

**Федеральное государственное бюджетное образовательное  
учреждение высшего образования «Московский  
государственный университет имени М. В. Ломоносова»**

**Кафедра вычислительной механики**

**ОТЧЁТ ПО ЗАДАЧЕ НА РАБОТУ С  
ИЗОБРАЖЕНИЯМИ ПО ТЕМЕ:  
ИЗМЕНЕНИЕ ЯРКОСТИ ВМР-ИЗОБРАЖЕНИЯ**

**Преподаватель: Почеревин Роман Владимирович  
Студент 223 группы: Скворцов Андрей Сергеевич**

Москва  
2024

## Отчёт по работе с BMP-изображениями в Python-3

*Задача.* Программа должна загрузить изображения из графических файлов InputFile1 и InputFile2 (размеры входных файлов должны совпадать), изменить яркость каждого пиксела первого изображения так, чтобы она равнялась бы яркости соответствующего пиксела второго изображения и вывести получившееся изображение в графический файл OutputFile. Под яркостью пиксела подразумевается сумма компонент всех трех компонент пиксела.

*Решение.* Используя библиотеку PIL, можно решить эту задачу эффективнее, чем, например, imageio:

```
from PIL import Image, ImageDraw
```

### 1 Загрузка изображений

Пусть изображения заданы внутри программы (на примере - "ironman.bmp" "osuzdayu.bmp"). Выгрузим их и посчитаем длину и ширину. На всякий случай, в выходном изображении сделаем минимальную из двух длин и ширин:

```
image = Image.open("ironman.bmp")
image2 = Image.open("osuzdayu.bmp")
draw = ImageDraw.Draw(image)
width = image.size[0]
height = image.size[1]
width2 = image2.size[0]
height2 = image2.size[1]
w=min(width, width2)
h=min(height, height2)
pix = image.load()
pix2 = image2.load()
```

### 2 Обработка изображений

Теперь - логика программы. Найдем яркость пикселя второго изображения и присвоим ей значение *factor*. Затем найдем яркость пикселя первого изображения и присвоим ей значение *factor1*.

```
for i in range(w):
    for j in range(h):
        a = pix2[i, j][0]
        b = pix2[i, j][1]
        c = pix2[i, j][2]
        factor = (a+b+c)//3
        a1 = pix[i, j][0]
        b1 = pix[i, j][1]
        c1 = pix[i, j][2]
        factor1 = (a1+b1+c1)//3
```

### 3 Перестройка и рисование изображения

Теперь перестроим первое изображение так, чтобы яркость его пикселей равнялась яркости яркости соответствующих пикселей второго.

```
a1 = round(pix[i, j][0] * (factor / factor1))  
b1 = round(pix[i, j][1] * (factor / factor1))  
c1 = round(pix[i, j][2] * (factor / factor1))
```

```
if (a1 < 0):  
    a1 = 0
```

```
if (b1 < 0):  
    b1 = 0
```

```
if (c1 < 0):  
    c1 = 0
```

```
if (a1 > 255):  
    a1 = 255
```

```
if (b1 > 255):  
    b1 = 255
```

```
if (c1 > 255):  
    c1 = 255
```

```
draw.point((i, j), (a1, b1, c1))
```

Сохраним изображение и очистим память от элемента *draw*:

```
image.save("result.bmp "BMP")  
del draw
```

### 4 Примеры



Рис. 1: Первая картинка



Рис. 2: Вторая картинка



Рис. 3: Результат