|  |  |
| --- | --- |
| тов_знак_прав | МИНОБРНАУКИ РОССИИ  федеральное государственное автономное образовательное учреждение  высшего образования  «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)»  **(СПбГЭТУ «ЛЭТИ»)** |

Кафедра САУ

ОТЧЕТ

по лабораторно-практической работе № 3

По курсу: «Техническое зрение»

Тема: Введение в OpenCV.

|  |  |  |
| --- | --- | --- |
| Студенты гр. 6491 |  | Зверев Г. Ю.  Михайленко Д. М. |
| Преподаватель |  | Моклева К. А. |

Санкт-Петербург

2020 г.

Цель работы: научиться выполнять простые операции над изображениями.

Основные задания:

1) Скачайте любое полноцветное изображение. Откройте его в окне с

названием “RGB”. Откройте его же так, чтобы оно было

представлено в оттенках серого. Откройте его в окне с названием

“GRAYSCALE”. Сохраните рядом с исходным изображением то же

изображение в оттенках серого.

2) Откройте изображение в оттенках серого. Для каждого пикселя

инвертируйте его значение (Если возможные значения - диапазон от

0 до 255, то противоположным значением для 0 является 255, для 1 -

254, для 2 - 253, и так далее).

3) Откройте полноцветное изображение. Поменяйте местами значения

красного и зеленого каналов.

Дополнительные задания:

1) Создайте пустое полноцветное изображение. Нарисуйте флаг

России.

2) Создайте пустое полноцветное изображение. Проверьте, в каком

порядке записываются в ndarray красный, зеленый и синий каналы.

Проведите эксперимент: вычислите, какое сочетание значений

каналов дает различные цвета. Приведите примеры для как минимум

7 цветов.

Код для основных заданий:

from cv2 import cv2

import numpy as np

import pprint

print('OpenCD verison:', cv2.\_\_version\_\_)

# Загрузить изображение

img = cv2.imread('default.png')

gray\_img = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

# Сохранить изображение как 'имя.png'

cv2.imwrite('gray.png', gray\_img)

gray\_img2 = np.copy(gray\_img)

# Проходим по каждому элементу массива и инвертируем его в диапазоне 0...255

for i in range(len(gray\_img2)):

for j in range(len(gray\_img2[i])):

gray\_img2[i][j] = 255 - gray\_img2[i][j]

rgb\_img = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)

print(rgb\_img.shape)

# Можно обойтись и без создания массива из нулей такого же размера, так как merge сделает своё дело

# new = np.zeros(rgb\_img.shape)

red = rgb\_img[:,:,0]

print("RED: ", red)

green = rgb\_img[:,:,1]

print("GREEN: ", green)

blue = rgb\_img[:,:,2]

print("BLUE :", blue)

# # merge - слияние, здесь цвета в порядке BGR

new = cv2.merge((red, green, blue))

# cv2.imshow() - метод, который показывает изображение в отдельном окне

# Дефолтное изображение; серое; инверсия серого

cv2.imshow('Default', img)

cv2.imshow('Gray', gray\_img)

cv2.imshow('InversionGray', gray\_img2)

# Изображение, где поменяны местами каналы

cv2.imshow('BGR -> RGB', new)

cv2.waitKey(0)

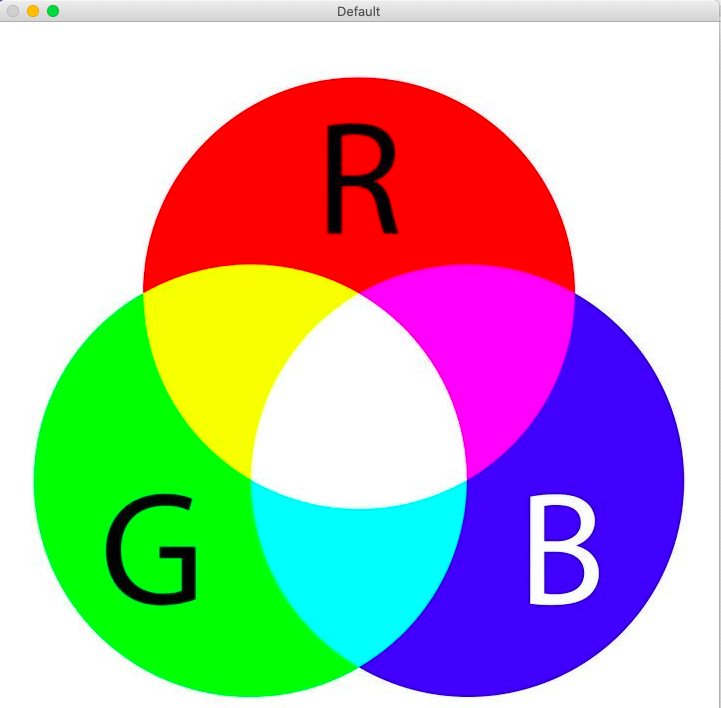
cv2.destroyAllWindows()

