

УДК 004.5

Построение эффективных сложных информационных моделей на основе трекинга глаз

Савельев Алексей Александрович

savelevvaa@mail.ru

SPIN-код: 4905-1602

Горячкин Борис Сергеевич^(*)

bsgor@mail.ru

SPIN-код: 5465-3012

МГТУ им. Н.Э. Баумана, Москва, Россия

Аннотация. Статья посвящена исследованию возможностей системы отслеживания глаз как инструмента для анализа акцентов внимания и последующего построения или улучшения, на основе полученных данных, эффективных сложных информационных моделей. Описываемые алгоритмы направлены на улучшение точности интерпретации поведения взгляда и понимания зон интереса пользователя. Применение новых методов обработки позволяет повысить точность расчета координат точек взгляда и гибкость визуализации полученных результатов, что, в свою очередь, способствует более детализированному анализу поведения пользователей при восприятии сложных графических изображений.

Ключевые слова: зрение, взгляд, отслеживание, интерполяция, Python, диаграмма рассеивания, тепловая карта

Building effective complex information models based on eye tracking

Savelyev Alexey Alexandrovich

savelevvaa@mail.ru

SPIN-code: 4905-1602

Goryachkin Boris Sergeevich^(*)

bsgor@mail.ru

SPIN-code: 5465-3012

BMSTU, Moscow, Russia

Abstract. The article explores the capabilities of eye-tracking systems as a tool for analyzing attention focus and subsequently constructing or enhancing effective complex information models based on the obtained data. The described algorithms aim to improve the accuracy of gaze behavior interpretation and the understanding of users' areas of interest. The application of novel processing methods enhances the precision of calculating gaze point coordinates and the flexibility of visualizing the results, thereby enabling a more detailed analysis of user behavior when perceiving complex graphical images.

Keywords: vision, gaze, tracking, interpolation, Python, scatter plot, heatmap

Введение. Современные информационные системы все чаще сталкиваются с задачей обработки и представления сложных данных таким образом, чтобы они были интуитивно понятны и полезны для пользователя. Но эффективность таких систем во многом зависит от того, насколько точно удастся

учесть особенности человеческого восприятия в графически изображаемых информационных моделях. В этой сфере технологии отслеживания взгляда открывают новые горизонты, позволяя не только фиксировать, куда направлен взгляд, но и анализировать, как человек взаимодействует с визуальной информацией.

Движения глаз — это не просто случайные траектории, а отражение когнитивных процессов, предпочтений и зон интереса. Использование данных о поведении взгляда для построения или оптимизации сложных информационных моделей становится перспективным направлением, объединяющим психологию, программирование и дизайн интерфейсов. Однако реализация таких систем сталкивается с рядом вызовов: от неточности расчетов координат взгляда до необходимости адаптации результатов под динамические условия восприятия.

Настоящее исследование посвящено разработке и совершенствованию алгоритмов трекинга глаз [1] на основе открытых библиотек Python, таких как OpenCV и Mediapipe, с целью создания эффективных информационных моделей. Особое внимание уделено преодолению ключевых ограничений базовых трекеров, таких как разброс координат взгляда, вызванный шумом данных, и недостаточная точность при изменении внешних условий. В работе предлагаются методы фильтрации шума, включая скользящее среднее и экспоненциальное сглаживание, а также новый подход к расчету зоны внимания пользователя. Эти улучшения направлены на повышение точности анализа и визуализации данных, что делает модели более применимыми для решения практических задач — от оптимизации интерфейсов до изучения поведения пользователей при работе со сложными графическими изображениями.

Материалы и методы. Исходный функционал трекера реализован на Python [2] с OpenCV [3] и рассчитывает взгляд, интерполируя координаты зрачков в координаты на экране с помощью двух калибровочных координат — граничных значений зрачков (1):

$$X_{\text{screen}} = \frac{(X_{\text{gaze}} - X_{\text{top_left}})}{X_{\text{bottom_right}} - X_{\text{top_left}}} \cdot w, \quad Y_{\text{screen}} = \frac{(Y_{\text{gaze}} - Y_{\text{top_left}})}{Y_{\text{bottom_right}} - Y_{\text{top_left}}} \cdot h, \quad (1)$$

где $X_{\text{screen}}, Y_{\text{screen}}$ — рассчитываемые координаты взгляда; $X_{\text{gaze}}, Y_{\text{gaze}}$ — координаты зрачков пользователя; $X_{\text{top_left}}, Y_{\text{top_left}}$ — координаты зрачков пользователя, при направлении взгляда в левый верхний угол; $X_{\text{bottom_right}}, Y_{\text{bottom_right}}$ — координаты зрачков пользователя, при направлении взгляда в правый нижний угол; w, h — ширина и высота экрана соответственно.

В базе были организованы подготовка и сохранение данных взгляда для дальнейшего построения по ним диаграммы рассеивания [4, 5] (рис. 1) с полупрозрачными окружностями, отражающими точки взгляда пользователя.

Диаграмма рассеивания в этом случае является аналогом тепловой карты [6, 7], по которой можно понять на какие из объектов на экране [8] смотрел пользователь и определить объекты интереса среди них.

