

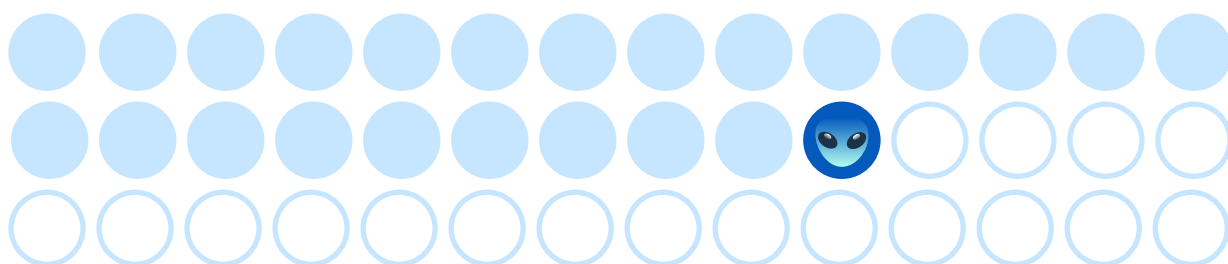
# Инструменты ИИ: применяем и программируем

2 модуль

## Занятие 2.3.2

### Выделение объектов по цвету на изображениях

О занятии .....	1
Практическая работа .....	3
Итоги занятия .....	6



# О занятии

Рабочая тетрадь предназначена для слушателей курса «Инструменты ИИ: применяем и программируем».

Данное пособие поможет слушателям лучше усвоить материал, представленный в видеоролике. Содержит практические задания для закрепления навыка работы с нейросетями, а также инструкции по регистрации и авторизации на сайтах нейросетей.

В рабочей тетради предложены термины, которые помогут лучше понять темы и закрепить знания об искусственном интеллекте.

## Цель практического занятия:

На этом занятии мы узнаем, как OpenCV используется для поиска объектов по цвету с помощью метода пороговой сегментации. Настроим рабочую среду для написания кода. Познакомимся с цветовыми пространствами HSV и BGR, а также преобразуем изображение из одного пространства в другое. Научимся строить и вращать трёхмерный график цветового рассеивания, найдём пороговые значения для определения нужных цветов. В заключение займёмся сегментацией изображения по выбранным цветам, накладывая маски на изображение.

## Задачи:

- Узнать метод пороговой сегментации.
- Преобразовать из одного цветового пространства в другое.
- Построить трёхмерную графику цветового рассеивания.
- Установить пороги для определения нужного диапазона цветов.
- Сегментировать изображения по выбранным цветам.

## Необходимые инструменты:

Gigachat, Visual Studio Code.

# О занятии

## Теория и термины

**Пороговая сегментация** — метод, в котором устанавливается пороговое значение для разделения изображения на две части.

**Цветовое пространство HSV** — модель представления цвета, включающая три компонента: Hue (Оттенок), измеряемый в градусах от 0 до 360, Saturation (Насыщенность), измеряемую в процентах, и Value (Яркость), также измеряемую в процентах.

**Трёхмерный график** — графическое представление данных в трёх измерениях, где каждая точка отображает значение по трём осям.

# Практическая работа

## Задание 1

Прежде чем приступить к преобразованию изображения в нужное цветовое пространство, необходимо настроить рабочую среду:

1. Создать новую папку «Сегментация».
2. В Visual Studio Code в левом верхнем углу нажать кнопку «Файл», выбрать «Открыть папку», затем выбрать на компьютере созданную папку «Сегментация» и нажать «Выбор папки».
3. Создать в этой же папке новый файл с расширением .py, а также добавить в эту папку все необходимые для урока изображения.
4. Убедиться, что установлены библиотеки OpenCV, Matplotlib и NumPy.

## Задание для самостоятельной работы:

Настройте рабочую среду, следуя рекомендациям в материалах к занятию.

## Задание 2

Возьмите одну из подготовленных картинок под названием **tiger1.png** и преобразуйте её в цветовое пространство HSV.

Картинка находится в материалах к занятию в первом пункте: «Фотография в формате PNG».

Для начала подключите библиотеки:

- OpenCV – для обработки изображения.
- Matplotlib – для построения графиков.
- NumPy – для вычислений.

OpenCV по умолчанию использует цветовое пространство BGR, которое аналогично RGB, но с другим порядком цветов.

Поэтому вам нужно сначала преобразовать изображение из BGR в RGB, а затем в HSV, чтобы цвета отображались правильно.

Для этого используйте функцию `imread`, чтобы загрузить изображение в формате BGR: `tiger = cv2.imread('tiger1.png')`.

**Задание для самостоятельной работы:**

Выполните преобразование изображения в HSV, применив данный код:

```
import cv2
import matplotlib.pyplot as plt
import numpy as np
from matplotlib import colors
tiger = cv2.imread('tiger1.png')
tiger = cv2.cvtColor(tiger, cv2.COLOR_BGR2RGB)
tiger = cv2.cvtColor(tiger, cv2.COLOR_BGR2RGB)
```

**Задание 3**

Самостоятельно перенесите код по построению трёхмерного графика к себе в файл. Он находится в материалах к занятию во втором пункте: «Код для построения трёхмерного графика цветового рассеивания».

**Задание 4****Промпт:**

Объясни этот код кратко:

```
pixel_colors = tiger.reshape((np.shape(tiger)[0]*np.shape(tiger)[1], 3))
norm = colors.Normalize(vmin=-1.,vmax=1.)
norm.autoscale(pixel_colors)
pixel_colors = norm(pixel_colors).tolist()
axis.scatter(h.flatten(), s.flatten(), v.flatten(), \
facecolors=pixel_colors, marker=".")

axis.set_xlabel("Hue")
axis.set_ylabel("Saturation")
axis.set_zlabel("Value")
plt.show()
```

**Возможный результат (Критерии оценивания и формат оформления):**

Этот код создает цветовой спектр изображения тигра:

1. Создаётся переменная `pixel_colors` с цветами каждого пикселя изображения.
2. Функция `reshape` преобразует переменную в нужный формат.
3. Применяется инструмент `Normalize` из `Matplotlib` для нормализации цветов.
4. Цвета преобразуются в список.
5. Функция `scatter` визуализирует цветовые спектры пикселей.
6. Устанавливаются названия осей графика и отображается результат.

**Задание 5****Промпт:**

Для определения нужного диапазона оранжевого цвета установите пороговые значения.

Оцените значения по осям на графике. Для этого поворачивайте график, двигая мышь с зажатой левой кнопкой.

Для более точных значений смотрите в правый нижний угол графика, наведя мышку на нужную область. Обращайте внимание только на целые числа.

Затем составьте светлую и тёмную версию, прописав в каждом три целых числа. Получите значения нижнего и верхнего порога оранжевого цвета.

```
light_orange = (11, 80, 0)
dark_orange = (19, 255, 255)
```

**Задание для самостоятельной работы:**

Попробуйте таким же образом определить пороговые значения белого цвета, используя тот же график цветового рассеивания.

**Задание 6**

Самостоятельно перенесите код для сегментации изображения по выбранным цветам.

Он находится в материалах к занятию в третьем пункте: «Код для установления порогов и маскирование изображения».

# Итоги занятия

На этом занятии вы:

- Познакомились с пороговой сегментацией.
- Узнали новые модели цветового пространства.
- Научились строить трёхмерный график цветового рассеивания.
- Приобрели навык установки порогов для поиска.

Дальше интереснее. До встречи на следующих занятиях!

Молодцы!

Переходите на платформу для решения задач.

Увидимся на следующем занятии!