. N. Polyakov and M. M. Stepanova, "Hyperparameter tuning of neural network for high-dimensional problems in the case of Helmholtz equation," *Moscow University Physics Bulletin*, Vol. 78, No. Suppl 1, pp. S243-S255, 2023.

[21] R. Schmucker, M. Donini, M. B. Zafar, D. Salinas and C. Archambeau, "Multi-objective asynchronous successive halving," *arXiv preprint*, arXiv:2106.12639, 2021.

[22] K. Das, J. Jiang and J. N. K. Rao, Mean squared error of empirical predictor, pp. 818-840, 2004.

[23] T. O. Hodson, "Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not," *Geoscientific Model Development Discussions*, pp. 1-10, 2022.

Kolidov I.A., Postgraduate Student of the Department of Data Collection and Processing Systems, Novosibirsk State Technical University, Novosibirsk, Russia (e-mail: kolidovivan@yandex.ru

Bakaev M.A., Ph.D., Professor (Associate), Head of the Department of Data Collection and Processing Systems, Novosibirsk State Technical University, Novosibirsk, Russia (e-mail: bakaev@corp.nstu.ru).