

Министерство науки и высшего образования Российской Федерации  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
ИТМО»**

Факультет Систем Управления и Робототехники

**Дисциплина:** Обработка цифровых изображений

ОТЧЁТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ №1  
**Гистограммы, профили и проекции**

Выполнил студент

Топольницкий А.А.

Группа

R4134с

Преподаватель

Шаветов С.В.

Санкт-Петербург

2023 г.

### **Цель работы:**

Освоение основных яркостных и геометрических характеристик изображений и их использование для анализа изображений.

### **Краткое теоретическое обоснование:**

В работе используются понятия пикселя, гистограммы изображения, профиля и проекции на ось.

Пиксель характеризуется тремя параметрами – геометрическим положением  $(x, y)$  и яркостным параметром – интенсивностью  $I$ .

Гистограмма – это распределение частоты встречаемости пикселей одинаковой яркости на изображении.

Профиль вдоль прямой – это функция интенсивности изображения, распределённого вдоль данной линии.

Проекция на ось – это сумма интенсивностей пикселей изображения, взятая в направлении перпендикулярном данной оси.

В работе используются следующие виды операций с изображениями:

- Арифметические операции;
- Растяжение динамического диапазона;
- Равномерное преобразование;
- Экспоненциальное преобразование;
- Преобразование по закону Рэлея;
- Преобразование по закону степени  $2/3$ ;
- Гиперболическое преобразование

### **Ход работы**

Лабораторная работы выполнялась в MATLAB.

#### **Листинг 1. Построение гистограммы изображения**

```
clc
clear all
I = imread("evening.jpg");
[numRows, numCols, Layers] = size(I);

figure;
```

```
subplot(2,2,1)
imhist(I(:, :, 1))
title('Красный')
subplot(2,2,2)
imhist(I(:, :, 2))
title('Зелёный')
subplot(2,2,3)
imhist(I(:, :, 3))
title('Синий')
subplot(2,2,4)
imshow(I)
title('Оригинал')
```

*Комментарий:* для цветного RGB – изображения необходимо построить три гистограммы по каждому цвету, при этом элементом диаграммы является сумма пикселей изображения с одинаковой яркостью.

В коде считываем изображение с помощью *imread*, а с помощью *imhist* строим гистограмму для каждого слоя.



Рисунок 1. Оригинальное изображение

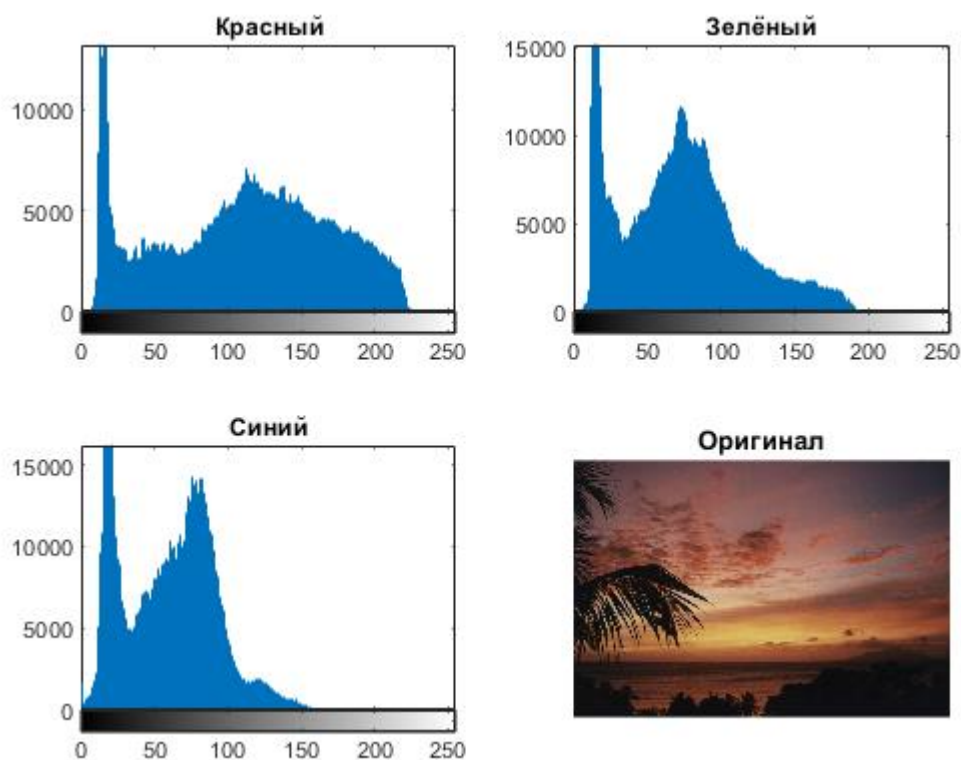


Рисунок 2. Гистограммы оригинального изображения по цветам и само изображение

**Листинг 2.** *Пример арифметической операции*

```
for i = 1:1: Layers
    for j = 1:1: numRows
        for k = 1:1: numCols
            Inew(j,k,i) = I(j,k,i) + 30;
        end
    end
end
```

```
figure;
subplot(2,3,1)
imhist(Inew(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew(:, :, 3))
title('Синий')
```

```

subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew)
title('После сдвига')

```

*Комментарий:* по гистограммам с прошлого шага видно, что бóльшая часть пикселей лежит в тёмной части. Можно сделать картинку ярче, если к каждому пикселю прибавить число, например, число 30. Тогда изображение станет посветлее. Именно это и сделано в листинге выше.

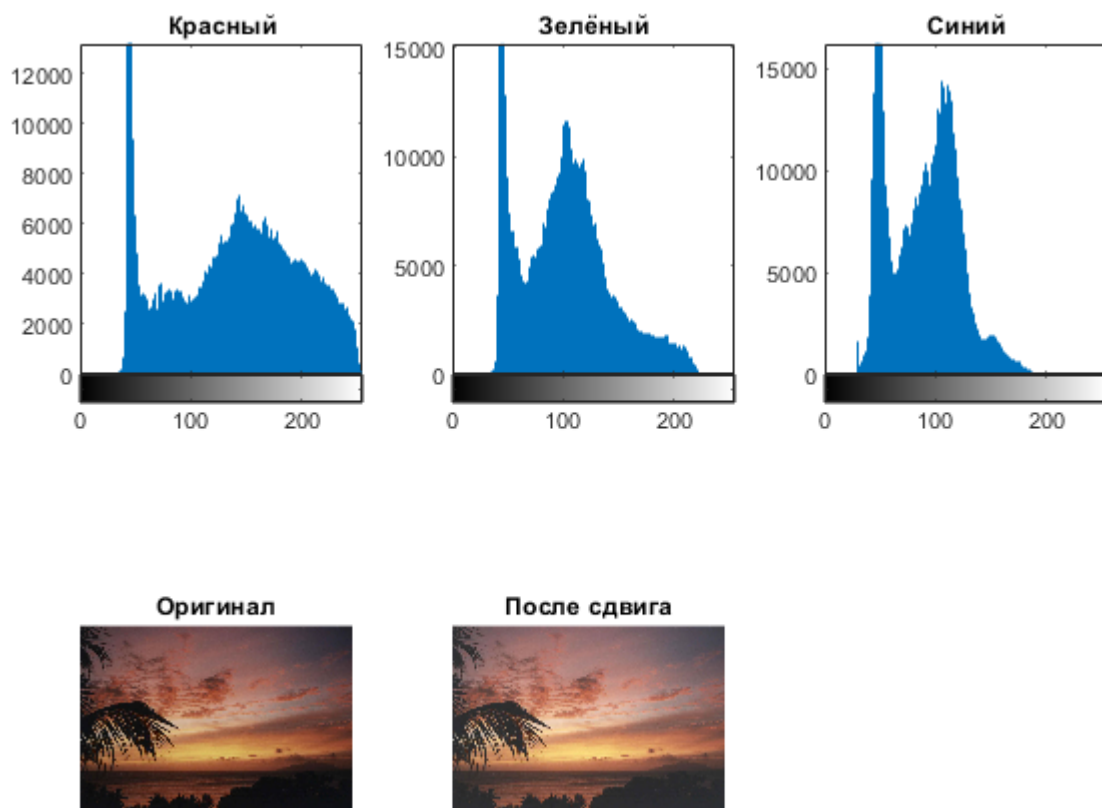


Рисунок 3. Применение сдвига вправо для изображения. Оригинал и последствия применения

