Министерство образования Республики Беларусь

Учреждение Образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Кафедра электронных вычислительных средств

Лабораторная работа № 3

«Преобразование яркости изображений и пространственная фильтрация»

|  |  |
| --- | --- |
| Проверил: | Выполнил: |
| Рыбенков Е.В. | ст. гр. 850701 |
|  | Филипцов Д. А. |
|  | |

Минск 2021

# Цель работы

Изменение яркости, синтез и использование цифровых фильтров в задачах обработки изображений.

# **Ход работы**

Задание на эквализацию гистограммы:

a) Подсказка: функции unique, cumsum.

b) Загрузить все изображения (1\_1.tif – 1\_4.tif) с помощью команды imread.

c) Преобразовать изображение в тип double. Приводить изображения к диапазону [0;1] ненужно. Задавать диапазон входных яркостей изображения от 0 до 255.

d) Рассчитать гистограмму изображения в соответствии с уравнением (1.1).

e) Рассчитать значения sk по формуле (1.2) и сохранить их в массиве T.

f) Выполнить эквализацию гистограммы sk=T(rk).

g) Отобразить изображения и их гистограммы после эквализации с помощью команд imshow.

MATLAB-код:

% преобразованние исходных изображений к типу double [0 255]

I1 = double(imread('1\_1.tif'));

I2 = double(imread('1\_2.tif'));

I3 = double(imread('1\_3.tif'));

I4 = double(imread('1\_4.tif'));

scale = 256;

gist1 = gist(I1);

gist2 = gist(I2);

gist3 = gist(I3);

gist4 = gist(I4);

figure('Name','Before'); % вывод оригиналных изображений и их гистрограмм

subplot(4, 2, 1); imshow(I1, [0 (scale - 1)]);

subplot(4, 2, 2); plot(gist1, 'o');

subplot(4, 2, 3); imshow(I2, [0 (scale - 1)]);

subplot(4, 2, 4); plot(gist2, 'o');

subplot(4, 2, 5); imshow(I3, [0 (scale - 1)]);

subplot(4, 2, 6); plot(gist3, 'o');

subplot(4, 2, 7); imshow(I4, [0 (scale - 1)]);

subplot(4, 2, 8); plot(gist4, 'o');

% эквализация гистрограмм

[T1, Ieq1] = ekval(gist1, I1);

[T2, Ieq2] = ekval(gist2, I2);

[T3, Ieq3] = ekval(gist3, I3);

[T4, Ieq4] = ekval(gist4, I4);

figure('Name','After'); % вывод изображений после эквализации и их гистрограмм

subplot(4, 2, 1); imshow(Ieq1, [0 (scale - 1)]);

subplot(4, 2, 2); plot(gist(Ieq1), 'o');

subplot(4, 2, 3); imshow(Ieq2, [0 (scale - 1)]);

subplot(4, 2, 4); plot(gist(Ieq2), 'o');

subplot(4, 2, 5); imshow(Ieq3, [0 (scale - 1)]);

subplot(4, 2, 6); plot(gist(Ieq3), 'o');

subplot(4, 2, 7); imshow(Ieq4, [0 (scale - 1)]);

subplot(4, 2, 8); plot(gist(Ieq4), 'o');

Код функции gistag:

function [gistag] = gist(imag)

%GIST making histogram from image 2-dimention matrix

gistag = zeros(1, 256);

sizeImag = size(imag);

for a = 1:(sizeImag(1)\*sizeImag(2))

gistag(imag(a) + 1) = gistag(imag(a) + 1) + 1;

end

gistag = gistag ./ (sizeImag(1) \* sizeImag(2));

end

Код функции ekval:

function [T, Ieq] = ekval(gistag, I)

%EKVAL making histogram and image equalization

scale = size(gistag);

T = zeros(scale);

for k = 1:scale(2)

T(k) = round( (scale(2) - 1) \* sum(gistag(1:k)) );

end

sizeI = size(I);

Ieq = zeros(sizeI);

for a = 1:(sizeI(1)\*sizeI(2))