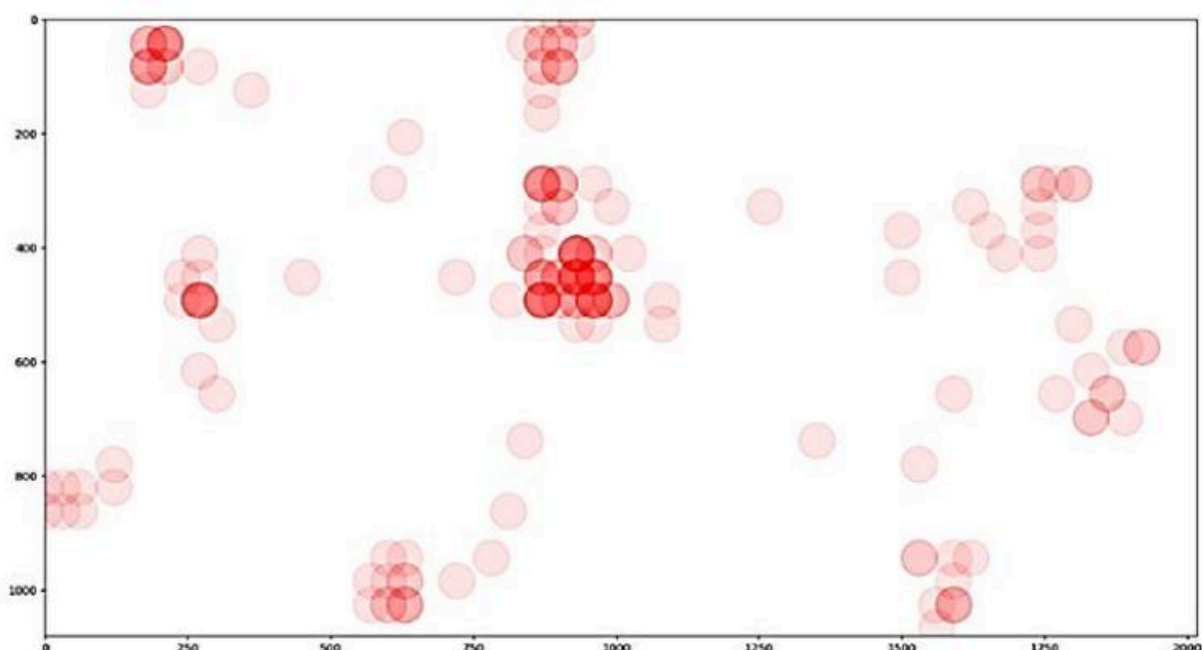


Рис. 1. Диаграмма рассеивания

Координаты зрачков трекер получает от модуля mediapipe [9] в виде сетки точек лица (FaceMesh, рис. 2), из которой выбирались необходимые для расчета. Граничные значения зрачков сохранялись во время калибровки системы. Пользователю предлагалось посмотреть в левый верхний и в правый нижний углы экрана.

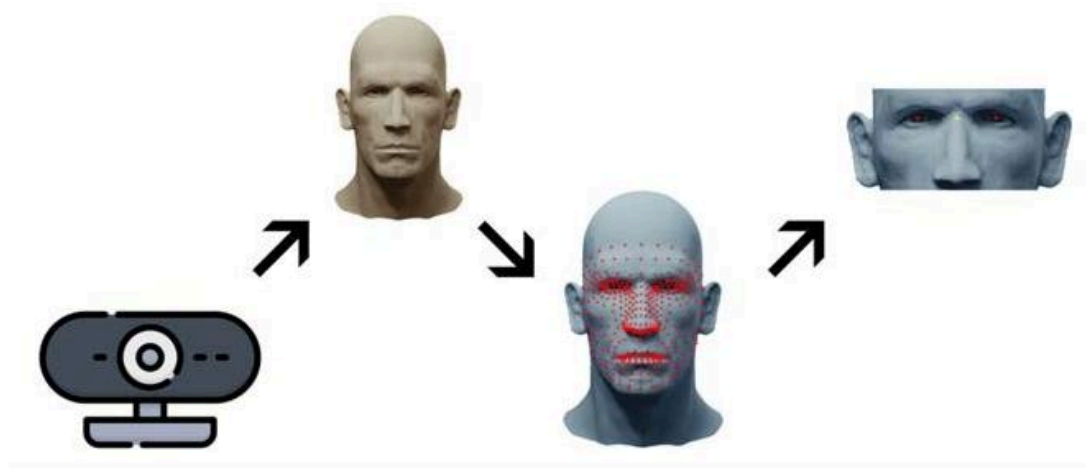


Рис. 2. Общий алгоритм работы системы

Применение шумовых фильтров. Работая, исходный трекер выдавал достаточно большой разброс рассчитанных точек взгляда, имея в конечном счете достаточно низкую точность, которая зависела от качества освещения и разрешения веб-камеры. В качестве решения проблемы разброса точки взгляда

да было решено использовать шумовой фильтр [10], а именно скользящее среднее.

Скользящее среднее (уравнение 2) — это алгоритм фильтрации, который используется для уменьшения шума и сглаживания сигнала. Он работает путем взятия среднего значения заданного количества выборок из входного сигнала и использования этого среднего значения в качестве выходного:

$$M_t = \frac{1}{N} \sum_{i=0}^{N-1} X_{t-i}, \quad (2)$$

где M_t — усредненное значение в момент времени t ; N — количество элементов в очереди (в данном случае 5); X_{t-i} — значение координаты в момент времени $t - i$.

