

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра математического обеспечения и применения ЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Цифровая обработка изображений»
Тема: Сегментация изображений

Студент гр. 7381

Минуллин М.А.

Преподаватель

Черниченко Д.А.

Санкт-Петербург

2021

Постановка задачи.

Реализовать на языке Python с использованием библиотеки OpenCV программу, выполняющую сегментацию/кластеризацию изображения по цвету с использованием метода K-Means.

Входные данные.

1. Цветное изображение в формате bmp, jpg
2. Количество кластеров

Выходные данные.

Цветное изображение в формате bmp (result.bmp), являющиеся результатом сегментации (сегмент окрашивается средним цветом в кластере).

Выполнение работы.

Исходные данные будем получать из командной строки:

```
filename = sys.argv[1]  
k = int(sys.argv[2])
```

Откроем исходное изображение и отобразим его на экране (результат представлен на рис. 1):

```
image = cv2.imread(filename)  
cv2.imshow("Original image", image)
```

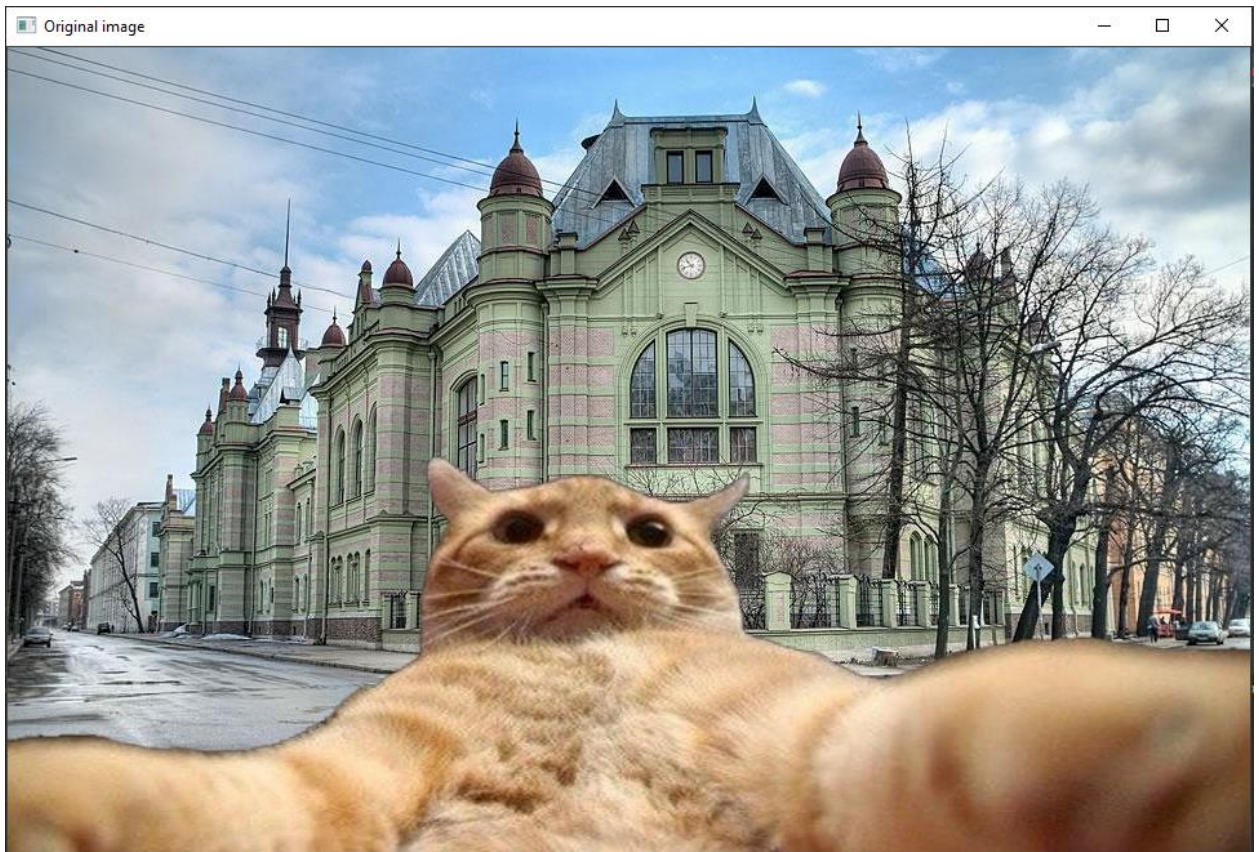


Рисунок 1 – Исходное изображение

Кластеризуем изображение и отобразим его на экране (результаты выполнения для различных значений количества кластеров представлены на рис. 2-4):

```
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER,  
10, 1.0)  
ret, label, center = cv2.kmeans(np.float32(image.reshape((-1,  
3))), k, None, criteria, 10, cv2.KMEANS_RANDOM_CENTERS)  
center = np.uint8(center)  
clusters = center[label.flatten()].reshape((image.shape))  
cv2.imshow('Clusters', clusters)
```

