Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ЛАБОРАТОРНАЯ РАБОТА №2**

**Дисциплина: Алгоритмы цифровой обработки мультимедиа**

Работу выполнил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Пинский Д.А.

Направление подготовки: 02.03.02 Фундаментальная информатика и информационные технологии

Преподаватель: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_А.А. Крамаренко

Лабораторная работа №2

Реализация трекинга

**Задание 1.** Прочитать изображение с камеры и перевести его в формат HSV.

Переведём изображение в формат HSV с помощью ранее используемой команды cv2.cvtColor(), указав в аргументах флаг HSV cv2.COLOR\_BGR2HSV.

RGB и HSV — это два различных цветовых пространства, которые используются для представления цветов. RGB — это аддитивное цветовое пространство, которое используется для представления цветов на экране. Оно состоит из красного, зеленого и синего цветов. HSV — это цветовое пространство, которое используется для представления цветов в виде оттенка, насыщенности и значения. Он также называется HSB (оттенок, насыщенность, яркость). Он позволяет легче управлять цветом, чем RGB.

**Задание 2.** Применить фильтрацию изображения с помощью команды inRange и оставить только красную часть, вывести получившееся изображение на экран(treshold), выбрать красный объект и протестировать параметры фильтрации, подобрав их нужного уровня.

Преобразуем кадр в цветовое пространство HSV при помощи cv2.cvtColor(frame, cv2.COLOR\_BGR2HSV), что облегчает фильтрацию по цвету.

Определим цветовую маску при помощи метода cv2.inRange(), который создает маску, которая выделяет области, соответствующие заданному диапазону красного цвета. Далее при помощи cv2.bitwise\_or() применим маску к кадру, оставляя только пиксели, соответствующие красному цвету.

**Задание 3.** Провести морфологические преобразования (открытие и закрытие) фильтрованного изображения, вывести результаты на экран, посмотреть смысл подобного применения операций erode и dilate.

Морфологические преобразования - это операции, применяемые к изображениям с целью изменения их формы и структуры на основе формы и структуры объектов на изображении. Два из наиболее распространенного морфологического преобразования - это "открытие" и "закрытие". Эти операции часто используются в обработке изображений для удаления шума, заполнения дыр в объектах и изменения размера объектов. Давайте рассмотрим каждую из них более подробно:

Операция открытия состоит из двух шагов: сначала применяется эрозия (erode), а затем – дилатация (dilate). Сначала эрозия удаляет маленькие объекты и "шум" на изображении, уменьшая объекты и заполняя небольшие прорехи. Затем дилатация восстанавливает объекты близкой к их исходному размеру.

Открытие полезно для удаления шума, разделения объектов, связанных друг с другом, и выделения объектов, близких к заданной форме.

Операция закрытия также состоит из двух шагов: сначала применяется дилатация (dilate), а затем – эрозия (erode). Сначала дилатация увеличивает объекты и заполняет небольшие отверстия в объектах. Затем эрозия уменьшает объекты обратно к их исходному размеру. Закрытие полезно для закрытия небольших отверстий в объектах и объединения близко расположенных объектов.

Применение открытия и закрытия может быть полезным при обработке изображений в различных задачах компьютерного зрения. Эти операции особенно эффективны в тех случаях, когда объекты на изображении имеют различные размеры или, когда на изображении присутствует некоторый уровень шума. Использование морфологических преобразований требует выбора правильных структурирующих элементов (ядро) и настройки их размера в зависимости от конкретной задачи.

**Задание 5.** На основе анализа площади объекта найти его центр и построить черный прямоугольник вокруг объекта. Сделать так, чтобы на видео выводился полученный черный прямоугольник, причем на новом кадре.