Это означает, что каждое новое значение X_t координаты учитывается с одинаковым весом, а старые значения постепенно «забываются», когда очередь заполняется новыми данными. Это позволяет сгладить обычный небольшой шум и практически нивелировать влияние сильных выбросов. Также для сглаживания перемещения усредненной точки взгляда использовалось экспоненциальное сглаживание.

Экспоненциальное сглаживание (уравнение 3) — это метод, который используется для обработки последовательных данных, таких как, например, временные ряды, с целью уменьшения влияния колебаний этих данных и создания более плавной последовательности. В отличие от простого усреднения, экспоненциальное сглаживание учитывает все предыдущие значения, но при этом новые данные получают больший вес, чем старые. Этот метод часто используется в случаях, когда требуется быстрее реагировать на изменения:

$$S_t = \alpha \cdot X_t + (1 - \alpha) \cdot S_{t-1}, \quad (3)$$

где S_t — сглаженное значение на текущем шаге t ; X_t — новое (не сглаженное) значение на текущем шаге t ; S_{t-1} — сглаженное значение на предыдущем шаге $t - 1$; α — коэффициент сглаживания ($0 < \alpha < 1$). Чем больше значение α , тем больше влияние новых данных на сглаженное значение.

Расчет зоны взгляда. На этом слайде представлен новый внедренный функционал — методика расчета размера зоны взгляда. В основе метода лежит динамическое изменение размера окружности, отражающей зону внимания, в зависимости от разброса координат (рис. 3). Этот подход, при хорошем освещении, позволяет учитывать микросаккады — небольшие, быстрые движения глаз, которые могут влиять на точность фиксации.

Все рассчитанные координаты взгляда, хранящиеся в очередях, используемых для сглаживания скользящим средним, можно использовать для расчета области взгляда, и величина этой области будет зависеть от разброса точек в очередях. Для этого используем формулу разброса, в которой используется стандартное отклонение по каждой из координат. На инфографике видно, как вычисляется средняя точка взгляда (x , y), исходя из среднего зна-

чения координат всех зафиксированных точек очереди. Радиус окружности зоны взгляда рассчитывается как половина величины разброса координат. Формула для разброса включает стандартное отклонение по осям X и Y , что позволяет учитывать вариации положения взгляда. Таким образом, чем больше разброс точек, тем больше размер окружности, что более точно отражает зону внимания пользователя.

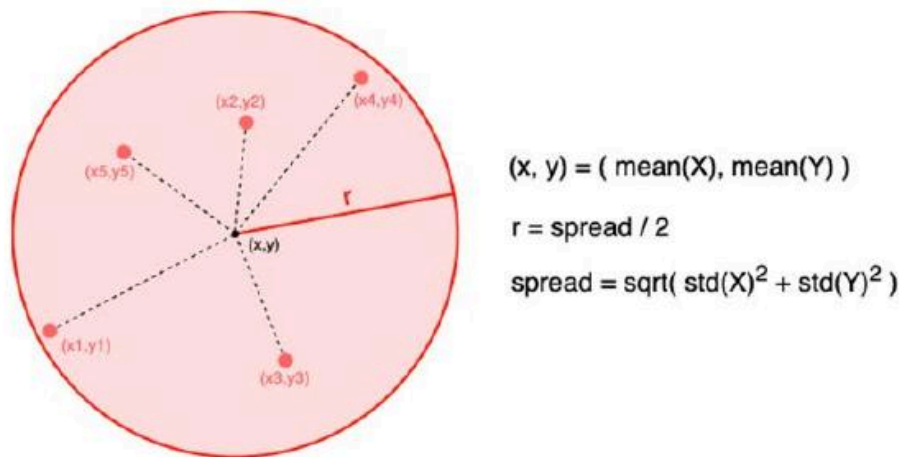


Рис. 3. Расчет области взгляда пользователя

Алгоритм регистрации фиксации взгляда. Фиксация — это состояние, когда взгляд стабильно удерживается в одной области экрана, в отличие от быстрого перемещения взгляда, называемого саккадой. Время фиксации взгляда у человека варьируется в диапазоне от 100 до 600 мс. Для реализации данного функционала пороговое значение для регистрации фиксации было выбрано 300 мс как середина диапазона. Второе ключевое пороговое значение для фиксаций — это максимальный радиус области фиксации. Оно установлено как 150 пикселей — это примерно 1,5 размера от фовеального зрения (1 градуса области зрения) [11], которое вычислили как 101 пикселей для экрана 13 дюймов с разрешением 2560 на 1440 пикселей при удалении глаз на 65 см. Расчет представлен уравнениями

$$\begin{aligned}
 d &= 89,35 \text{ px/cm}; \\
 l &= 2 \cdot 65 \cdot \text{tg}\left(\frac{\theta}{2}\right) = 1,135 \text{ см}; \\
 R &= 89,35 \cdot 1,135 = 101,4 \text{ px},
 \end{aligned} \tag{4}$$

где d — плотность пикселей на сантиметр экрана; l — длина участка фовеального зрения ($\theta = 1$); R — радиус области фиксации взгляда.