Ieq(a) = T(I(a)+1);

end

end

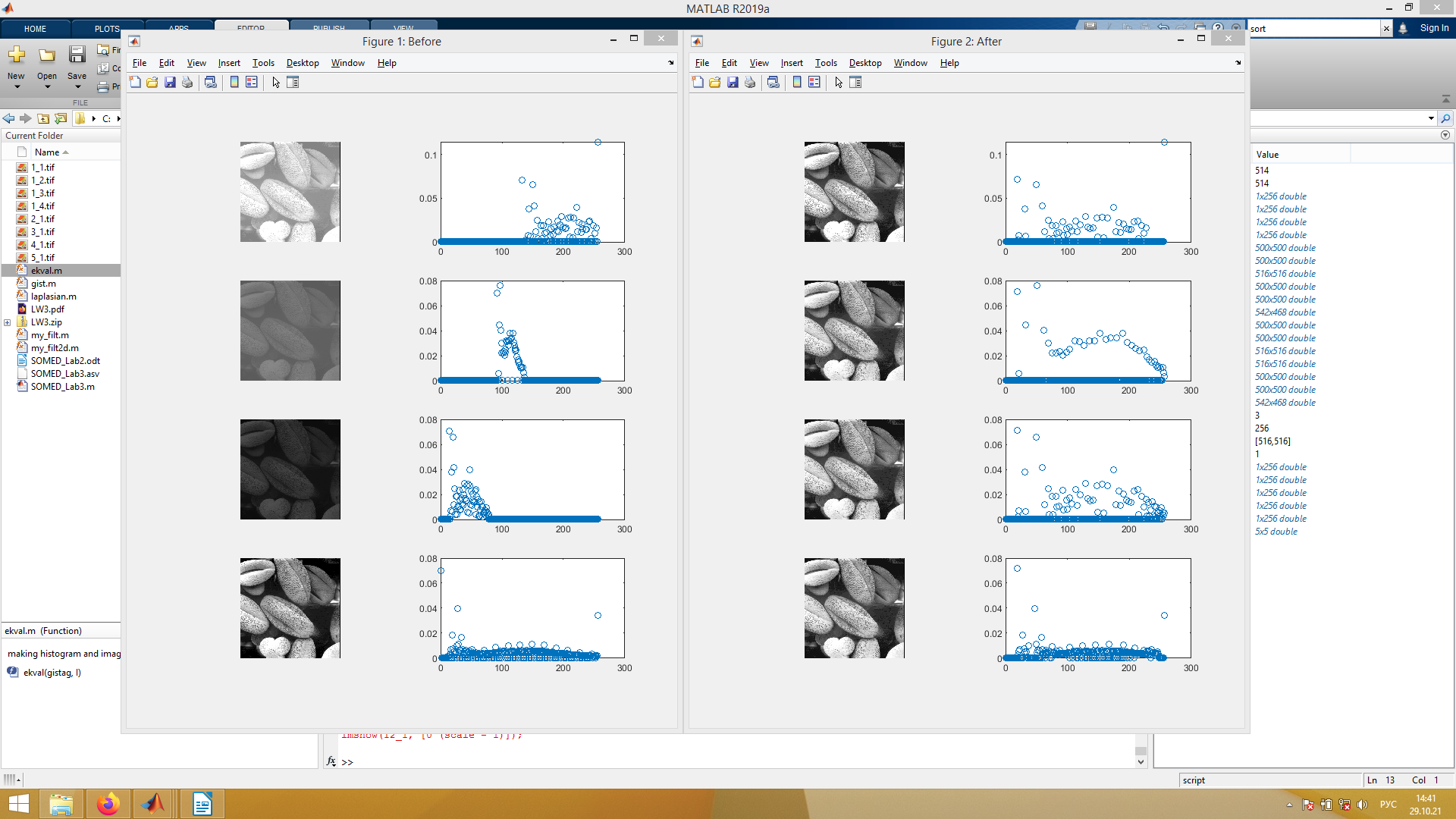


Рисунок 1 – Исходные изображения с их гистрограммами (слева) и изображения после эквализации (справа)

Задание на локальную эквализацию гистограммы:

a) Загрузить изображение (2\_1.tif) с помощью команды imread.

b) Преобразовать изображение в тип double. Приводить изображения к диапазону [0;1] ненужно. Задавать диапазон входных яркостей изображения от 0 до 255.

c) Дополнить исходное изображение нулями для правильной работы на границе изображения (wextend).

d) Выполнить локальную гистограммную обработку для области размером 3x3 и 5x5(см. теор. Часть).

e) Отобразить изображения с помощью команд imshow.