**Листинг 3.** *Нелинейное растяжение динамического диапазона*

```

alpha = 0.8;
for k = 1:1:Layers
    Imin = min(min(I(:,k)));
    Imax = max(max(I(:,k)));
    for i = 1:1:numRows

```

```

    for j = 1:1:numCols
        Inew_din(i,j,k) = ((double(I(i,j,k) - Imin) / double(Imax - Imin))) ^ alpha;
    end
end
end

figure;
subplot(2,3,1)
imhist(Inew_din(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_din(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_din(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_din)
title('После растяжения')

```

*Комментарий:* данный способ применяется в случае, когда интенсивности пикселей находятся в узком динамическом диапазоне, растягивая этот диапазон. Преобразование выполняется по следующей формуле:

$$I_{new} = \left( \frac{I - I_{min}}{I_{max} - I_{min}} \right)^{\alpha}, \quad (1)$$

где  $I$  и  $I_{new}$  – массивы значений интенсивностей исходного и нового изображений соответственно;  $I_{min}$  и  $I_{max}$  – минимальное и максимальное значения интенсивностей исходного изображения соответственно;  $\alpha$  – коэффициент нелинейности.

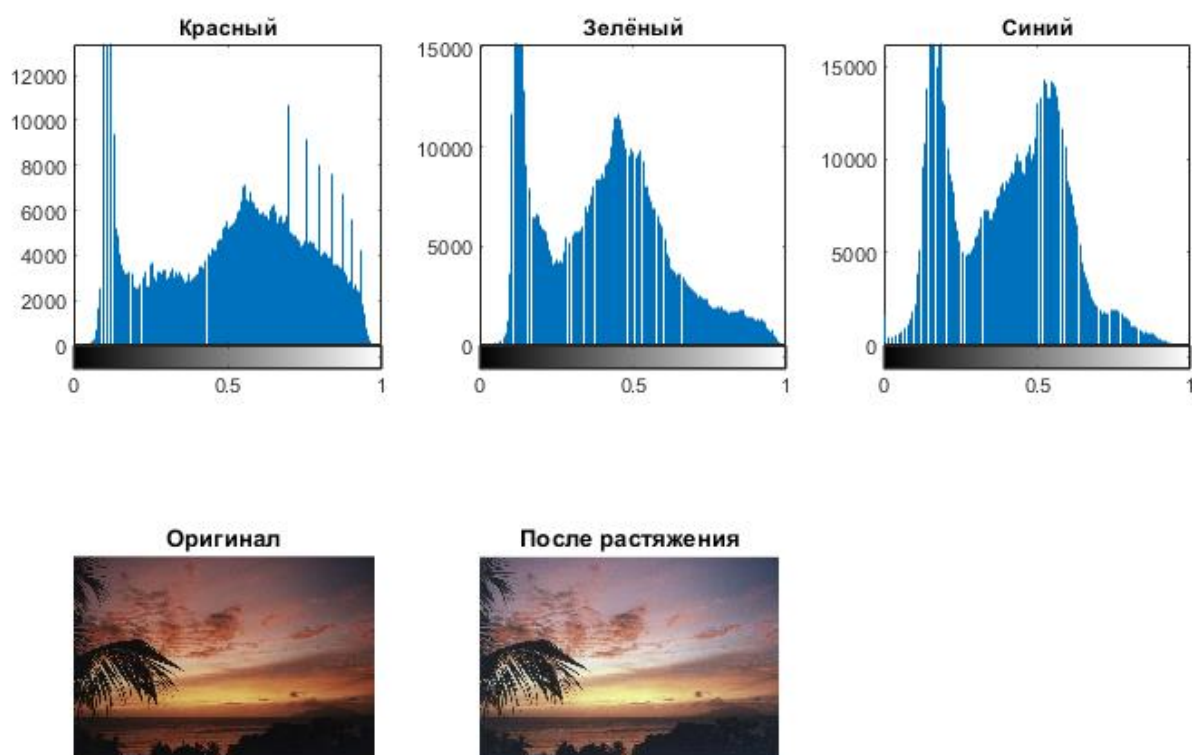


Рисунок 4. Применение нелинейного растяжения динамического диапазона с коэффициентом  $\alpha = 0.8$

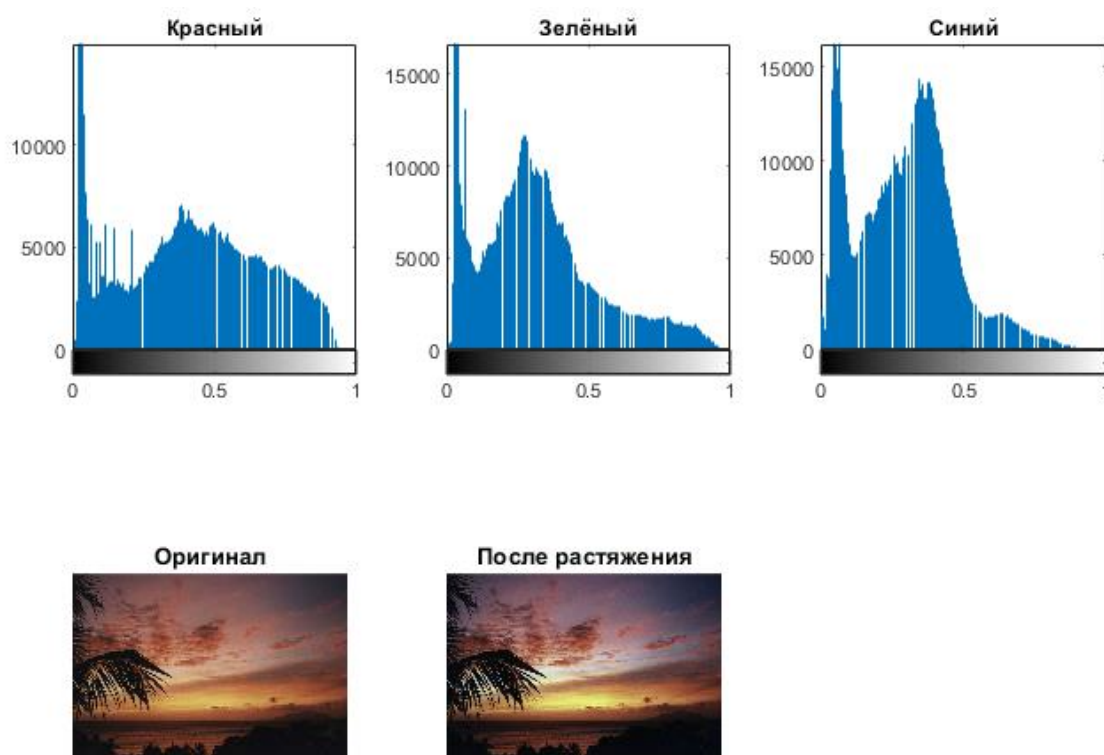


Рисунок 5. Применение нелинейного растяжения динамического диапазона с коэффициентом  $\alpha = 1.3$