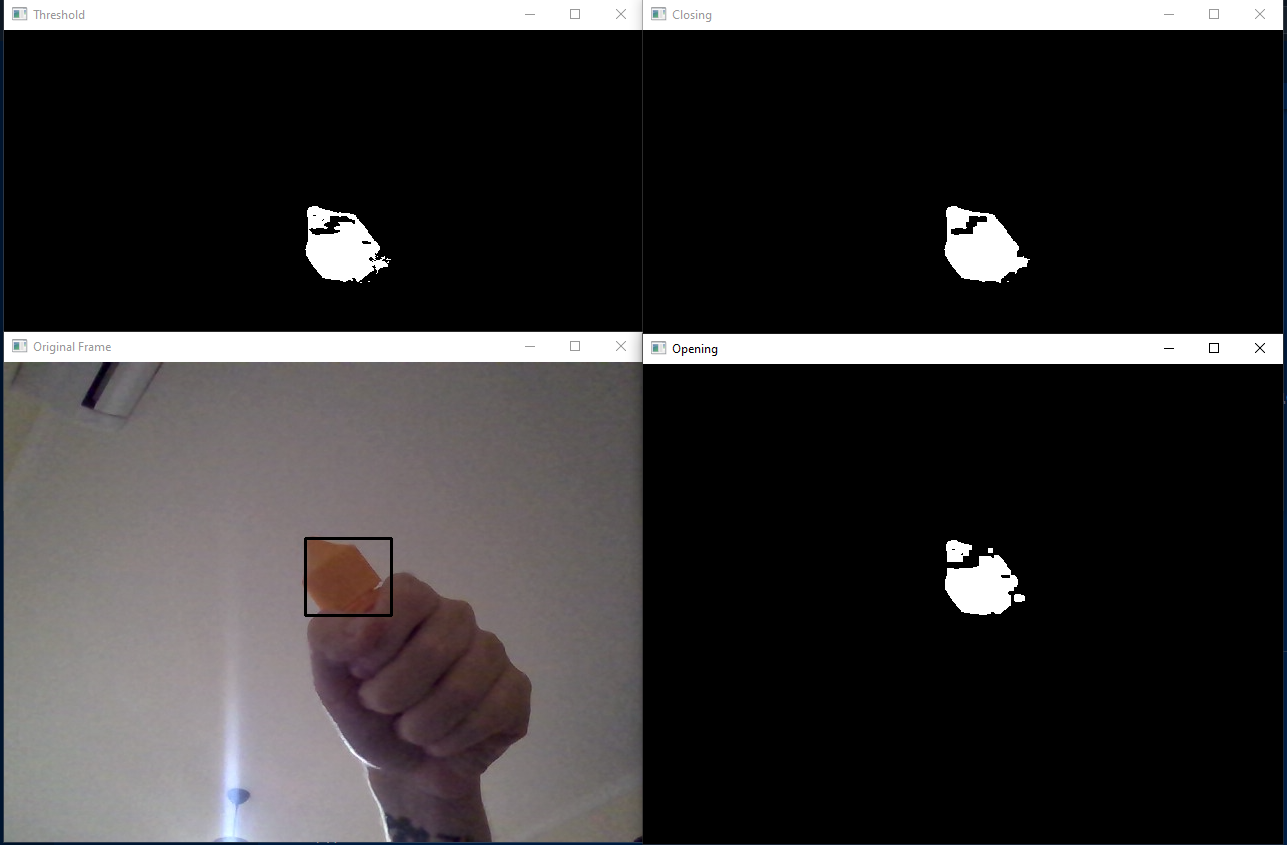
Моменты изображения - это статистические показатели, которые описывают форму, структуру и распределение пикселей в изображении.

В нашем случае мы используем метод cv2.moments().

m00 - это один из моментов изображения, который называется нулевым моментом. Нулевой момент представляет собой интегральную яркость (или массу) всего изображения и используется для вычисления центра масс объекта.

По координатам m10 и m01 можно вычислить координаты центра масс объекта.

Далее рисуется черный прямоугольник вокруг центра масс объекта.

Рисунок 1 – пример выполнения программы

**Ответы на вопросы:**

1. Функция inRange используется для фильтрации пикселей в пределах заданного диапазона значений в изображениях. Функция принимает три параметра: входное изображение, нижнюю границу диапазона и верхнюю границу диапазона. Пикселям, попадающим в указанный диапазон, присваивается значение белого, в то время как пикселям, выходящим за пределы диапазона, присваивается значение черного
2. Методы размывания и расширения изменяют форму и структуру объектов на изображении с использованием математических преобразований.

erode: метод стирания уменьшает размер объекта переднего плана и стирает границы пикселей объекта. Он удаляет пиксели на границах объекта и уменьшает изображение. Эрозия полезна для таких задач, как подавление шума и удаление мелких объектов или деталей с изображения.

dilate: метод расширения увеличивает размер объекта переднего плана и расширяет границы пикселей объекта. Это добавляет пиксели к границам объекта и увеличивает его размер. Расширение полезно для таких задач, как заполнение отверстий, соединение разорванных частей объекта или утолщение линий на изображении.