Код программы:

I2\_1 = wextend('2d', 'zpd', double(imread('2\_1.tif')), 2); sizeI2\_1 = size(I2\_1);

figure('Name','Before');

imshow(I2\_1, [0 (scale - 1)]);

% локальная эквализация окрестностью 3x3

Ieq2\_1\_1 = zeros(sizeI2\_1);

for a = 3:(sizeI2\_1(1)-2)

for b = 3:(sizeI2\_1(2)-2)

tempI = I2\_1(a-1:a+1, b-1:b+1);

[~, Ieq2\_1\_1(a-1:a+1, b-1:b+1)] = ekval(gist(tempI), tempI);

end

end

% локальная эквализация окрестностью 5x5

Ieq2\_1\_2 = zeros(sizeI2\_1);

for a = 3:(sizeI2\_1(1)-2)

for b = 3:(sizeI2\_1(2)-2)

tempI = I2\_1(a-2:a+2, b-2:b+2);

[T, Ieq2\_1\_2(a-2:a+2, b-2:b+2)] = ekval(gist(tempI), tempI);

end

end

figure('Name','After');

subplot(1, 2, 1); imshow(Ieq2\_1\_1, [0 (scale - 1)]);

subplot(1, 2, 2); imshow(Ieq2\_1\_2, [0 (scale - 1)]);