#### Листинг 4. Равномерное преобразование

```
for k = 1:1:Layers
    H = imhist(I(:,:,k));
    CH = cumsum(H) / (numCols * numRows);
    Imin = min(min(I(:,:,k)));
    Imax = max(max(I(:,:,k)));
    for i = 1:1:numRows
        for j = 1:1:numCols
            ind = I(i,j,k);
            if ind == 0 % чтобы не ругался на ошибку
                ind = 1; % "индексы должны быть целыми положительными"
            end
            Inew_uni(i,j,k) = double(Imax-Imin) * CH(ind) + Imin/255;
        end
    end
end

figure;
subplot(2,3,1)
imhist(Inew_uni(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_uni(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_uni(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_uni)
title('После равномерного преобразования')
```

*Комментарий:* вводится понятие кумулятивной гистограммы – это гистограмма, в которой по вертикальной оси указаны не значения для каждой интенсивности, а значение для этой интенсивности + все меньшие



интенсивности, то есть постоянно неубывающая функция. Задаётся формулой ниже.

$$P(I) = \sum_{m=0}^i \text{Hist}(m) \quad (2)$$

Кумулятивную гистограмму можно вычислить с помощью `cumsum()`, само же равномерное преобразование работает по формуле:

$$I_{new} = (I_{max} - I_{min}) * P(I) + I_{min} \quad (3)$$

В коде послойно считаем кумулятивную диаграмму, добавляем отладочный момент, когда значение интенсивности равно 0, чтобы MATLAB не ругался на индекс «0».

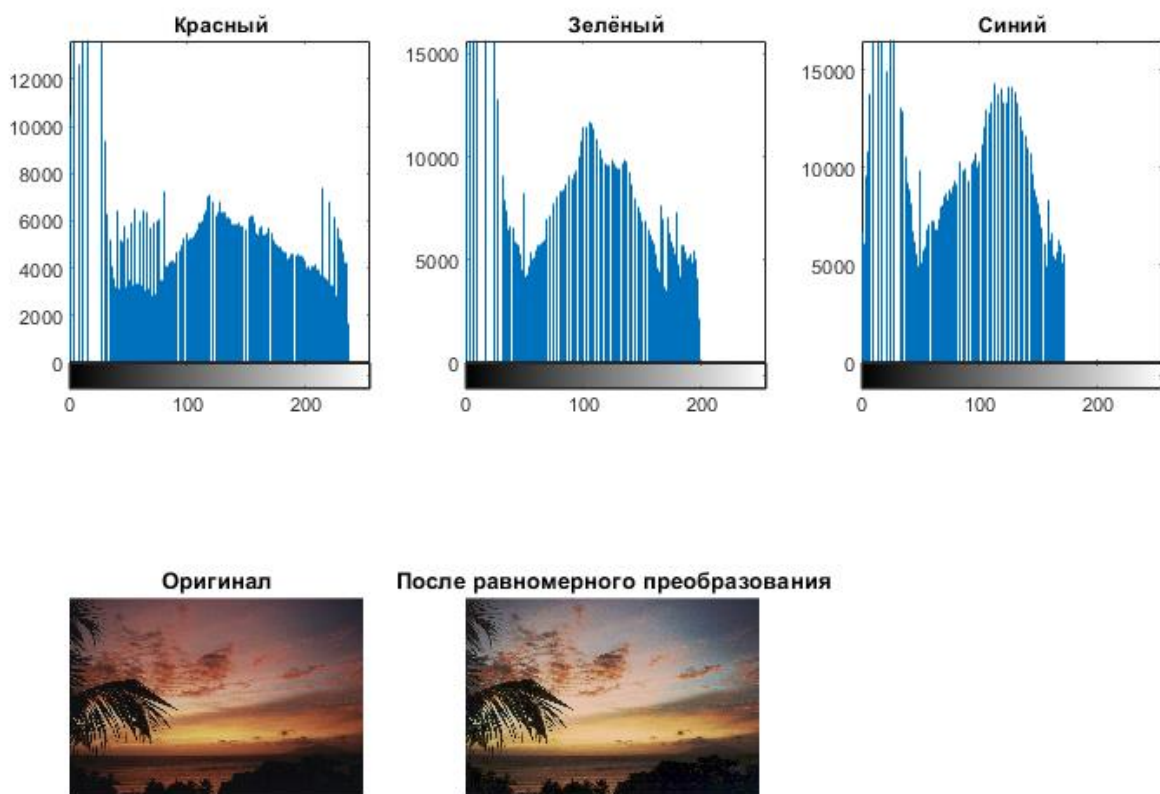


Рисунок 6. Применение равномерного преобразования

#### Листинг 5. Экспоненциальное преобразование

```
alpha = 2.5;
for k = 1:1:Layers
    H = imhist(I(:,:,k));
    CH = cumsum(H) / (numCols * numRows);
    Imin = min(min(I(:,:,k)));
```

```

I_max = max(max(I(:,:,k)));
for i = 1:1:numRows
    for j = 1:1:numCols
        ind = I(i,j,k);
        if ind == 0 % чтобы не ругался на ошибку
            ind = 1; % "индексы должны быть целыми положительными"
        end
        I_new_exp(i,j,k) = double(I_min) - 1/alpha * log(1 - CH(ind)) ;
    end
end
end
end

```

```

figure;
subplot(2,3,1)
imhist(I_new_exp(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(I_new_exp(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(I_new_exp(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(I_new_exp)
title('После экспоненциального преобразования')

```

*Комментарий:* в плане кода меняется только формула для данного преобразования:

$$I_{new} = I_{min} - \frac{1}{\alpha} * \ln (1 - P(I)) \quad (4)$$

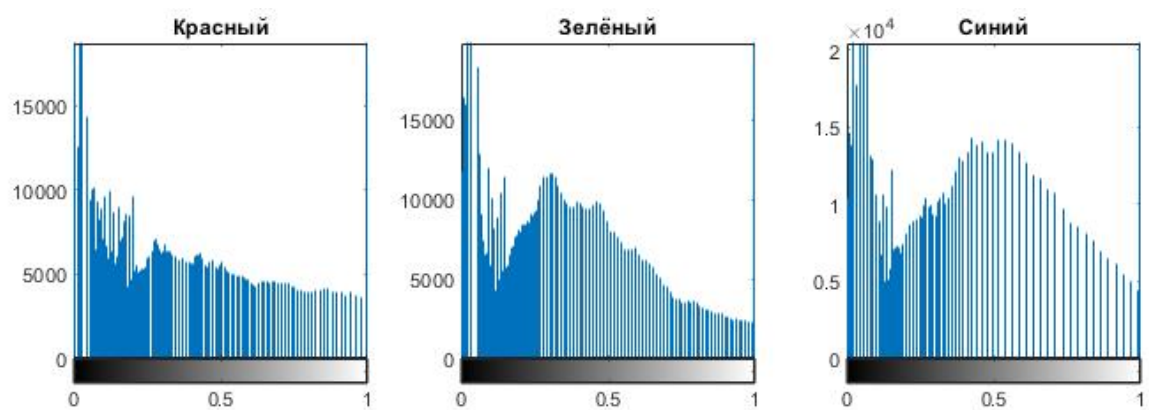


Рисунок 7. Применение экспоненциального преобразования с постоянной преобразования  $\alpha = 2.5$