Выбор порога в 1,5 размера фовеального зрения объясняется необходимостью учета небольших движений головы и глаз пользователя, которые могут происходить, пока он сфокусирован на одном объекте. Фиксация не обязательно строго ограничена областью в 1° , поскольку даже при удержании взгляда на определенной точке могут происходить малые перемещения

взгляда внутри области. Это своего рода «зона допуска», позволяющая считать фиксацией удержание взгляда на области.

На рис. 4 представлен пример расчета фиксации взгляда. В случае, если фиксация взгляда перешагнула за пороговое значение времени, начинаем отслеживание фиксации, отслеживания как именно взгляд вел себя. Чем больше разброс точек взгляда внутри фиксации, тем больше будет ее окружность в последующем на визуализации. Чем дольше длилась фиксация по времени, тем более темнее будет ее синий цвет.

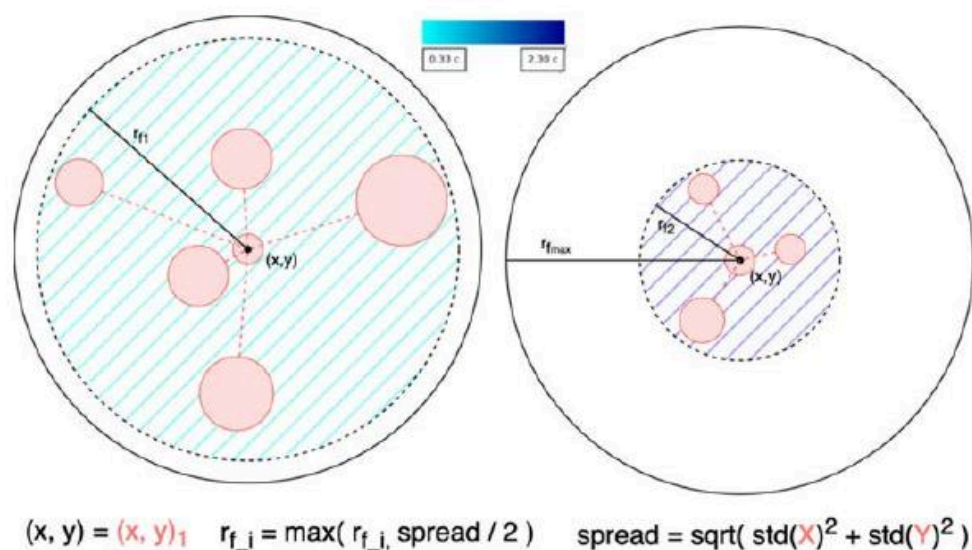


Рис. 4. Расчет фиксации взгляда

Результаты. Для проверки работоспособности трекера взгляда был реализован графический интерфейс пользователя, позволяющий настроить все параметры работы системы (рис. 5).

На рис. 6 представлены результаты испытания трекера взгляда на динамическом объекте управления (ОУ), на который предлагалось смотреть испытуемому.

При отслеживании взгляда, следящего за динамическим объектом, трекер показал достаточно высокую точность (~80 %) и небольшую среднеквадратичную ошибку (~10 %). Также системой были зарегистрированы небольшие фиксации взгляда во время слежения, вероятно потому, что цель выводится без заливки (только граница окружности), и пользователю не к чему было привязаться взглядом. Тем не менее, по тепловой карте видно, что пользователь перемещал взгляд равномерно.

На рис. 7 представлены результаты просмотра пользователем сгенерированных картинок: диаграммы разбросы и тепловые карты взгляда.