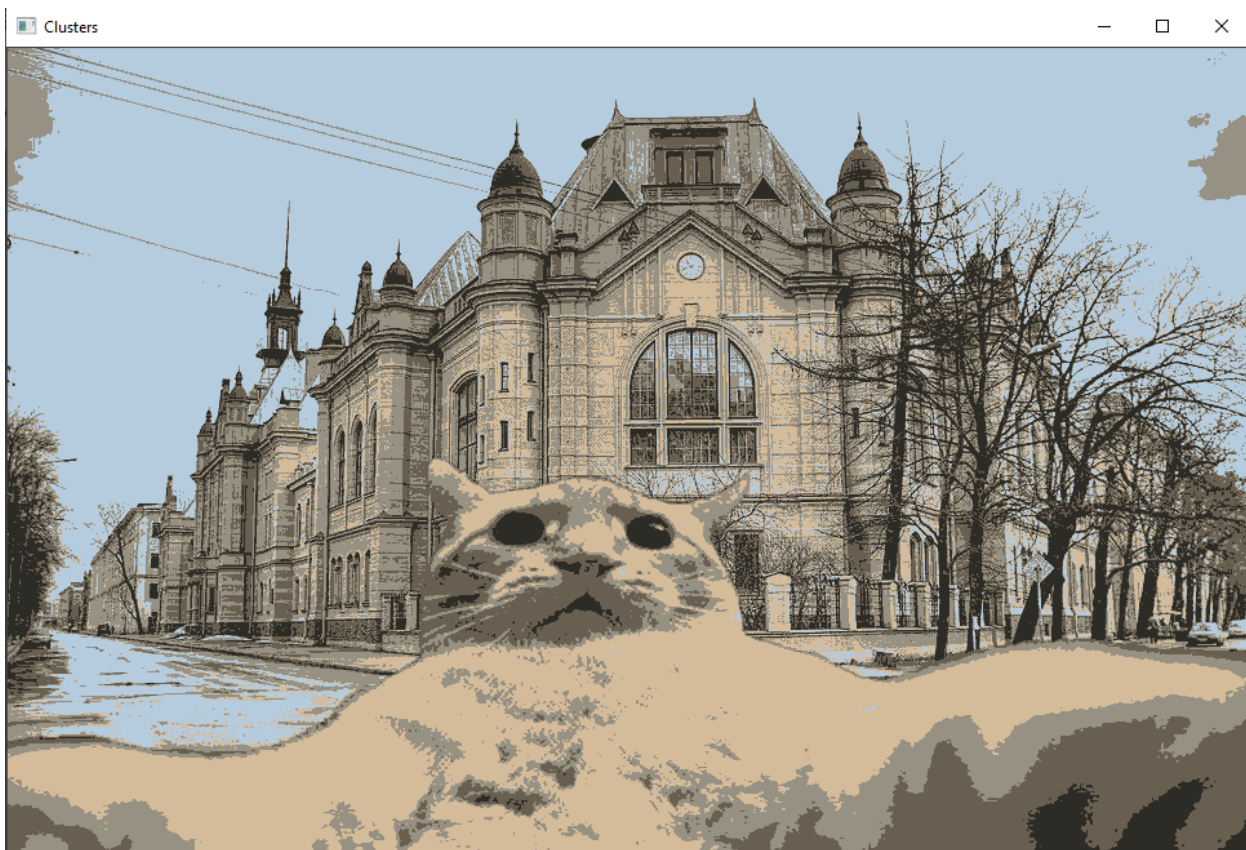


Рисунок 2 – Сегментированное изображение с $k = 5$

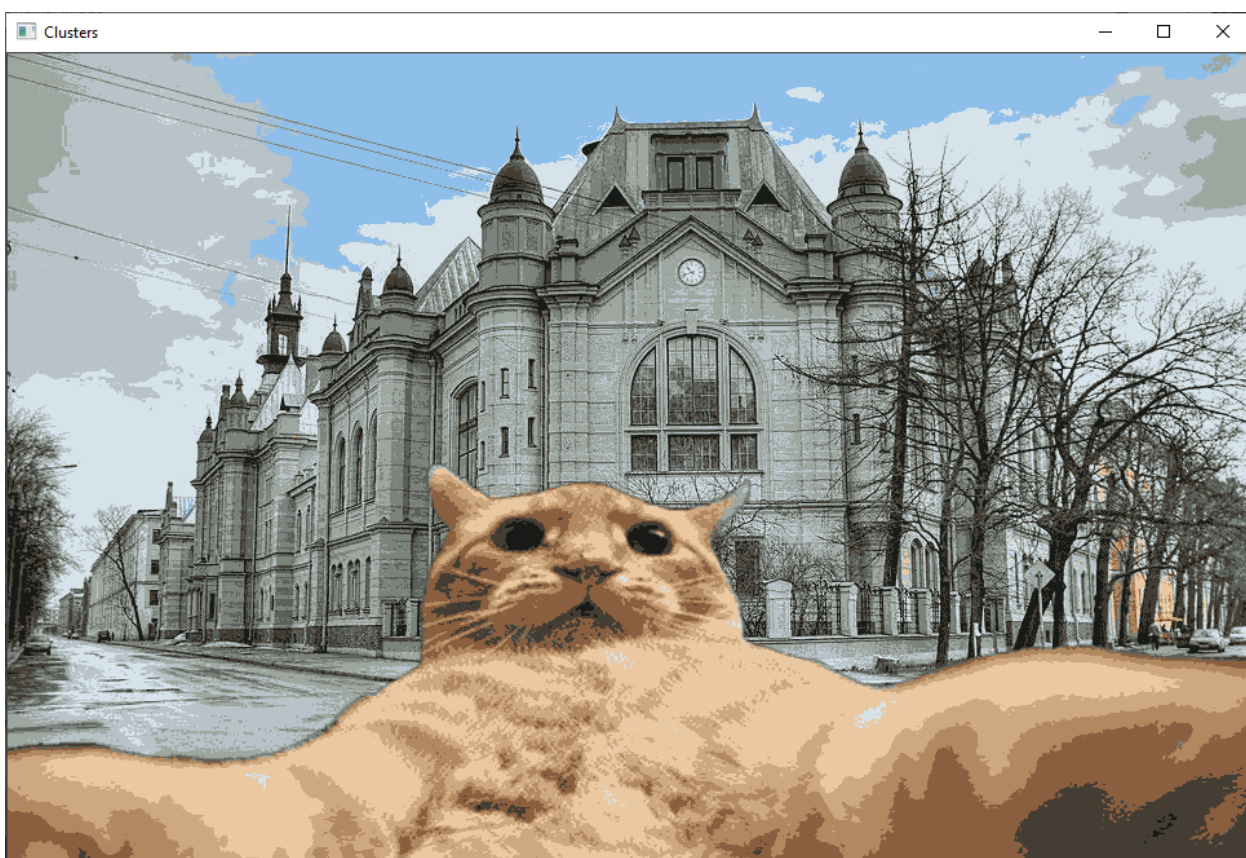


Рисунок 3 – Сегментированное изображение с $k = 12$

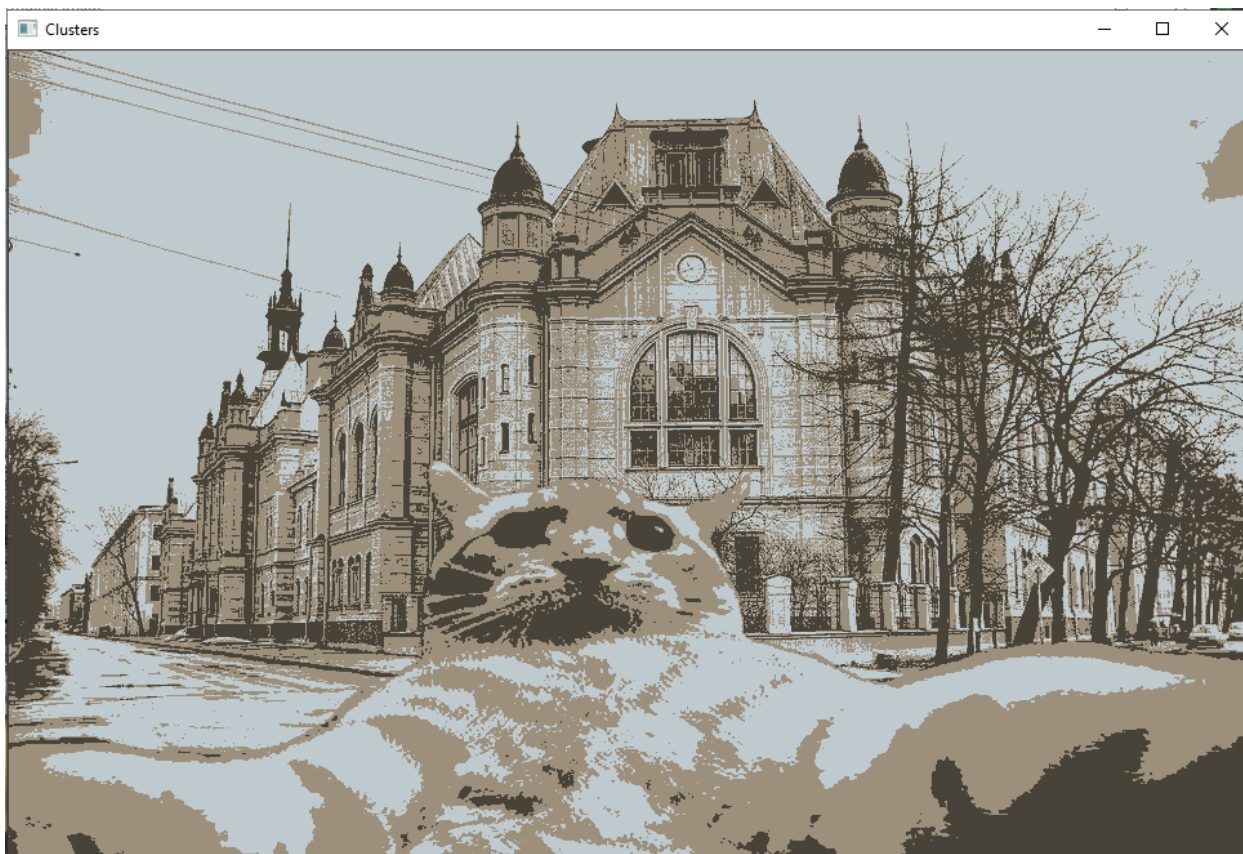


Рисунок 4 – Сегментированное изображение с $k = 3$

Полученное изображение сохраним с именем «result.bmp»:

```
cv2.imwrite('result.bmp', image)
```

Полный код программы представлен в приложении А.

Выводы.

Был изучен процесс сегментирования данных с использованием алгоритма кластеризации K-Means.

Была реализована на языке Python с использованием библиотеки OpenCV программа, выполняющая сегментацию/кластеризацию изображения по цвету с использованием метода K-Means.

ПРИЛОЖЕНИЕ А

КОД ПРОГРАММЫ

```
import cv2
import numpy as np
import sys

filename = sys.argv[1]
k = int(sys.argv[2])

image = cv2.imread(filename)
cv2.imshow("Original image", image)

criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 10,
1.0)
ret, label, center = cv2.kmeans(np.float32(image.reshape((-1, 3))), k,
None, criteria, 10, cv2.KMEANS_RANDOM_CENTERS)
center = np.uint8(center)
clusters = center[label.flatten()].reshape((image.shape))
cv2.imshow('Clusters', clusters)

cv2.imwrite('result.bmp', image)
cv2.waitKey(0)
```