1. Морфологическое открытие и закрытие - это сочетание эрозии и расширения, которые используются для удаления шума и мелких объектов, а также для заделки небольших отверстий или зазоров.

Opening: Морфологическое вскрытие - это сочетание эрозии с последующей дилатацией. Открытие помогает сгладить границы объектов и разделить связанные объекты.

Closing: Морфологическое закрытие - это сочетание дилатации с последующей эрозией. Закрытие помогает завершить форму объекта и устранить небольшие пробелы.

4. Моменты изображения - числовые характеристики, описывающие распределение интенсивности пикселей в изображении. Различают несколько типов моментов:

* Геометрические моменты - описывают форму и размер объекта. Нулевой момент равен площади объекта, первые моменты - координатам центра масс (центроида), центральные моменты инвариантны к сдвигу.
* Физические моменты - связаны с физическими характеристиками объекта, например, массой.
* Алгебраические моменты - используются для построения инвариантных признаков объекта, например, для распознавания.

Моменты широко применяются в задачах компьютерного зрения для анализа формы объектов, вычисления их характеристик, распознавания и классификации.

m00 - это один из моментов изображения, который называется нулевым моментом. Нулевой момент представляет собой интегральную яркость (или массу) всего изображения и используется для вычисления центра масс объекта.

area представляет собой значение нулевого момента, которое соответствует площади объекта на изображении. Это значение используется для определения площади объекта.

5. Центроид (центр масс) объекта - точка, определяющая геометрический центр объекта. Она вычисляется как среднее арифметическое координат всех пикселей объекта. Центроид используется для определения местоположения объекта на изображении. Он может служить опорной точкой для дальнейшего анализа, например, для отслеживания движения объекта, вычисления расстояния до него и т.д.

**Листинг программы**

import cv2

import numpy as np

cap = cv2.VideoCapture(0)

while True:

    ret, frame = cap.read()

    if not ret:

        break

    # Переводим кадр в формат HSV

    hsv\_frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2HSV)

    # Определяем диапазоны красного цвета в HSV

    lower\_red1 = np.array([0, 100, 100])

    upper\_red1 = np.array([10, 255, 255])

    lower\_red2 = np.array([160, 100, 100])

    upper\_red2 = np.array([180, 255, 255])

    # Применяем фильтрацию

    mask1 = cv2.inRange(hsv\_frame, lower\_red1, upper\_red1)

    mask2 = cv2.inRange(hsv\_frame, lower\_red2, upper\_red2)

    mask = cv2.bitwise\_or(mask1, mask2)

    # Морфологические преобразования

    kernel = np.ones((5, 5), np.uint8)

    # Применяем операцию открытия (эрозия + дилатация)

    opened\_image = cv2.erode(mask, kernel, iterations=1)

    opened\_image = cv2.dilate(opened\_image, kernel, iterations=1)

    # Применяем операцию закрытия (дилатация + эрозия)

    closed\_image = cv2.dilate(mask, kernel, iterations=1)

    closed\_image = cv2.erode(closed\_image, kernel, iterations=1)

    # Применяем операции открытия и зарытия cv2

    # opening = cv2.morphologyEx(mask, cv2.MORPH\_OPEN, kernel)

    # closing = cv2.morphologyEx(mask, cv2.MORPH\_CLOSE, kernel)

    contours, \_ = cv2.findContours(opened\_image, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)

    for cnt in contours:

        M = cv2.moments(cnt)

        if M['m00'] > 0:

            area = M['m00']

            X = int(M['m10'] / M['m00'])

            Y = int(M['m01'] / M['m00'])

            print(f'Площадь объекта: {area}')

            x, y, w, h = cv2.boundingRect(cnt)

            cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 0, 0), 2)

    cv2.imshow('Original Frame', frame) # Выводим оригинальное изображение и площадь найденных обьектов

    cv2.imshow('Threshold', mask) # Фильтрация

    cv2.imshow('Opening', opened\_image) # Открытие (эрозия + дилатация)

    cv2.imshow('Closing', closed\_image) # Закрытие (дилатация + эрозия)

    # cv2.imshow('Opening', opening) # Открытие

    # cv2.imshow('Closing', closing) # Закрытие

    if cv2.waitKey(1) & 0xFF == 27:

        break

cap.release()

cv2.destroyAllWindows()