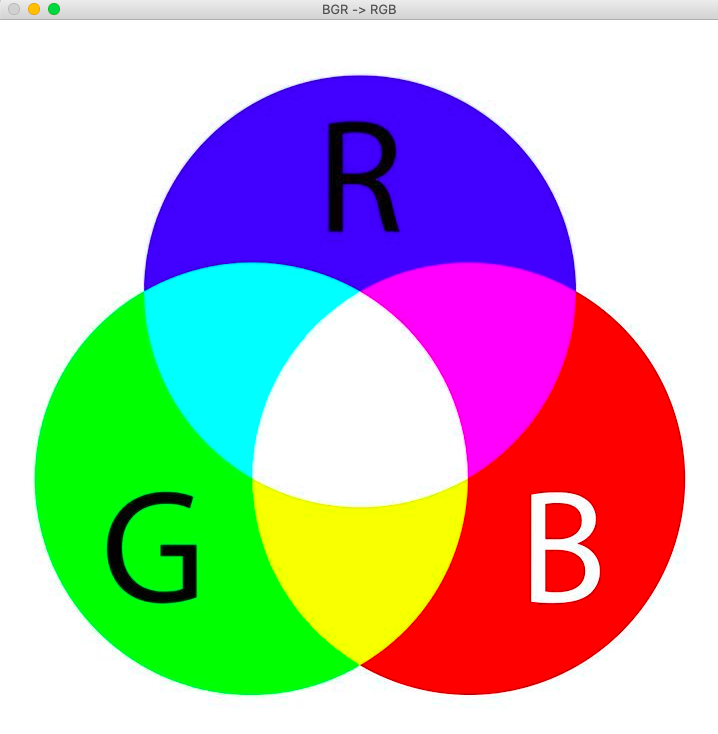
# Сохранит два изображения после закрытия:

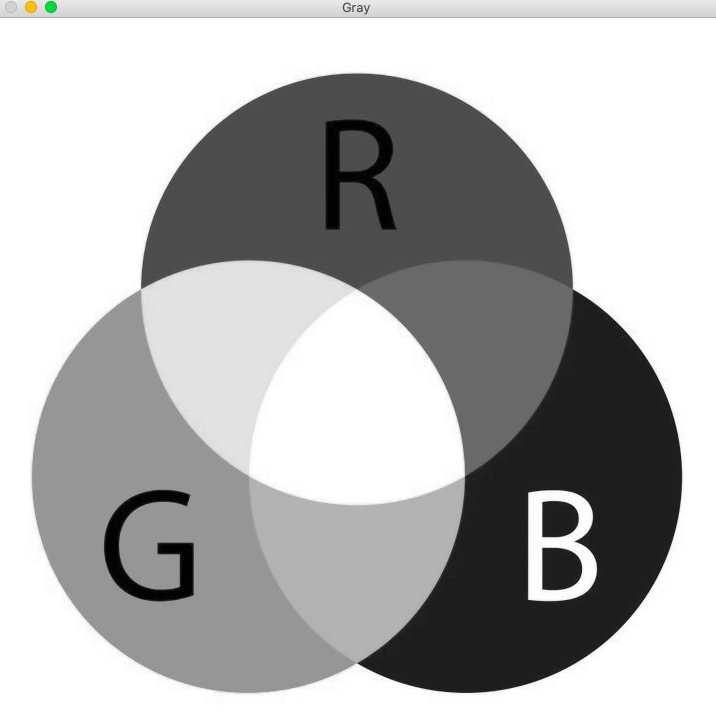
cv2.imwrite('inversion.png', gray\_img2)