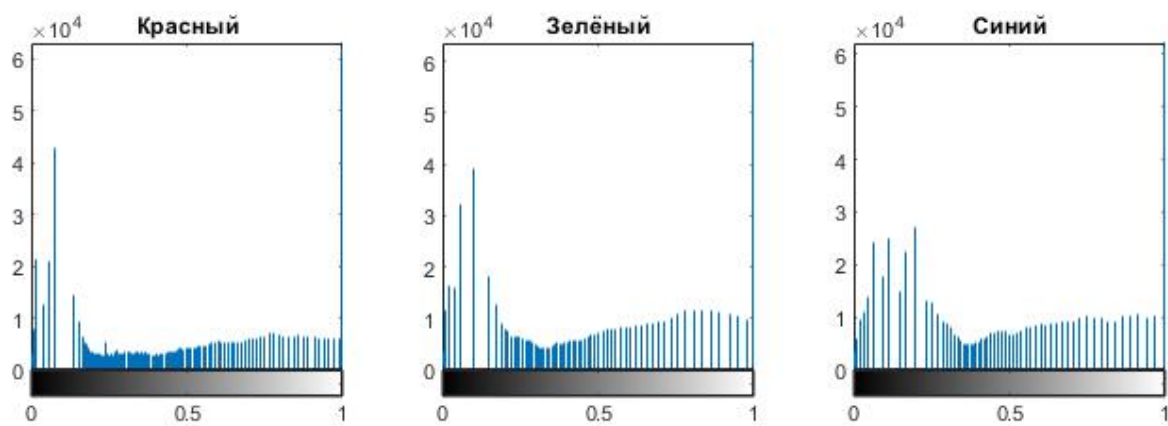


Рисунок 8. Применение экспоненциального преобразования с постоянной преобразования  $\alpha = 0.9$

### Листинг 6. Преобразование по закону Рэлея

```
alpha = 0.4;
for k = 1:1:Layers
    H = imhist(I(:,:,k));
    CH = cumsum(H) ./ (numCols * numRows);
    Imin = min(min(I(:,:,k)));
    for i = 1:1:numRows
        for j = 1:1:numCols
            ind = I(i,j,k);
            if ind == 0 % чтобы не ругался на ошибку
                ind = 1; % "индексы должны быть целыми положительными"
            end
            Inew_rel(i,j,k) = double(Imin) + (2 * alpha^2 * log(1 / (1 - CH(ind))))^(1/2);
        end
    end
end

figure;
subplot(2,3,1)
imhist(Inew_rel(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_rel(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_rel(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_rel)
title('Преобразование по закону Рэлея')
```

*Комментарий:* в плане кода меняется только формула для данного преобразования:

$$I_{new} = I_{min} + \left( 2 * \alpha^2 * \ln * \left( \frac{1}{1 - P(I)} \right) \right)^{1/2} \quad (5)$$

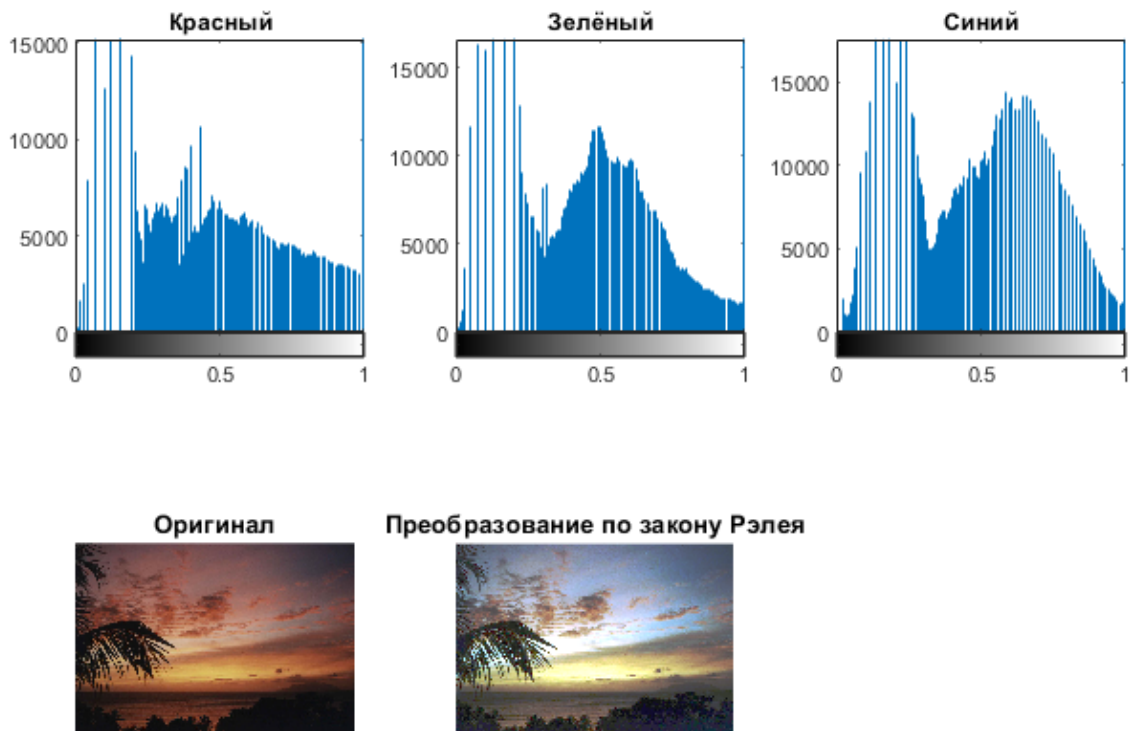


Рисунок 9. Применение преобразования по закону Рэля с постоянной преобразования  $\alpha = 0.4$

**Листинг 7.** Преобразование по закону степени  $2/3$

```
for k = 1:1:Layers
    H = imhist(I(:,:,k));
    CH = cumsum(H) ./ (numCols * numRows);
    for i = 1:1:numRows
        for j = 1:1:numCols
            ind = I(i,j,k);
            if ind == 0 % чтобы не ругался на ошибку
                ind = 1; % "индексы должны быть целыми положительными"
            end
            Inew_23(i,j,k) = CH(ind)^(2/3);
        end
    end
end

figure;
subplot(2,3,1)
imhist(Inew_23(:, :, 1))
title('Красный')
```

```

subplot(2,3,2)
imhist(Inew_23(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_23(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_23)
title('Преобразование по закону 2/3')