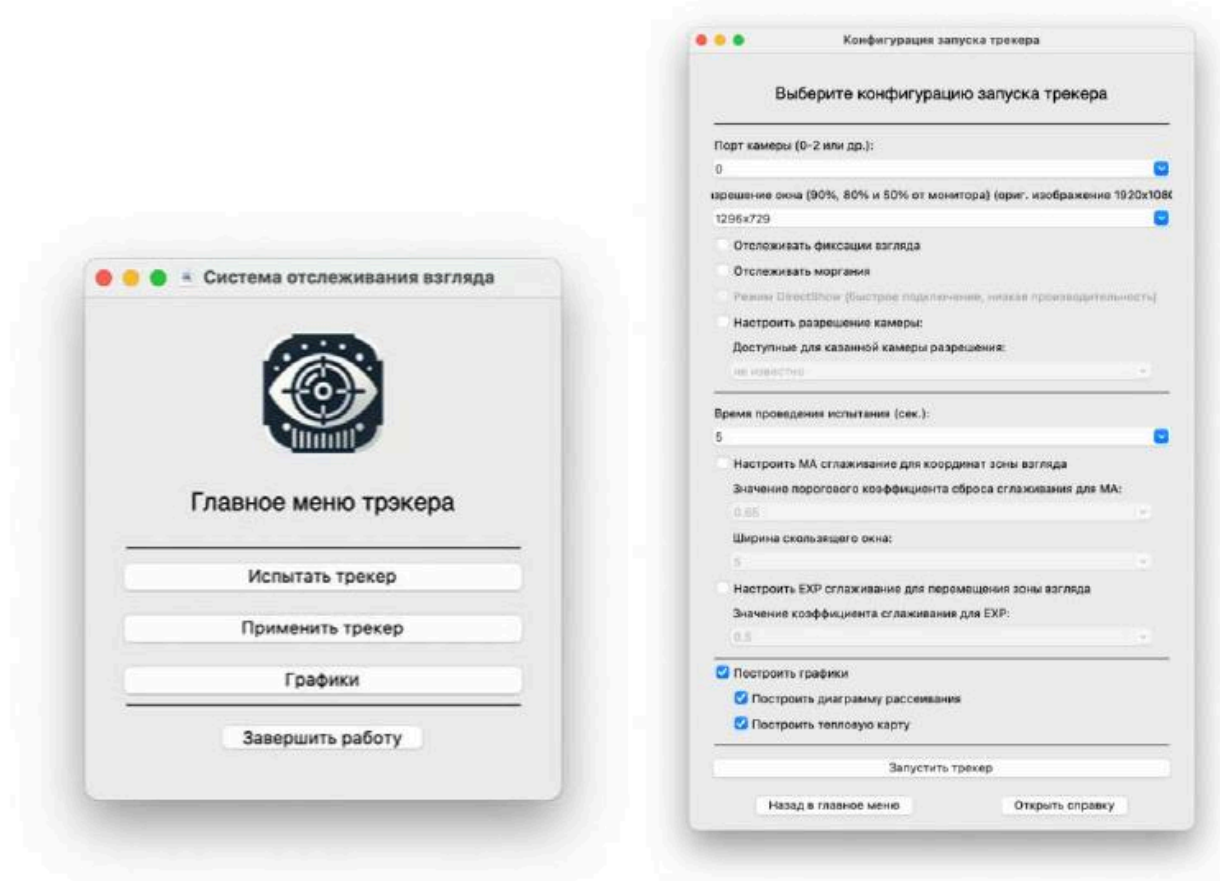


Рис. 5. Главное меню и окно конфигурации запуска

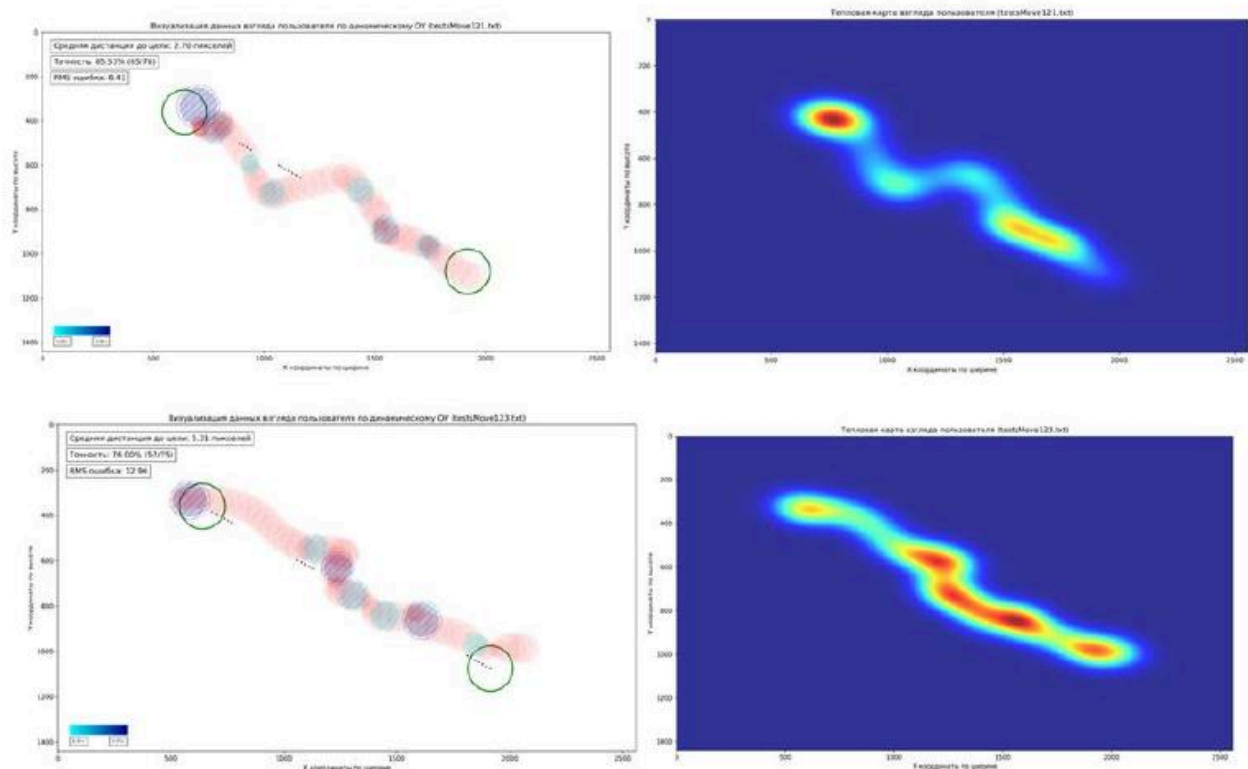


Рис. 6. Диаграммы рассеивания и тепловые карты для динамического объекта управления

