

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	2
АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ .....	3
ЭТАПЫ СОЗДАНИЯ ПРОЕКТА .....	4
АЛГОРИТМ РАБОТЫ ПРОГРАММЫ.....	5
ЗАКЛЮЧЕНИЕ И ВЫВОДЫ.....	5
СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ.....	6
ПРИЛОЖЕНИЕ 1 .....	7
ПРИЛОЖЕНИЕ 2.....	7

## ВВЕДЕНИЕ

В 21 веке человек не может жить полноценно без обмена информацией с внешним миром. Наличие компьютеров и мобильных устройств позволяет в несколько раз упростить этот процесс, но для людей, которым трудно или невозможно управлять своими движениями, например, с детским церебральным параличом (ДЦП), необходим особый инструмент. Невозможность управления компьютерами или другими средствами коммуникации при помощи рук является для них **актуальной проблемой**.

**Целевая аудитория:** люди, лишённые возможности управлять своими руками.

**Цель:** Разработать систему управления компьютером при помощи взгляда.

**Задачи:**

- Поиск и анализ существующих решений;
- Разработка собственной системы управления взглядом;
- Написание нейронной сети на языке Python, согласование её работы со всей системой;
- Объединение всех наработок в готовый продукт.

## АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Существующие системы управления взглядом (Eye tracking) - комплекс аппаратно-программных решений для обеспечения возможности коммуникации и управления ПК для людей с ограниченной подвижностью.

Технология Eye tracking разработана для того, чтобы обездвиженные люди могли вести разносторонний образ жизни: общаться с родственниками или обучаться, управляя своим взглядом компьютером, планшетом, смартфоном, телевизором и так далее. Конечно, эта идея не нова. В мире уже существуют такие системы.

Наименование	Цена	Примечание
Приставка PCEye Mini	170 000 руб.	-
PCEye 5	190 000 руб.	Только английский интерфейс
Приставка Tobii eyetraking 4с	32 000 руб.	Погрешность $\pm$ градус, есть русский язык
Программа GazeRecorder	Бесплатно	Только английский интерфейс, погрешность 2-3 градуса (3-4 см)

Как видно из приведённой таблицы, цены продуктов для большинства российских потребителей слишком высоки. В некоторых случаях не предусмотрено послепродажное обслуживание. Бесплатно распространяемое программное обеспечение не предусматривает работу с русским языком: интерфейс программы только на английском языке и без клавиатуры с кириллицей.

Проведённые исследования показали, что есть программное обеспечение с открытым исходным кодом для создания самодельных систем. Эксперимен-

ты, проведённые с таким программным обеспечением (ПО) и со своей web-камерой показали, что они не дают приемлемого качества работы.

## ЭТАПЫ СОЗДАНИЯ ПРОЕКТА

Я решил попробовать свои силы в создании системы управления компьютером с помощью взгляда. Опыт разработки искусственного интеллекта у меня уже есть, в частности, нейронной сети для распознавания маски на лице, тоже на языке Python.

- 1 этап. На этом этапе был разработан класс *Image* (для обработки изображений). Метод *eye\_search* принимает на вход изображение с камеры, определяет лицо и глаза, обрезает и преобразовывает картинку. Используются библиотеки *numpy*, *pandas*, *opencv*.
- 2 этап. Разработка класса *Prediction* (для обучения сети и предсказаний). Методы этого класса готовят данные для обучения сети и предсказания точки, затем происходит обучение сети. Используются библиотеки *numpy*, *tensorflow*, *sklearn*, *pandas*, *opencv*.

Добавлю, что для обучения нейросети можно использовать интерактивные облачные среды Google Colab или Jupiter Notebook. Но я разработал и обучил её на своём ПК для того, чтобы удостовериться, что нейросеть может обучаться и работать на любом, даже маломощном устройстве.

- 3 этап. Разработка класса *MainWindow* (для управления окном). Методы этого класса настраивают окно, видеопоток, в общем, отвечают за программный интерфейс (GUI). Используются библиотеки *Opencv*, *PIL*, *Pyautogui*, *Win32api*, *Tkinter*.

## АЛГОРИТМ РАБОТЫ ПРОГРАММЫ

- Обнаружение лица и глаз.
- Стандартизация изображения.
- Запуск, и последующее обучение сети (задачи по типу: «посмотрите наверх», «влево», «вниз», и ещё 20–30 точек), которое производится при первом запуске программы.
- Обучение сети на точках: нужно смотреть на курсор и ставить им точки на экране. Тут можно запустить шарик или любую другую фигуру, за перемещениями которой нужно следить взглядом. Сеть соотносит изображение глаз с координатами точки на экране. (см. [Приложение 1](#))
- Перемещение указателя мыши в предсказанную точку.
- Если человек смотрит в одну точку более 2-3 секунд, происходит нажатие.

На изображении в [Приложении 2](#) программа в действии. Она перемещает курсор близко к точке взгляда. Зелёная точка - это то, куда я смотрю, а красная, там, где находится курсор.

## ЗАКЛЮЧЕНИЕ И ВЫВОДЫ

В результате работы над проектом мне удалось создать систему для управления компьютером при помощи взгляда - только глаза и компьютер с видеокамерой. Действие программы обеспечивает свёрточная нейронная сеть, разработанная на языке Python.

На данный момент программа работает с погрешностью 3-4 см. Но она имеет большой потенциал для доработки. Можно адаптировать её к наклону и повороту головы, усовершенствовать алгоритм распознавания лица и глаз (использовать библиотеку face\_recognition), или применить другую архитектуру сети.

СПОСИБО ЗА ВНИМАНИЕ!

#### СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. <https://habr.com/ru/post/208108/>
2. <https://www.eyetracking.care>
3. <https://invamagazin.ru/g25663339-ustrojstva-dlya-upravleniya>
4. <http://www.cameramouse.org/>
5. <https://eviacam.crea-si.com/>
6. <https://www.eyetracking.care/>
7. [https://www.igromania.ru/article/29445/Kak\\_upravlyat\\_igrami\\_vzglyadom\\_Eye\\_Tracking\\_istoriya\\_i\\_Tobii\\_Eye\\_Tracker\\_4C.html](https://www.igromania.ru/article/29445/Kak_upravlyat_igrami_vzglyadom_Eye_Tracking_istoriya_i_Tobii_Eye_Tracker_4C.html)
8. <https://blogs.windows.com/russia/2017/08/02/eye-control/>
9. <https://invamagazin.ru/g25663339-ustrojstva-dlya-upravleniya>
10. <https://www.smartaids.ru/catalog/eyetracking/ustroystva-ay-treking/>
11. <https://www.eyetracking.care/informatsiya/stati/upravlenie-vzglyadom-i-tserebralnyy-paralich/>

## ПРИЛОЖЕНИЕ 1



## ПРИЛОЖЕНИЕ 2