```

*Комментарий:* в плане кода меняется только формула для данного преобразования:

$$I_{new} = P(I)^{2/3} \quad (6)$$

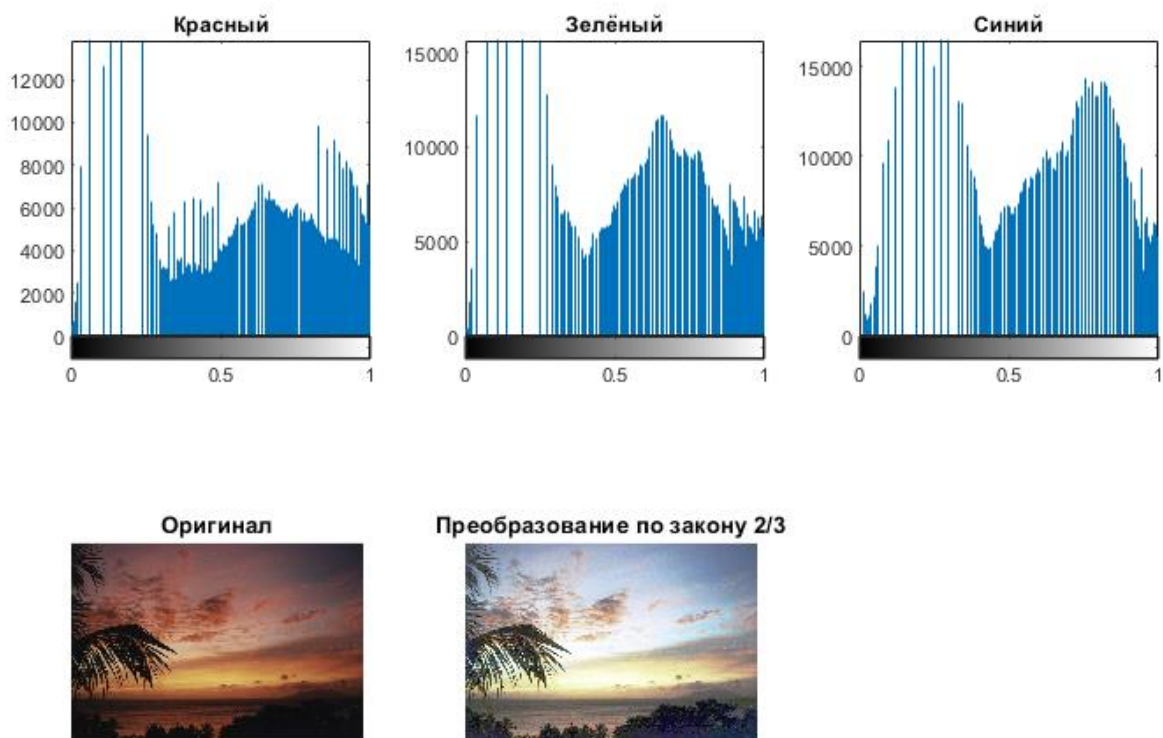


Рисунок 10. Применение преобразования по закону степени 2/3

### Листинг 8. Гиперболическое преобразование

% альфа не должна равняться 0, иначе будет чёрный квадрат  
alpha = 0.1;

```

for k = 1:1:Layers
    H = imhist(I(:,:,k));
    CH = cumsum(H) ./ (numCols * numRows);
    Imin = min(min(I(:,:,k)));
    for i = 1:1:numRows
        for j = 1:1:numCols
            ind = I(i,j,k);
            if ind == 0 % чтобы не ругался на ошибку
                ind = 1; % "индексы должны быть целыми положительными"
            end
            Inew_hip(i,j,k) = alpha^CH(ind);
        end
    end
end
figure;
subplot(2,3,1)
imhist(Inew_23(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_23(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_23(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_hip)
title('Гиперболическое преобразование')

```

*Комментарий:* в этом преобразовании важно смотреть на формулу:

$$I_{new} = \alpha^{P(I)} \quad (7)$$

Обычно  $\alpha$  равняется минимальному значению интенсивности элементов исходного изображения. Однако, если  $\alpha = 0$ , то всё изображение будет представлять собой чёрный цвет. Чтобы этого не произошло, следует взять  $\alpha$  немного больше.

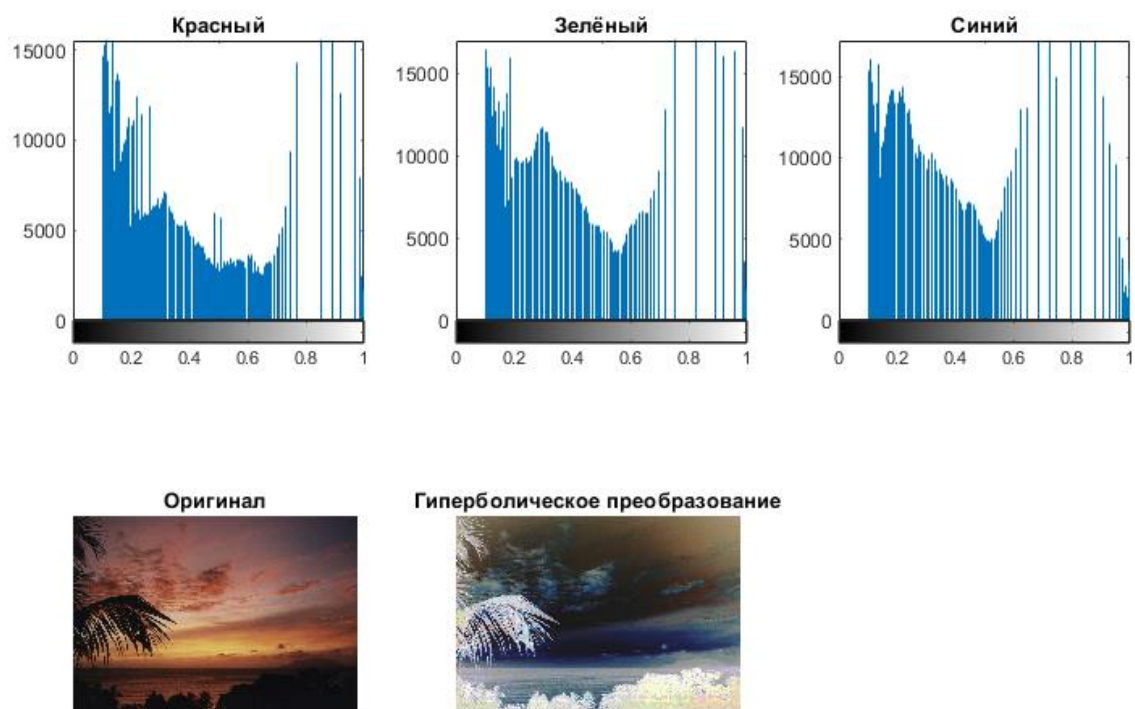


Рисунок 11. Результат применения гиперболического преобразования

*Теперь рассмотрим применение встроенных функций*

**Листинг 9.** *Повышение контрастности с помощью `imadjust`*

```
Inew_imag = imadjust(I,[0.1 0.7]);
```

```
figure;
subplot(2,3,1)
imhist(Inew_imag(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_imag(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_imag(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_imag)
title('Повышение контрастности')
```



*Комментарий:* функция *imadjust* повышает контрастность изображения, изменяя диапазон интенсивностей исходного изображения.