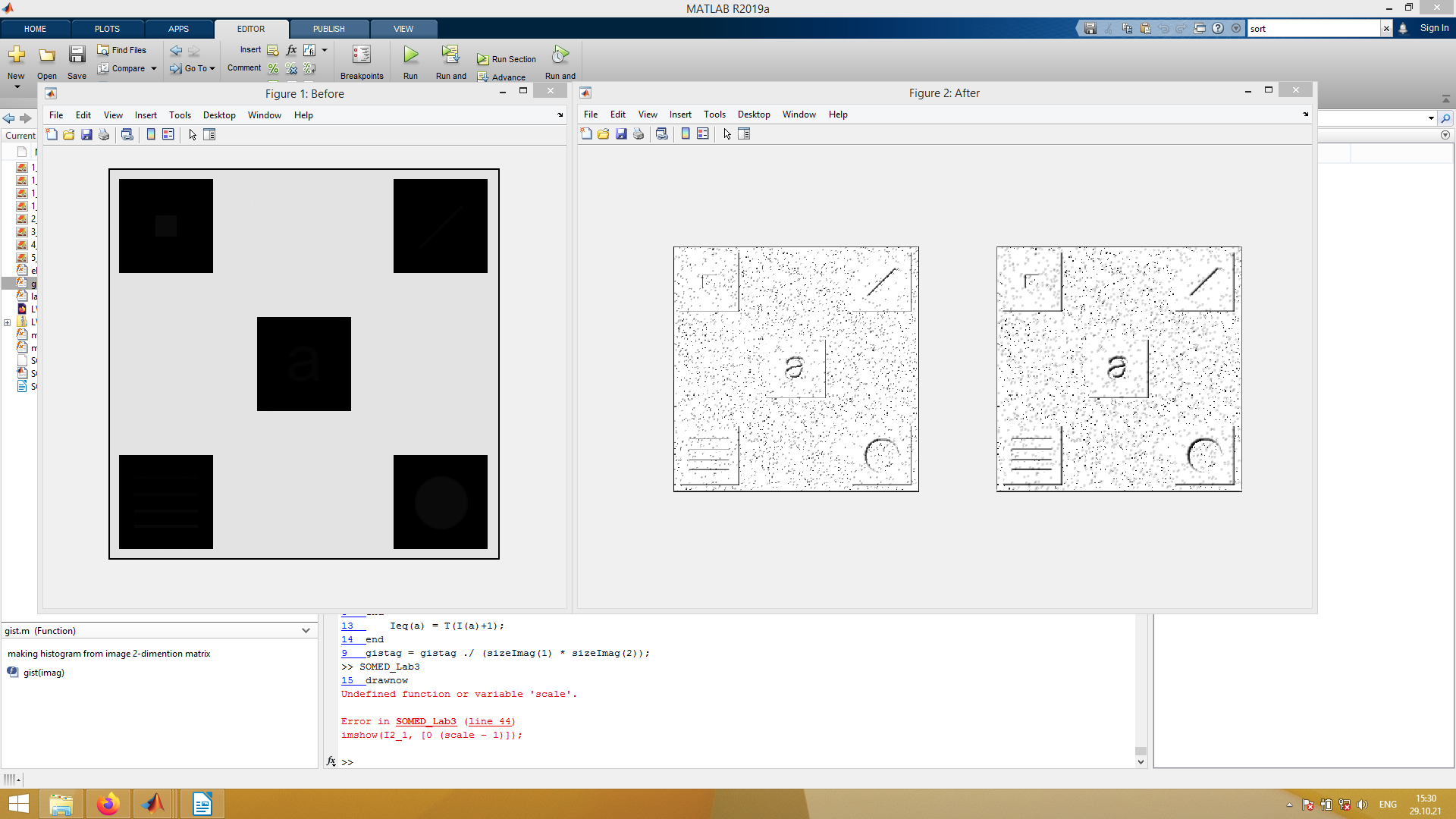


Рисунок 2 – Исходное изображение (слева) и изображения, обработанные по алгоритму локальной эквализации области 3x3 (по центру) и 5x5 (справа)

Задание на пространственную фильтрацию:

a) Загрузить изображение (3\_1.tif) с помощью команды imread.

b) Преобразовать изображение в тип double. Приводить изображения к диапазону [0;1] ненужно. Задавать диапазон входных яркостей изображения от 0 до 255.

c) Написать функцию рассчета 2D свертки my\_filt2d в соответствии с уравнением (1.3).

d) Рассчитать ядро сглаживающего фильтра размером 15x15 по формуле (1.4).

e) Выполнить сглаживание тестового изображения.

f) Отобразить изображение с помощью команд imshow.

g) Выполнить пороговую обработку сглаженного изображения путем присвоения значения 255 всем пикселям большим 64, иначе присвоить значение 0.

h) Отобразить изображение с помощью команд imshow.

Код программы:

m = 15;

I3\_1 = wextend('2d', 'zpd', double(imread('3\_1.tif')), (m-1)/2);

figure('Name','Before');

imshow(I3\_1, [0 (scale - 1)]);

Ifiltr3\_1 = my\_filt2d(I3\_1, my\_filt(m, m)); % фильтрация сглаживающим фильтром 15 x 15

figure('Name','After');

subplot(1, 2, 1); imshow(Ifiltr3\_1, [0 (scale - 1)]);

for x = 1:size(Ifiltr3\_1, 1) % пороговая фильтрация

for y = 1:size(Ifiltr3\_1, 2)

if Ifiltr3\_1(x, y) > 64

Ifiltr3\_1(x, y) = 255;

else

Ifiltr3\_1(x, y) = 0;

end

end

end

subplot(1, 2, 2); imshow(Ifiltr3\_1, [0 (scale - 1)]);

Код функции my\_filt:

function [w] = my\_filt(m, n)

%MY\_FILT сглаживающий фильтр m x n

a = (m-1)/2;

b = (n-1)/2;

w = zeros(m, n);

for s = -a:a

for t = -b:b

w(s+a+1, t+b+1) = 1/(m\*n);

end

end

end

Код функции my\_filt2d:

function [Ifltr] = my\_filt2d(I, w)

%MY\_FILT2D filtering

sizeI = size(I);

Ifltr = zeros(sizeI);

[m, n] = size(w);

a = (m-1)/2;

b = (n-1)/2;

for x = (1 + a):(sizeI(1) - a)

for y = (1 + b):(sizeI(2) - b)

sum = 0;

for s = -a:a

for t = -b:b

sum = sum + (w(s+a+1, t+b+1) \* I(x - s, y - t));

end

end

Ifltr(x, y) = round(sum);

