

Работа 3. Яркостные преобразования изображений

автор: Попов Д.В.

дата: 2022-02-26T01:48:30

https://github.com/Popivzanin/opencv/tree/master/popov_d_v/prj.labs/lab03

Задание

1. В качестве тестового использовать изображение data/cross_0256x0256.png
2. Сгенерировать нетривиальную новую функцию преобразования яркости (не стоит использовать линейную функцию, гамму, случайная).
3. Сгенерировать визуализацию функции преобразования яркости в виде изображения размером 512x512, черные точки а белом фоне.
4. Преобразовать пиксели grayscale версии тестового изображения при помощи LUT для сгенерированной функции преобразования.
5. Преобразовать пиксели каждого канала тестового изображения при помощи LUT для сгенерированной функции преобразования.
6. Результаты сохранить для вставки в отчет.

Результаты



Рис. 1. Исходное тестовое изображение



Рис. 2. Тестовое изображение greyscale



Рис. 3. Результат применения функции преобразования яркости для greyscale

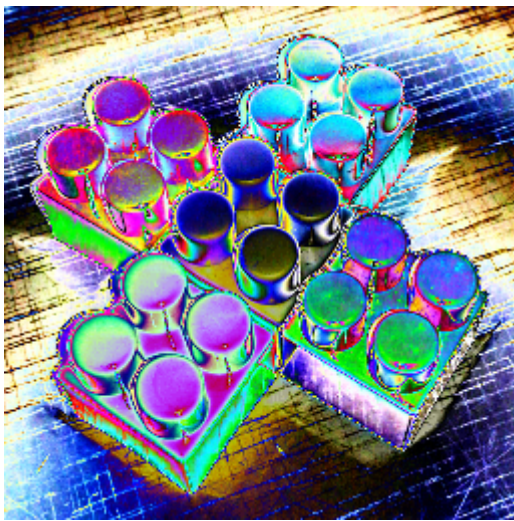


Рис. 4. Результат применения функции преобразования яркости для каналов

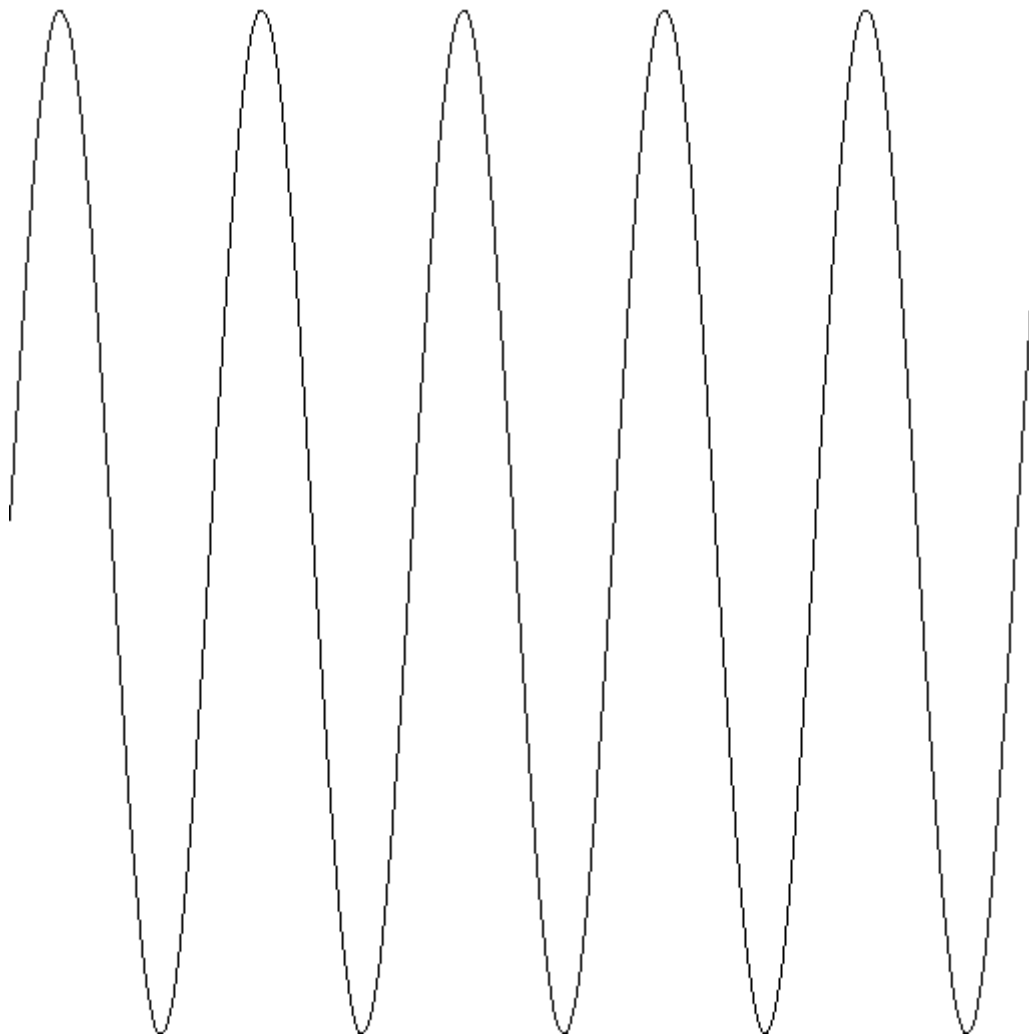


Рис. 5. Визуализация функции яркостного преобразования

Текст программы

```
#include <opencv2/opencv.hpp>
#include <cmath>
using namespace cv;
int main() {
    Mat src_gray_img = imread("cross_0256x0256.png", IMREAD_GRAYSCALE);
    Mat src_rgb_img = imread("cross_0256x0256.png");
    imwrite("lab03_gre.png", src_gray_img);
    imwrite("lab03_rgb.png", src_rgb_img);

    Mat lookUpTable(1, 256, CV_8U);
    uchar* p = lookUpTable.ptr();
    for (int i = 0; i < 256; ++i)
        p[i] = 128 * (sin(i * 0.125) + 1);
    Mat img;
    LUT(src_rgb_img, lookUpTable, img);
    imwrite("lab03_rgb_res.png", img);

    LUT(src_gray_img, lookUpTable, img);
    imwrite("lab03_gre_res.png", img);

    Mat src_func(512, 512, CV_8UC1);
    for (int i = 1; i < 256; i++)
    {
        line(
            src_func,
```

```
        Point(i - 1 << 1, 511 - (((int)p[i - 1]) << 1)),  
        Point(i << 1, 511 - (((int)p[i]) << 1)), 0, 1, 0);  
    }  
    imwrite("lab03_viz_func.png", src_func);  
    waitkey();  
}
```