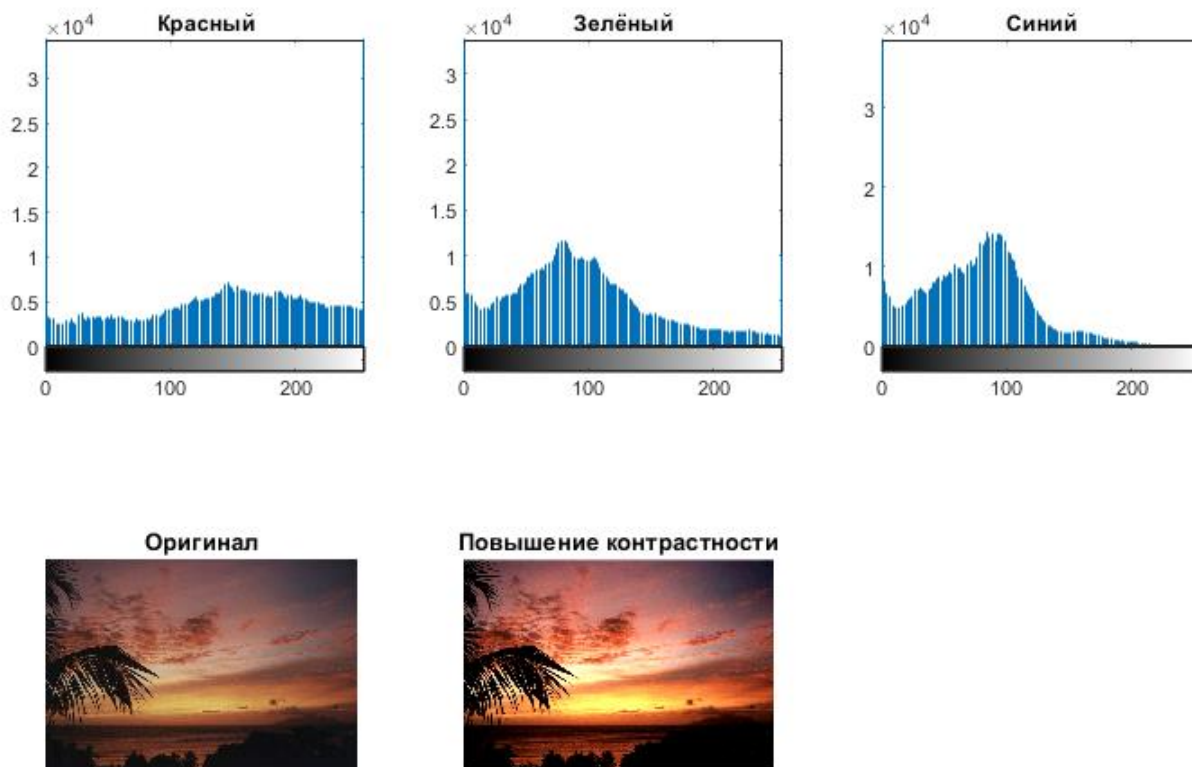


Рисунок 12. Повышение контрастности изображения с помощью *imadjust()*

**Листинг 10.** Эквиализация изображения с помощью *histeq()*

```
for k = 1:1:Layers;
    Inew_eq(:,:,k) = histeq(I(:,:,k));
End
```

```
figure;
subplot(2,3,1)
imhist(Inew_eq(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_eq(:, :, 2))
title('Зелёный')
subplot(2,3,3)
imhist(Inew_eq(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
```

```

title('Оригинал')
subplot(2,3,5)
imshow(Inew_eq)
title('Эквализация')

```

*Комментарий:* данная функция эквализирует (= выравнивает) гистограмму методом распределения значений интенсивностей элементов исходного изображения.

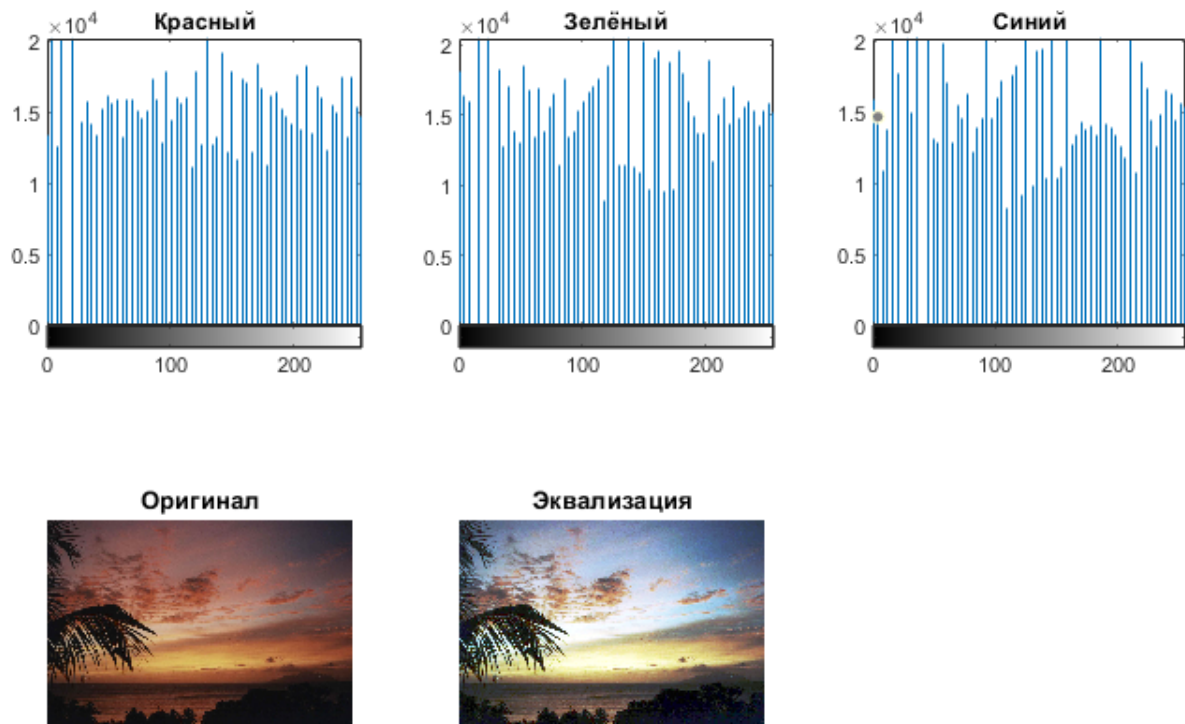


Рисунок 13. Применение функции histeq()

**Листинг 10.** Адаптивная эквализация изображения с помощью *adapthisteq()*

```

for k = 1:1:Layers;
    Inew_adeq(:, :, k) = adapthisteq(I(:, :, k));
end

```

```

figure;
subplot(2,3,1)
imhist(Inew_adeq(:, :, 1))
title('Красный')
subplot(2,3,2)
imhist(Inew_adeq(:, :, 2))
title('Зелёный')

```

```

subplot(2,3,3)
imhist(Inew_adeq(:, :, 3))
title('Синий')
subplot(2,3,4)
imshow(I)
title('Оригинал')
subplot(2,3,5)
imshow(Inew_adeq)
title('Адаптивная эквализация')

```

*Комментарий:* данная функция выполняет контрастно-ограниченное адаптивное выравнивание.