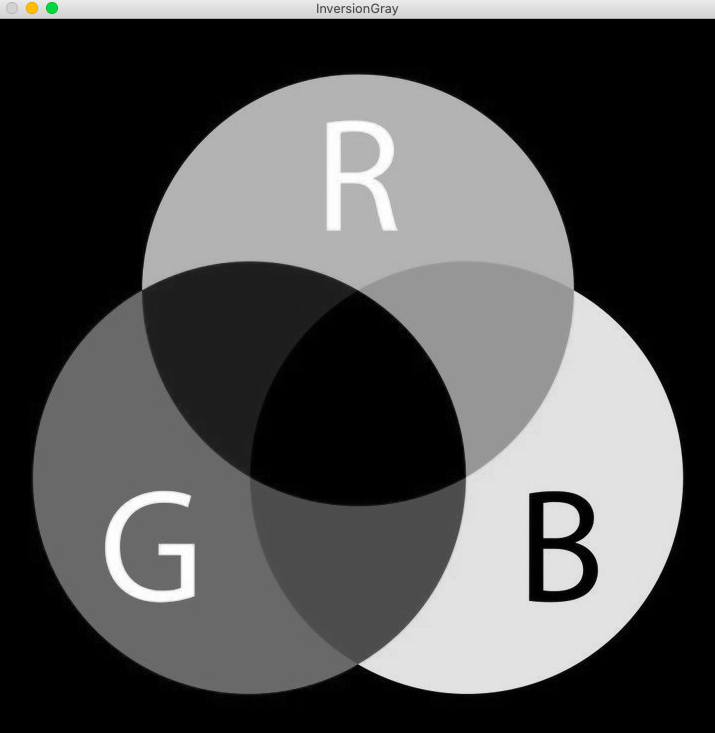
cv2.imwrite('BGRtoRGB.png', new)

Изображения:









Код для дополнительных заданий:

from cv2 import cv2

import numpy as np

img = np.zeros((512,512,3), np.uint8)

# img2 = cv2.cvtColor(img, cv2.COLOR\_BGR2RGB)

color\_yellow = (0,255,255)

color\_blue = (255,0,0)

cv2.rectangle(img,(0,512),(512,256),color\_yellow,-1)

cv2.rectangle(img,(0,0),(512,256),color\_blue,-1)

# np.zeroes(y, x, кол-во каналов)

img2 = np.zeros((210,512,3), dtype = np.uint8)

# в openCV цвета BGR - blue, green, red

# Синий

cv2.line(img2,(0,10),(512,10),(255,0,0),20)

# Зеленый

cv2.line(img2,(0,40),(512,40),(0,255,0),20)

# Красный

cv2.line(img2,(0,70),(512,70),(0,0,255),20)

# Фиолетовый

cv2.line(img2,(0,100),(512,100),(125,0,125),20)

# Коричневый

cv2.line(img2,(0,130),(512,130),(0,60,125),20)

# Серый

cv2.line(img2,(0,160),(512,160),(100,100,100),20)

# Абрикосовый

cv2.line(img2,(0,190),(512,190),(100,150,250),20)

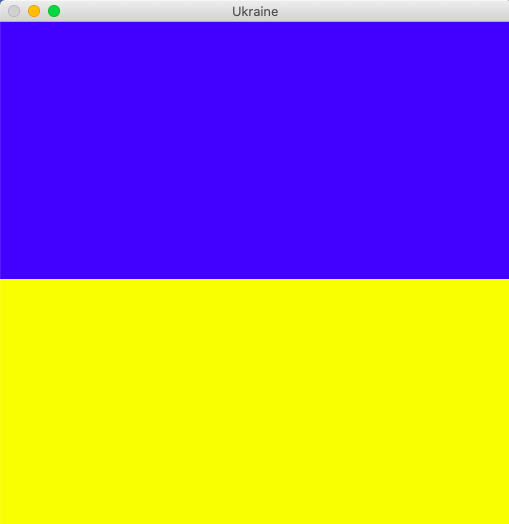
cv2.imshow('Ukraine', img)

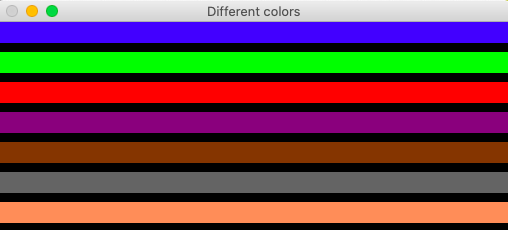
cv2.imshow('Different colors', img2)

cv2.waitKey(0)

cv2.destroyAllWindows()

Изображения:





Вывод: в ходе лабораторной работы были выполнены все основные и все дополнительные задания, получены новые знания по работе с библиотекой OpenCV.