Работа 1. Исследование гамма-коррекции

автор: Кочнев Р.Ю.

url: https://gitlab.com/mind2cloud/kochnev_r_u/-/tree/master/lab_1

Задание

1. Сгенерировать серое тестовое изображение I_1 в виде прямоугольника размером 768x60 пикселя с плавным изменение пикселей от черного к белому, одна градация серого занимает 3 пикселя по горизонтали.
2. Применить к изображению I_1 гамма-коррекцию с коэффициентом из интервала 2.2-2.4 и получить изображение G_1.
3. Сгенерировать серое тестовое изображение I_2 в виде прямоугольника размером 768x60 пикселя со ступенчатым изменением яркости от черного к белому (от уровня 5 с шагом 10), одна градация серого занимает 30 пикселя по горизонтали.
4. Применить к изображению I_2 гамма-коррекцию с коэффициентом из интервала 2.2-2.4 и получить изображение G_2.
5. Показать визуализацию результатов в виде одного изображения, об

Результаты

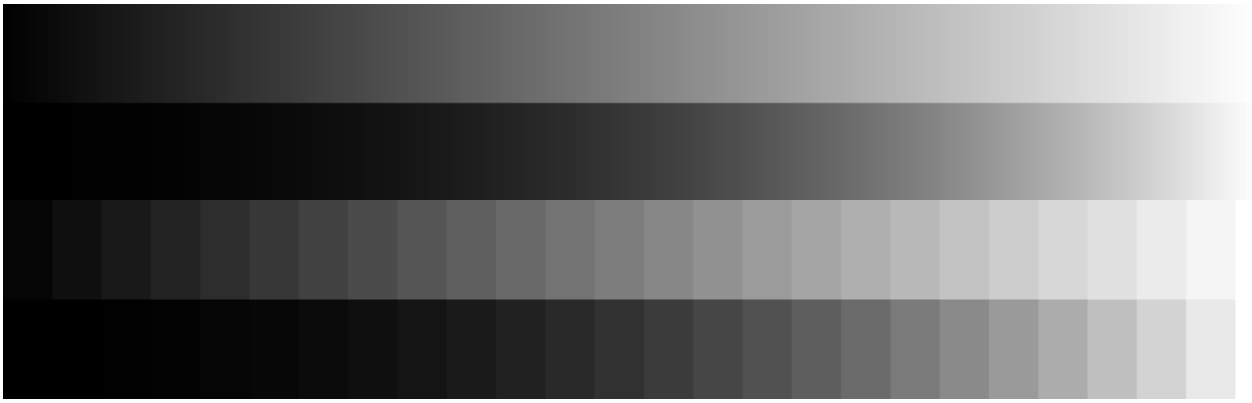

Рис. 1. Результаты работы программы (сверху вниз I_1, G_1, I_2, G_2)

Текст программы

#include <opencv2/opencv.hpp>
using namespace cv;

int main() {

```
int w = 786;
int h =60;

Mat image(Mat::zeros(h, w, CV_8U));
Mat resultedImage(Size(w, 2 * h), CV_8U);
for (int col = 0; col < w; ++col) {
```

```
    for (int row = 0; row < h; ++row) {
        image.at<uchar>(Point(col, row)) = col / 3;
    }
}

image.copyTo(resultedImage(Rect(0, 0, w, h)));
image.convertTo(image, CV_64F, 1.0 / 256);
cv::pow(image, 2.2, image);
image.convertTo(image, CV_8U, 256);
image.copyTo(resultedImage(Rect(0, h, w, h)));

imshow("result1", resultedImage);

//SECOND
Mat image_2(Mat::zeros(h, w, CV_8U));
Mat resultedImage_2(Size(w, 2 * h), CV_8U);
for (int col = 0; col < w; ++col) {
    for (int row = 0; row < h; ++row) {
        image_2.at<uchar>(Point(col, row)) = 5 + col / 30*10;
    }
}

image_2.copyTo(resultedImage_2(Rect(0, 0, w, h)));
image_2.convertTo(image_2, CV_64F, 1.0 / 256);
cv::pow(image_2, 2.3, image_2);
image_2.convertTo(image_2, CV_8U, 256);
image_2.copyTo(resultedImage_2(Rect(0, h, w, h)));
```

```
imshow("result2", resultedImage_2);

Mat sm[2] = { resultedImage, resultedImage_2 };
Mat saveResult;
vconcat(sm, 2, save);
imwrite("rlt.png", saveResult);
```

}