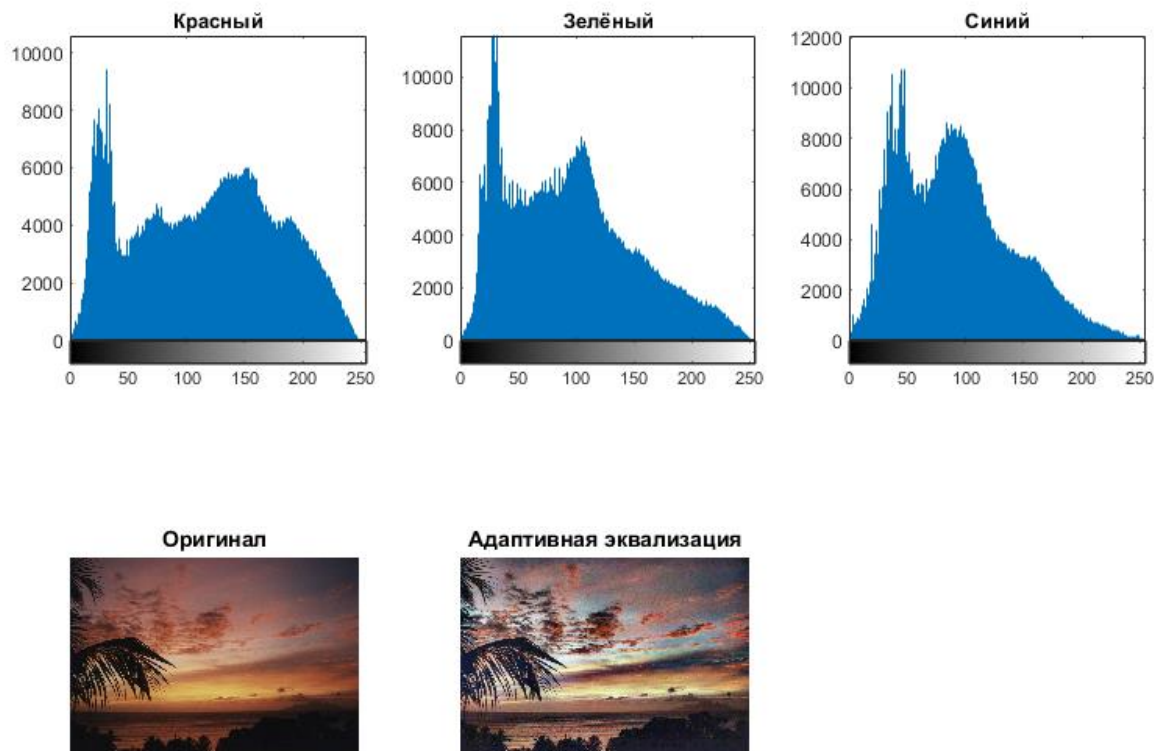


Рисунок 14. Применение адаптивной эквализации с помощью `adapthisteq()`

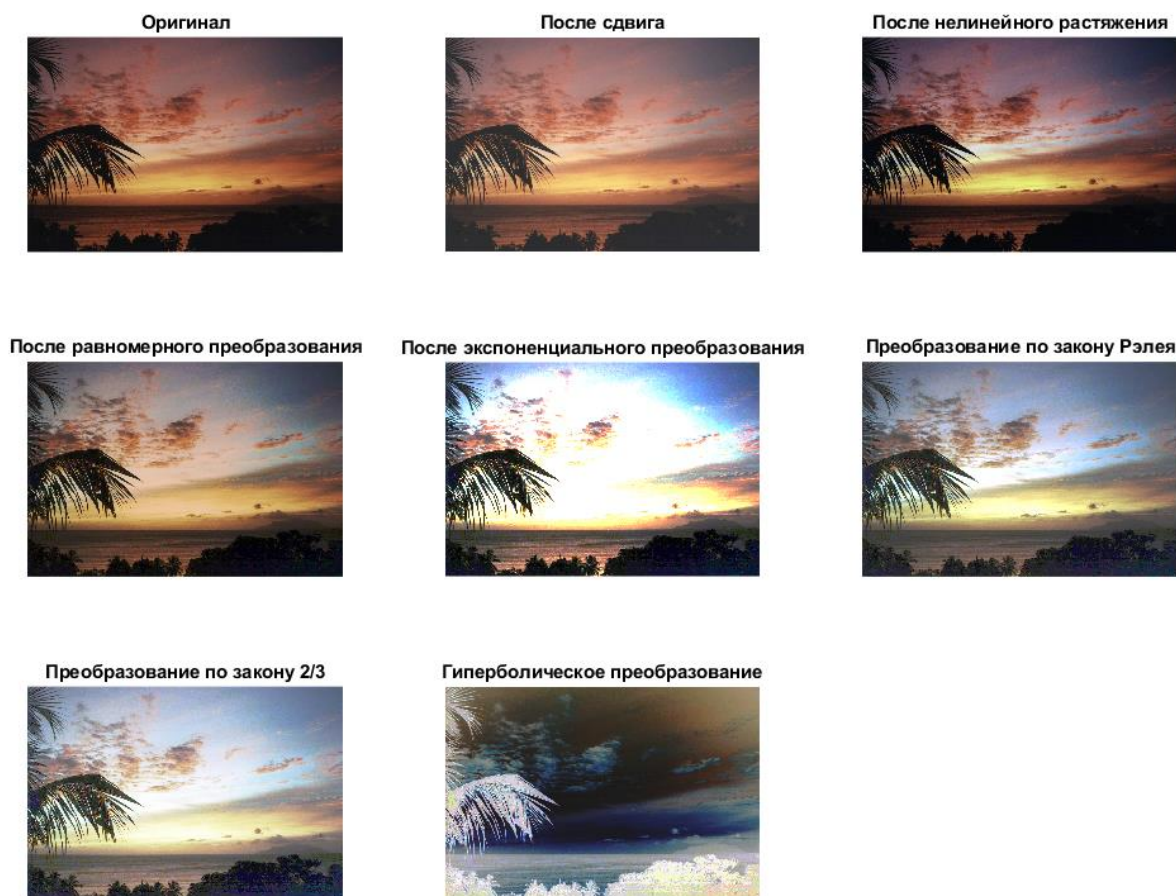


Рисунок 15. Сравнение результатов преобразований

*Комментарий:* будем сравнивать, двигаясь слева направо по ряду и сверху вниз по рядам.

Видно, что при применении сдвига тёмная область справа снизу стала посветлее. После применения нелинейного растяжения на небе добавился голубой оттенок и в целом цвета на картинке поменялись из-за расширения диапазона. При равномерном преобразовании картинка стала более спокойной из-за того, что небо стало голубым. При экспоненциальном преобразовании, преобразовании по закону Рэля, по закону степени  $2/3$  картинка становится светлее, но при этом резче. При гиперболическом преобразовании получаем негативный вариант оригинала.



Рисунок 16. Сравнение применения встроенных функций

*Комментарий:* в целом, функция повышения контрастности её и повышает – небо становится ярко-красным, даже алым. Функции эквализации добавляют на изображение чисто синий цвет.

***Листинг 11. Построение профиля изображения***

```
I = imread("code.jpg");
[numRows, numCols, Layers] = size(I);
x = [1 numCols];
y = [ceil(numRows/2) ceil(numRows/2)];
figure;
subplot(1,2,1)
imshow(I)
title('Штрих-код')
subplot(1,2,2)
improfile(I,x,y), grid minor;
title('Профиль штрих-кода вдоль оси Oх')
```



*Комментарий:* профиль изображения вдоль прямой вычисляется с помощью следующей функции:

$$Profile\ i(x) = I(x, i), \quad (8)$$

где  $i$  – номер строки изображения.

