Автономная некоммерческая организация дополнительного профессионального образования

«Центр обучения «Махаон»

«Умная камера на основе ESP32-cam»

Выполнил:

ученик 9В класса Полосухин Богдан

Научный руководитель:

педагог д.о.

Морозов Антон Дмитриевич

Омск - 2023

ПЛАН

[ВВЕДЕНИЕ 3](#_Toc1)

[ОПИСАНИЕ ПРОЕКТА 4](#_Toc2)

[ТЕХНИЧЕСКОЕ ОПИСАНИЕ 5](#_Toc3)

[ВЫВОД 6](#_Toc4)

[ИСТОЧНИКИ 7](#_Toc5)

# ВВЕДЕНИЕ

В наше время бума нейронных сетей (далее ИИ) мы можем наблюдать множество возможностей использовать их для работы, учебы и развлечений, нейронные сети могут позволить автоматизировать буквально всё, что до сих пор не автоматизировано. В своё время я занимался спортом и видел, как тренер должен был уследить буквально за каждым спортсменом, стараться дать знания и умения каждому, при этом понятно, что уследить за всеми будет сложно или даже невозможно, поэтому я решил разработать свой проект, умную камеру на основе микроконтроллера ESP-32.

Цель:

Разработать камеру с использованием ИИ и позволить следить за спортсменами, зачастую расположенными в разных местах

Задачи:

1.Выбрать микроконтроллер и плату

2.Изучить язык программирования C++

3.Разработать корпус смой камеры и продумать удобство работы

4.Разработать или найти ИИ для поиска людей на картинке

5.Разработать Web-интерфейс для работы с камерой

# ОПИСАНИЕ ПРОЕКТА

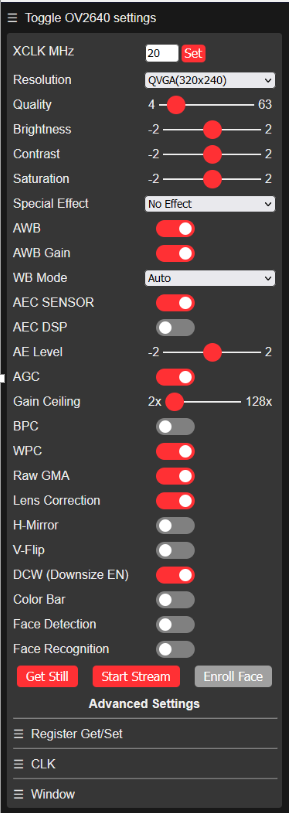
Камера помещается на подвижном штативе, которая поворачивается вслед за пользователем

При первом подключении камеры по USB вам следует скачать ARDUINO IDE и открыть код по ссылке, там вы вводите название своей Wi-FI сети и пароль, ваше устройство должно находиться в той же сети, чтобы получать изображение, после этого в буфер вам выведется IP, который вы должны ввести в браузер. Проверьте вывод изображения. После этого Нажмите «Start Stream», установите XCLK Mhz = 10, включите V-Flip, установите разрешение = QVGA (320x240), установите режим WB = «Sunny», а затем включите распознавание лиц. В перспективе я планирую написать небольшое клиентское приложение, которое будет выполнять все эти настройки автоматически. После этого вы устанавливаете камеру в нужное место цепляя её за специальное крепление внизу конструкции. После этого вы можете просматривать изображение с камеры прямо в браузере. После окончания работы с камерой вы можете отключиться от Wi-Fi соединения камеры.

На камере вы можете увидеть кнопку “Reset” На плате для срочной перезагрузки платы, например, при зависании или неправильной работе проекта

В низу корпуса вы можете увидеть разъем USB Type-c он нужен для подзарядки аккумулятора устройства, а так же для настройки камеры, при подключении к ПК.

В случае, если больше одного человека в кадре, камера направляется на того, который находится ближе к центру кадра



# ТЕХНИЧЕСКОЕ ОПИСАНИЕ

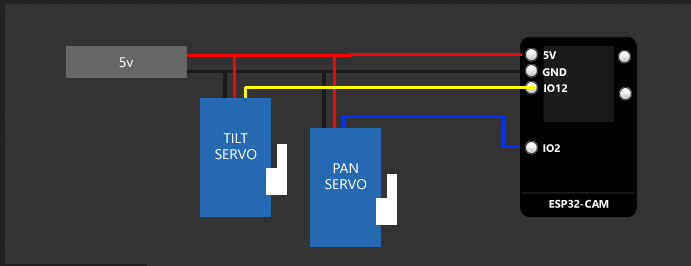
В разработке использовалась плата ESP32-cam, встроенная камера и программатор Wroom, все программирование проводилось в среде разработки Arduino IDE на языке C++ с фреймворком Wiring, были использованы библиотеки ArduinoWebSockets, Servo, WiFi.

Изображение берется с камеры и при помощи ArduinoWebSocket Отправляется в Web интерфейс по IP

Для использование машинного зрения использовалась смесь двух обученнаых нейросетей с сайта Pastebin (https://pastebin.com/cLY0MAYx) и проект на площадке GitHub пользователя AlexGyver (https://github.com/AlexGyver/FanCV).

Внутренняя конструкция состоит из следующих элементов: микроконтроллер ESP32-cam, два сервопривода (360o и 90o), аккумулятор, корпус напечатан на 3D-принтере.

Исходный код доступен по ссылке на : <https://github.com/Feodalniy-lift/Smart_Cam_ESP.git>



# ВЫВОД

Научился писать на языке программирования C++ и разработал прототип Умной камеры на основе ESP32-cam.

В ближайшее время планируется возможность использовать сразу несколько камер и переключаться между ними, а также написание программы под ОС Windows и Linux для автонастройки камеры

Исходный код проекта доступен по ссылке : <https://github.com/Feodalniy-lift/Smart_Cam_ESP.git>

# ИСТОЧНИКИ

* <https://www.learnrobotics.org/blog/face-tracking-opencv/?utm_source=youtube&utm_medium=description&utm_campaign=face_tracking_video>
* <https://robotzero.one/face-tracking-esp32-cam/>
* <https://pastebinp.com/ECQPxuec>
* <https://randomnerdtutorials.com/esp32-cam-video-streaming-face-recognition-arduino-ide/>
* <https://how2electronics.com/esp32-cam-based-face-eyes-recognition-system/>
* <https://www.instructables.com/ESP32-CAM-Video-Streaming-Face-Recognition-Using-A/>
* <https://www.hackster.io/abdularbi17/how-to-install-esp32-board-in-arduino-ide-1cd571>
* <https://dronebotworkshop.com/esp32-servo/#:~:text=The%20ESP32%20has%20analog%20outputs,can%20drive%20several%20servo%20motors>