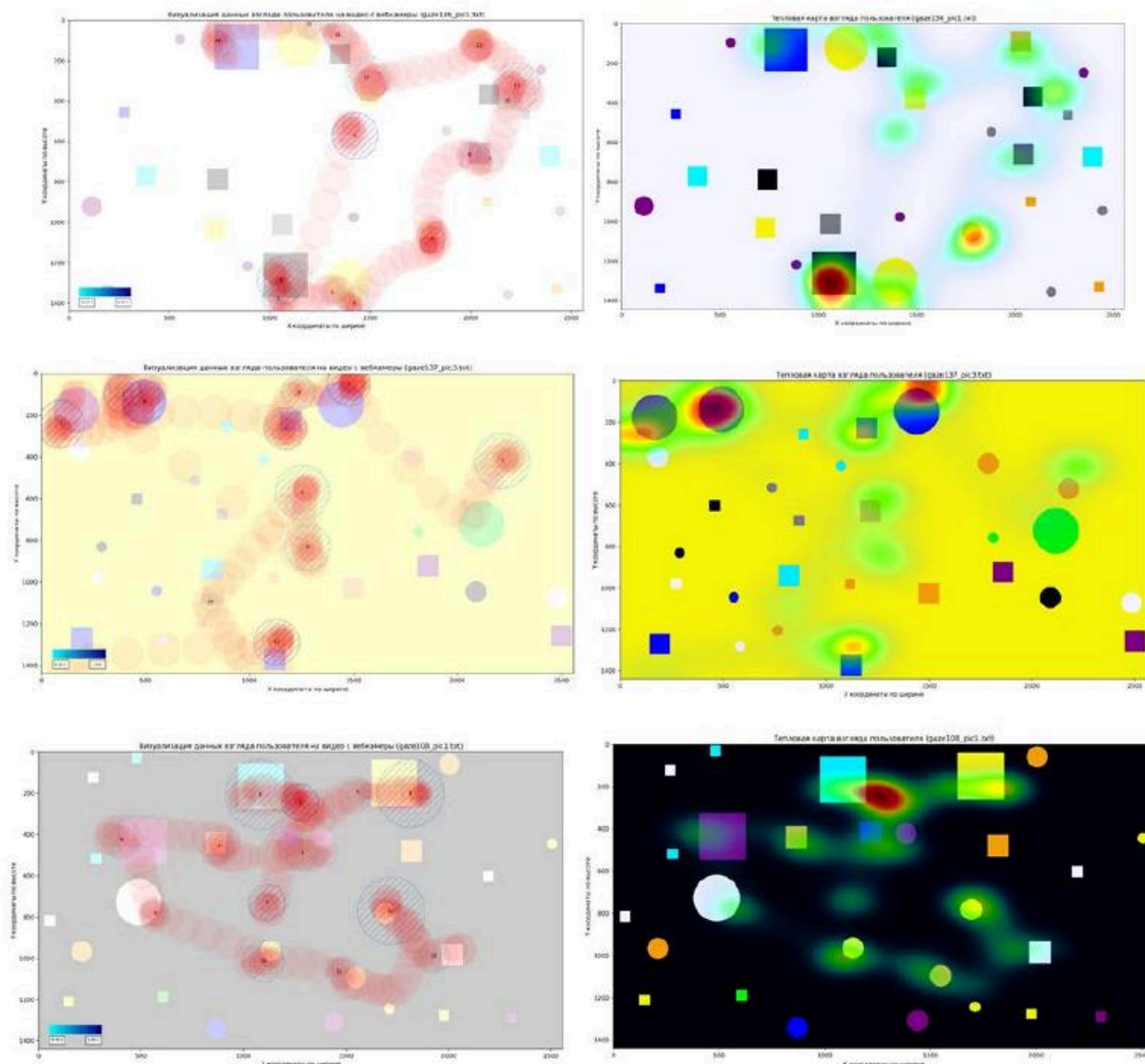


Рис. 7. Диаграммы разброса и тепловые карты для цветных контрастных ОУ

По полученным данным можно сказать следующее:

- при начале показа очередного изображения, реакция взгляда пользователя составляет ~1 секунду;
- первым объектом, на который пользователь обращает внимание, является более контрастный, и чем он больше при этом, тем более контрастным его видит пользователь периферическим зрением;
- дальнейшая последовательность переключения взгляда пользователя также зависит от контрастности очередного ОУ;
- после обхода всех больших и контрастных объектов, за оставшееся время осматриваются остальные, малоконтрастные объекты;
- по тепловым картам можно заключить, что на больших и высококонтрастных объектах пользователь задерживает взгляд дольше, чем на остальных.

Трекер не просто записал то, куда смотрел пользователь во время испытания, он записал как именно смотрел пользователь, уловил акценты внима-

ния [12] при взаимодействии пользователя с объектами различного размера и контрастности на экране. Также трекер был применен для изучения взаимодействия пользователя с интерфейсом. Так, на рис. 8 представлены результаты записи взгляда пользователя, при взаимодействии с интерфейсом OS Windows XP.

