# ***Занятие 25.*** *Как обрабатывается видео в OpenCV–Python*

## **О занятии**

**Краткое содержание:**

На этом занятии мы создадим медитацию с использованием функции Гигачата. Изучим детекцию в компьютерном зрении, познакомимся с каскадами Хаара и принципом их работы. Повторим шаги по настройке рабочей среды, напишем код для анализа изображения и подсчета количества лиц на фото, а также код для обнаружения лиц с использованием веб-камеры. В заключение мы научимся анализировать различные видео с помощью детекции лиц.

**Ключевые навыки:**

* Создание медитации с помощью Гигачата.
* Знание детекции, каскады Хаара в компьютерном зрении.
* Составление кода для подсчета количества лиц на фото.
* Создание кода для обнаружения лиц с использованием веб-камеры.

**Необходимые инструменты:**

Гигачат, Visual Studio Code.

**Термины:**

***Детекция*** — обнаружение объектов определенного типа на изображении или видео, например, лиц, жестов, движений.

***Каскады Хаара*** — это метод, который используется для обнаружения объектов на изображениях.

***Bounding box*** — это ограничивающий прямоугольник, который обводит обнаруженный объект на изображении.

## 

## **Практическая работа**

### Задание 1.

В главном меню Гигачата выберите раздел «Полезные AI-функции» и нажмите «Устройте себе сеанс медитации». Опишите, что сейчас чувствуете: например, усталость, упадок сил. Опишите желаемый результат медитации: вернуть энергию и снова стать продуктивным. Опишите куда вы отправитесь медитировать: например, в горы. Затем выберите «Сгенерировать текст» и нажмите «Продолжить». Выберите звуковое сопровождение и немного подождите – медитация будет готова.

#### Задание для самостоятельной работы:

При помощи Гигачата создайте медитацию, которая учтет ваши пожелания.

### Задание 2.

Создайте новую папку «Детекция» и файл с расширением .py под названием «Count\_detection» внутри этой папки. Выполняйте те же шаги по настройке рабочей среды, что делали в предыдущем занятии.

### Задание 3.

Откройте Visual Studio Code, импортируйте модуль OpenCV. Затем в переменной img загрузите изображение и преобразуйте его в оттенки серого. Далее в переменной face\_cascade запишите каскад Хаара, необходимый для детекции лиц. Создайте переменную “faces”, в которой будут храниться координаты обнаруженных лиц на изображении. Затем программа подсчитывает количество этих найденных лиц и выводит результат.

#### Задание для самостоятельной работы:

Выберите фото с семьей или с друзьями и используйте предложенный код для подсчета количества лиц. Код находится в материалах к занятию в файле “Count\_detection.py”

### Задание 4.

Создайте новый файл с расширением .py «Detection» и начните с похожих строк кода, как в предыдущем примере. Выберите источник видеоданных - веб-камеру, удалив строчку для чтения видео с веб-камеры.

### Задание 5.

**Промпт:** Почему в функции cv2.VideoCapture() используется аргумент 0 для захвата видео с веб-камеры?

**Возможный результат:**

Аргумент 0 в функции cv2.VideoCapture() используется для указания источника видео. В данном случае, когда мы хотим получить видео с веб-камеры, аргумент 0 указывает на то, что источником является первая доступная веб-камера. Если у вас есть несколько веб-камер, вы можете использовать аргументы 1, 2 и т.д. для выбора конкретной камеры.

#### Задание для самостоятельной работы:

При помощи Гигачата узнайте, какие форматы видео поддерживает OpenCV при использовании функции cv2.VideoCapture().

### Задание 6.

Продолжайте работать в файле «Detection.py». Переносится в код цикл, который будет считывать по очереди каждый кадр из видеопотока и анализировать его. Затем следующая часть кода рисует bounding box для лица. Далее показывается изображение, то есть видео, в отдельном окне. Это окно будет открыто, пока вы не нажмете клавишу ESC на клавиатуре. Каждые 30 миллисекунд проверяется, была ли нажата клавиша с кодом 27, то есть ESC и если да, то цикл прерывается.И наконец, после цикла дописывается в код команда cap.release(), останавливающая прием видеопотока. Код находится в материалах к занятию в файле “Detection.py”.

#### 

#### Задание для самостоятельной работы:

Выполните детекцию лица в видеопотоке, используя любое видео.

## 

## **Итоги занятия**

Сегодня на занятии вы:

* Познакомились с медитацией от Гигачата.
* Узнали, что такое детекция компьютерного зрения.
* Познакомились с каскадом Хаара для обнаружения лиц.
* Научились подсчитывать количество лиц на фотографии с помощью каскада Хаара.
* Узнали, как обрабатывать видеопотоки с веб-камеры и видеофайла при помощи OpenCV.

Увидимся на следующем занятии! До скорой встречи!