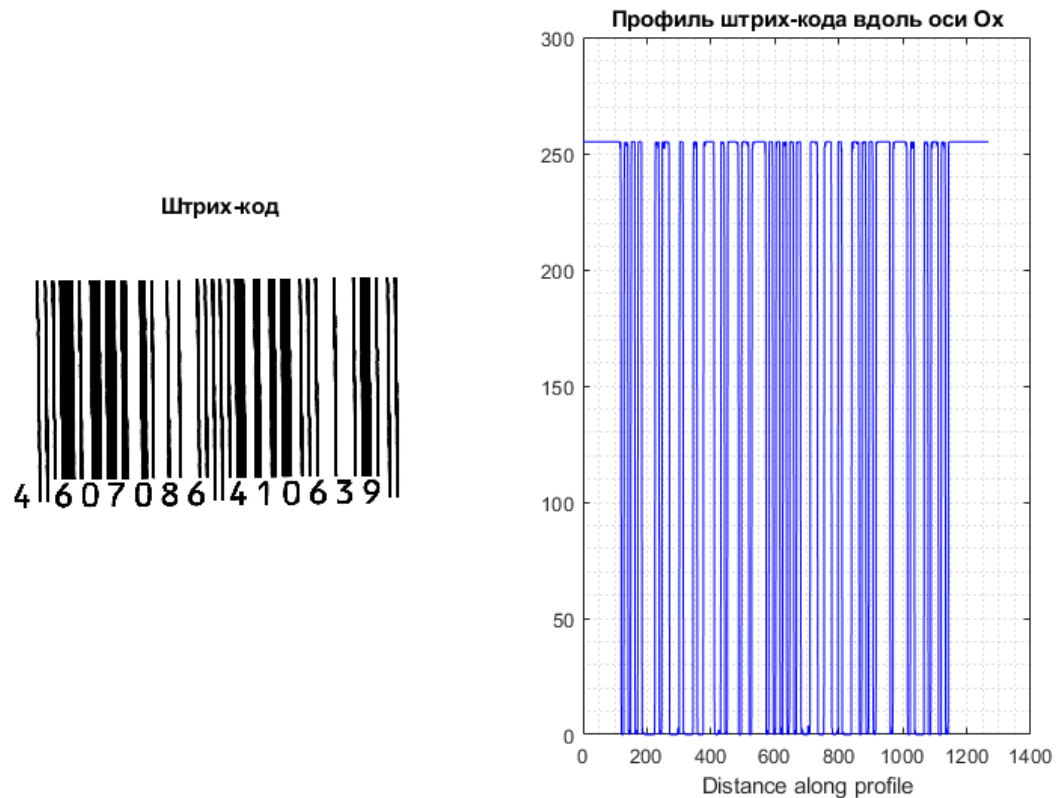


Рисунок 17. Построение профиля штрих-кода

### *Листинг 12. Построение проекции изображения*

```
I = imread("apple.jpg");
[numRows, numCols, Layers] = size(I);
for i=1:1:numRows
    Proj_Y(i,1)= (round(sum(I(i,:,1)))) + round(sum(I(i,:,2))) + round(sum(I(i,:,3))))
    / (256 * 3);
end
for i=1:1:numCols
    Proj_X(1,i)= (round(sum(I(:,i,1)))) + round(sum(I(:,i,2))) + round(sum(I(:,i,3))))
    / (256 * 3);
end
figure;
subplot(2,2,1)
```

```

imshow(I)
subplot(2,2,2)
plot(Proj_Y, 1:numRows)
grid minor
subplot(2,2,3)
plot(1:numCols, Proj_X)
grid minor

```

*Комментарий:* в простейшем случае, проекции двумерного изображения являются вертикальная проекция на ось Ох и горизонтальная проекция на ось Оу. Ниже представлены формулы для вычисления.

$$Proj\ X(y) = \sum_{y=0}^{\dim Y-1} I(x, y) \quad (9)$$

$$Proj\ Y(y) = \sum_{y=0}^{\dim X-1} I(x, y) \quad (10)$$

Для вертикальной проекции необходимо сложить интенсивность пикселей по столбцам, а для горизонтальной по строкам. Анализ с помощью проекций позволяет определять точки, которые соответствуют контурам изображений.

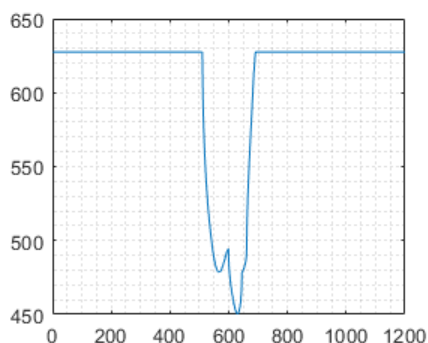
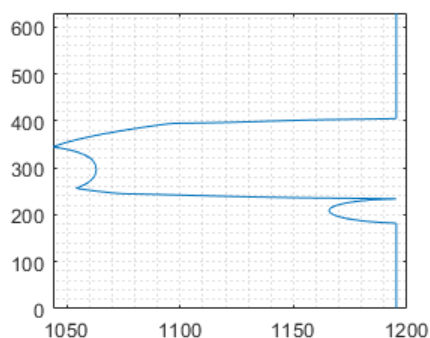


Рисунок 18. Пример проекции для логотипа Apple

**Выводы.** В ходе лабораторной работы были выполнены следующие шаги:

- Для изображения была получена его гистограмма по каждому из трёх цветов;
- Для гистограммы было применено смещение вправо, то есть осветление, было применено нелинейное растяжение динамического диапазона и равномерное преобразование;
- Были также применены следующие преобразования: экспоненциальное, по закону Рэлея, по закону степени  $2/3$  и гиперболическое. Первые три сделали фотографию светлее, а экспоненциальное даже засветило, а гиперболическое дало негативный вариант фотографии;
- Были рассмотрены встроенные в MATLAB функции по повышению контрастности и эквализации, то есть выравниванию, изображения;



- Для картинки типа «штрих-код» был получен профиль, позволяющий определить толщины и количество линий на таких картинках;

- Для картинки по типу «монотонный фон – выделяющийся объект» на примере логотипа Apple были получены две проекции на вертикальную и горизонтальную оси, позволяющие определить контуры этого объекта.

### **Ответы на вопросы к лабораторной работе:**

1. Что такое контрастность изображения и как можно её изменить?

Ответ:

Контрастность – это диапазон значений между максимальной и минимальной яркостями изображения.

Контрастность можно изменять с помощью арифметических операций, например, смещения гистограмм влево или вправо, можно с помощью различных преобразований, например, равномерного или экспоненциального, можно с помощью расширения динамического диапазона (если интенсивности лежат в узком диапазоне), можно с помощью выравнивания (эквалайзинга) гистограмм.

2. Чем эффективно использование профилей и проекций изображения?

Ответ:

Профили и проекции позволяют свести геометрические компоненты изображения ко всего лишь одномерному массиву данных, что позволяет увеличить скорость вычислений, упростить их, уменьшить затраты памяти.

3. Каким образом можно найти объект на равномерном фоне?

Ответ:

Это можно сделать с помощью проекций – суммируем интенсивности по столбцам для вертикальной проекции на ось  $Ox$  и интенсивности по строкам для горизонтальной проекции на ось  $Oy$ . Поскольку фон

равномерный, то места «скачков» интенсивностей будут свидетельствовать о том, что что-то изменилось вдоль оси, то есть появился контур объекта. Зная контур, можно вычислить координаты середины изображения.