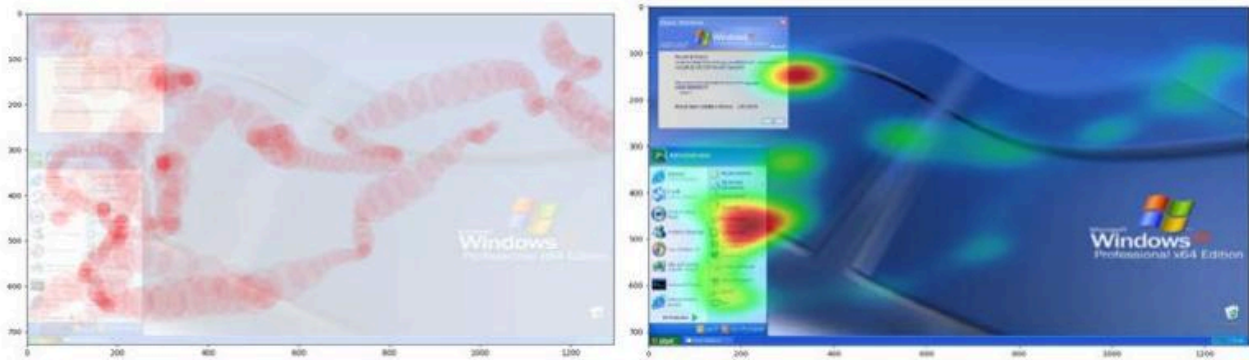


Рис. 8. Диаграммы разброса и тепловые карты WindowsXP

Обсуждение полученных результатов. Преимущества выбранной реализации визуализаций. Диаграмма рассеивания в модернизированном трекере значительно улучшает понимание того, как именно перемещается взгляд пользователя. В отличие от обычной тепловой карты. Она позволяет более точно проследить траекторию движения глаз. Каждая точка на диаграмме представляет конкретное положение взгляда, что делает возможным анализ даже мелких и быстрых движений, которые важно учитывать для точной оценки.

Кроме того, эта диаграмма помогает увидеть не только точку фиксации взгляда, но и радиус зоны внимания, что дает более полное представление о том, как пользователь взаимодействует с объектами на экране. Важно понимать не только, куда человек смотрит, но и насколько точно он удерживает внимание в нужной области (рис. 9).

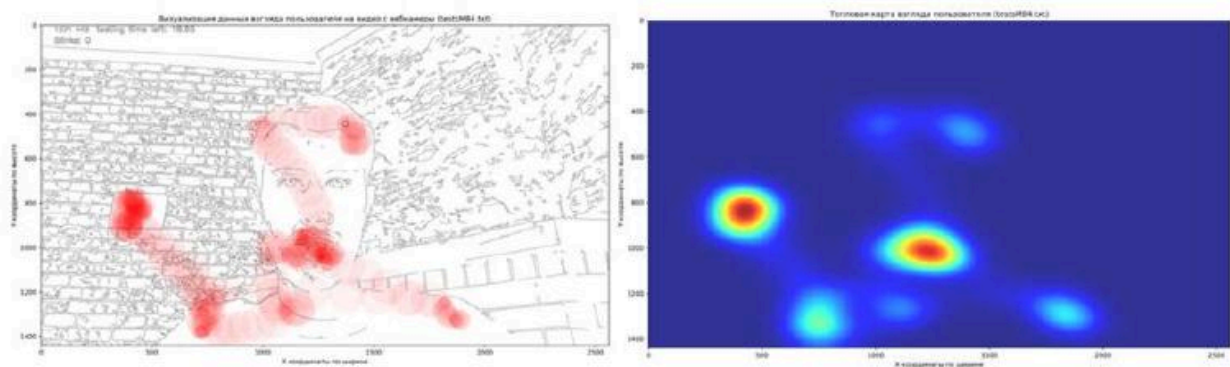


Рис. 9. Диаграмма разброса и тепловая карта модифицированного трекера

Наложение полупрозрачных кругов на диаграмму также играет важную роль: по мере увеличения числа точек взгляда в одной и той же зоне, интенсивность цвета усиливается. Это визуально аналогично тепловой карте [4, 5], но с дополнительной детализированной информацией. Чем ярче становится область, тем дольше пользователь фокусировался на ней, что делает диаграмму мощным инструментом для анализа точек максимального интереса и концентрации внимания. Визуализация данных взгляда совместно с зарегистрированными фиксациями (рис. 10) позволяет не только дополнительно подтвердить то, что в зоне с высокой интенсивностью цвета пользователь фиксировал свой взгляд, но и то, сколько именно продолжалась фиксация и какой разброс взгляда был у пользователя, что говорит нам о том, как именно он изучал объект на экране.

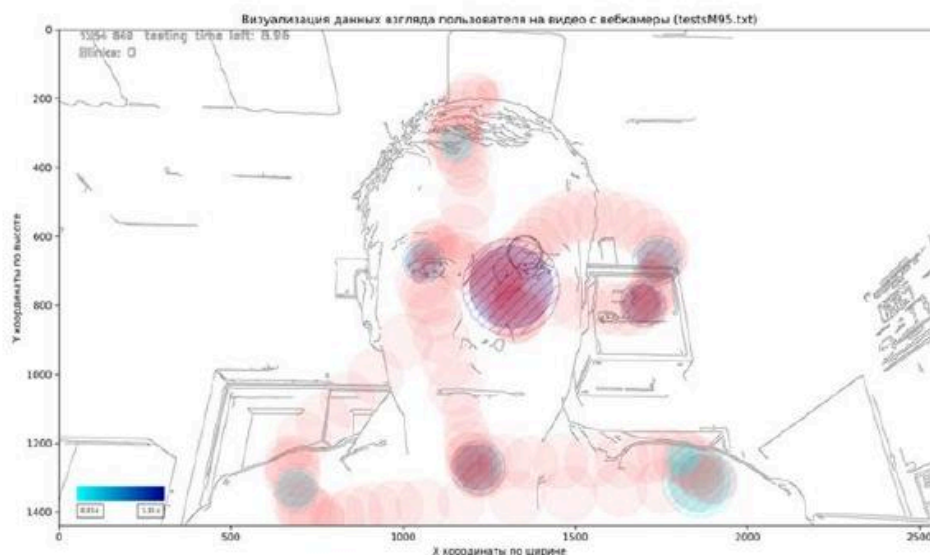


Рис. 10. Диаграмма разброса с фиксациями взгляда

Заключение. Настоящее исследование подчеркивает потенциал технологий трекинга глаз как основы для создания эффективных сложных информационных моделей, ориентированных на человеческое восприятие. Разработанный подход доказывает, что анализ поведения взгляда способен стать ключевым инструментом для построения систем, которые не только отражают данные, но и адаптируются к индивидуальным особенностям пользователей. Теоретически трекер может быть использован в самых разных областях, где важны точность интерпретации внимания и оптимизация взаимодействия с информацией. Прикладное значение технологии охватывает широкий спектр задач. В дизайне интерфейсов трекинг глаз позволяет создавать интуитивно понятные и эргономичные решения, размещая ключевые элементы там, где они естественным образом привлекают внимание. В маркетинге он открывает возможности для анализа восприятия рекламы, помогая выявлять, какие визуальные стимулы наиболее эффективны для целевой аудитории.

Потенциал трекера усиливается при интеграции с современными технологиями, такими как искусственный интеллект и дополненная реальность. Это открывает путь к созданию динамических, адаптивных [13], информационных моделей, способных в реальном времени реагировать на изменения внимания пользователя. Например, в виртуальных средах трекер может оптимизировать отображение данных, выделяя зоны интереса и минимизируя когнитивную нагрузку. Таким образом, трекинг глаз выходит за рамки простого инструмента анализа, становясь фундаментом для интеллектуальных систем, которые предугадывают потребности пользователя и повышают эффективность восприятия [14] пользователем информации в самых разных контекстах.

Список источников

- [1] Горячкин Б.С., Якубов А.Р., Аникин Ф.А. Разработка инструментария трекинга глаз. *Нейрокомпьютеры: разработка, применение*, 2024, т. 26, № 2, с. 5–12.
- [2] *Официальный сайт Python*. URL: <https://www.python.org> (дата обращения 08.05.2024).
- [3] *Документация по модулю "OpenCV"*. URL: <https://docs.opencv.org/4.x/index.html> (дата обращения 25.09.2024).
- [4] *Документация по модулю "matplotlib"*. URL: <https://matplotlib.org> (дата обращения 25.09.2024).
- [5] *Документация по модулю "pyplot.scatter"*. URL: https://matplotlib.org/stable/api/_as_gen/matplotlib.pyplot.scatter.html (дата обращения 28.09.2024).
- [6] Горячкин Б.С. Эргономический сертификат автоматизированной системы обработки и отображения информации и управления. *Международный научно-исследовательский журнал*, 2016, № 9–2 (51), с. 25–29.
- [7] *Понятие тепловой карты*. URL: <https://cloud.yandex.ru/ru/docs/datalens/visualization-ref/heat-map-chart> (дата обращения 25.09.2024).
- [8] Мунипов В.М., Зинченко В.П. *Эргономика: человекоориентированное проектирование техники, программных средств и среды*. Москва, Логос, 2001, 356 с.
- [9] *Документация по модулю «mediapipe»*. URL: <https://developers.google.com/mediapipe> (дата обращения 25.09.2024).
- [10] *Фильтрация шума сигнала*. URL: <https://habr.com/ru/articles/588270/> (дата обращения 01.10.2024).
- [11] *Основы оптимального стиля. Когнитивно-ориентированный анализ удобочитаемости текста программы*. URL: https://habr.com/ru/companies/epam_systems/articles/517398/ (дата обращения 12.10.2024).
- [12] Горячкин Б.С., Якубов А.Р., Аникин Ф.А. Исследование акцентов внимания для различных видов изображений с помощью самописного трекера глаз. *Наукосфера*, 2024, т. 5, № 2, с. 336–346. <https://doi.org/10.5281/zenodo.11487319>
- [13] Горячкин Б.С., Бобров Д.В. Эффективность принципов адаптивной верстки при разработке пользовательских интерфейсов. *Динамика сложных систем – XXI век*, 2023, т. 17, № 1, с. 55–62. <https://doi.org/10.18127/j19997493-202301-04>
- [14] Горячкин Б.С. Эргономический анализ систем обработки информации и управления. *Интернет-журнал Наукоедение*, 2017, т. 9, № 3. URL: <http://naukovedenie.ru/PDF/79TVN317.pdf> (дата обращения 01.04.2025).