end

end

end

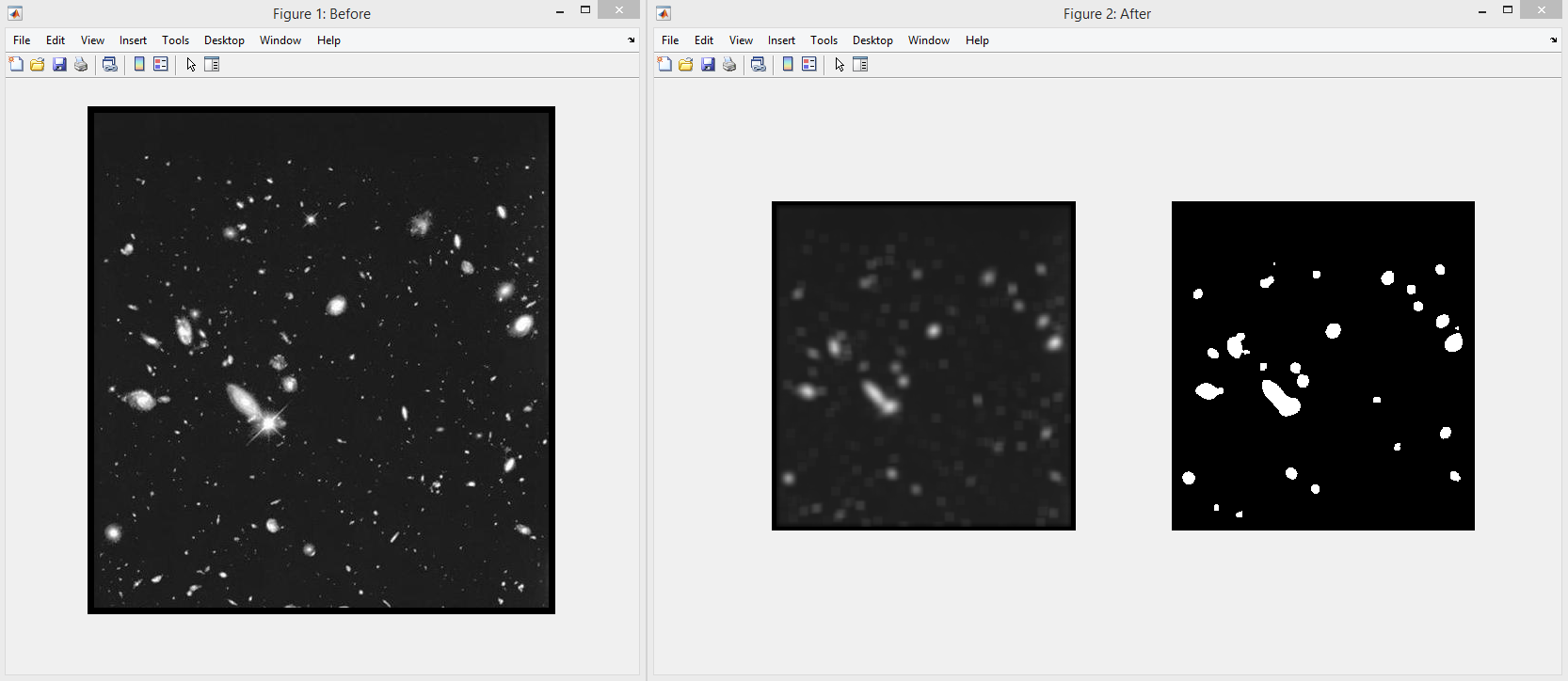


Рисунок 3 – Исходное изображение (слева) и изображения, обработанные по алгоритму сглаживания (по центру) и пороговой фильтрации впоследствии (справа)

Задание на медианную фильтрацию:

a) Загрузить изображение (4\_1.tif) с помощью команды imread.

b) Преобразовать изображение в тип double. Приводить изображения к диапазону [0;1] ненужно. Задавать диапазон входных яркостей изображения от 0 до 255.

c) Рассчитать ядро сглаживающего фильтра размером 3x3 по формуле (1.4).

d) Выполнить сглаживание тестового изображения.

e) Выполнить медианную фильтрацию тестового изображения.

f) Отобразить изображения с помощью команд imshow.

Код программы:

m = 3;

step = (m-1)/2;

I4\_1 = wextend('2d', 'zpd', double(imread('4\_1.tif')), step);

figure('Name','Before');

imshow(I4\_1, [0 (scale - 1)]);

Ifiltr4\_1 = my\_filt2d(I4\_1, my\_filt(m, m)); % фильтрация сглаживающим фильтром 3x3

figure('Name','After');

subplot(1, 2, 1); imshow(Ifiltr4\_1, [0 (scale - 1)]);

Temp = zeros(m, m);

Temp\_s = zeros(m^2, 1);

for x = (1 + step):(size(Ifiltr4\_1, 1) - step) % медианная фильтрация

for y = (1 + step):(size(Ifiltr4\_1, 2) - step)

Temp = Ifiltr4\_1((x - step):(x + step), (y - step):(y + step));

Temp\_s = sort(Temp(:));

Ifiltr4\_1(x, y) = Temp\_s((m^2+1)/2);

end

end

subplot(1, 2, 2); imshow(Ifiltr4\_1, [0 (scale - 1)]);

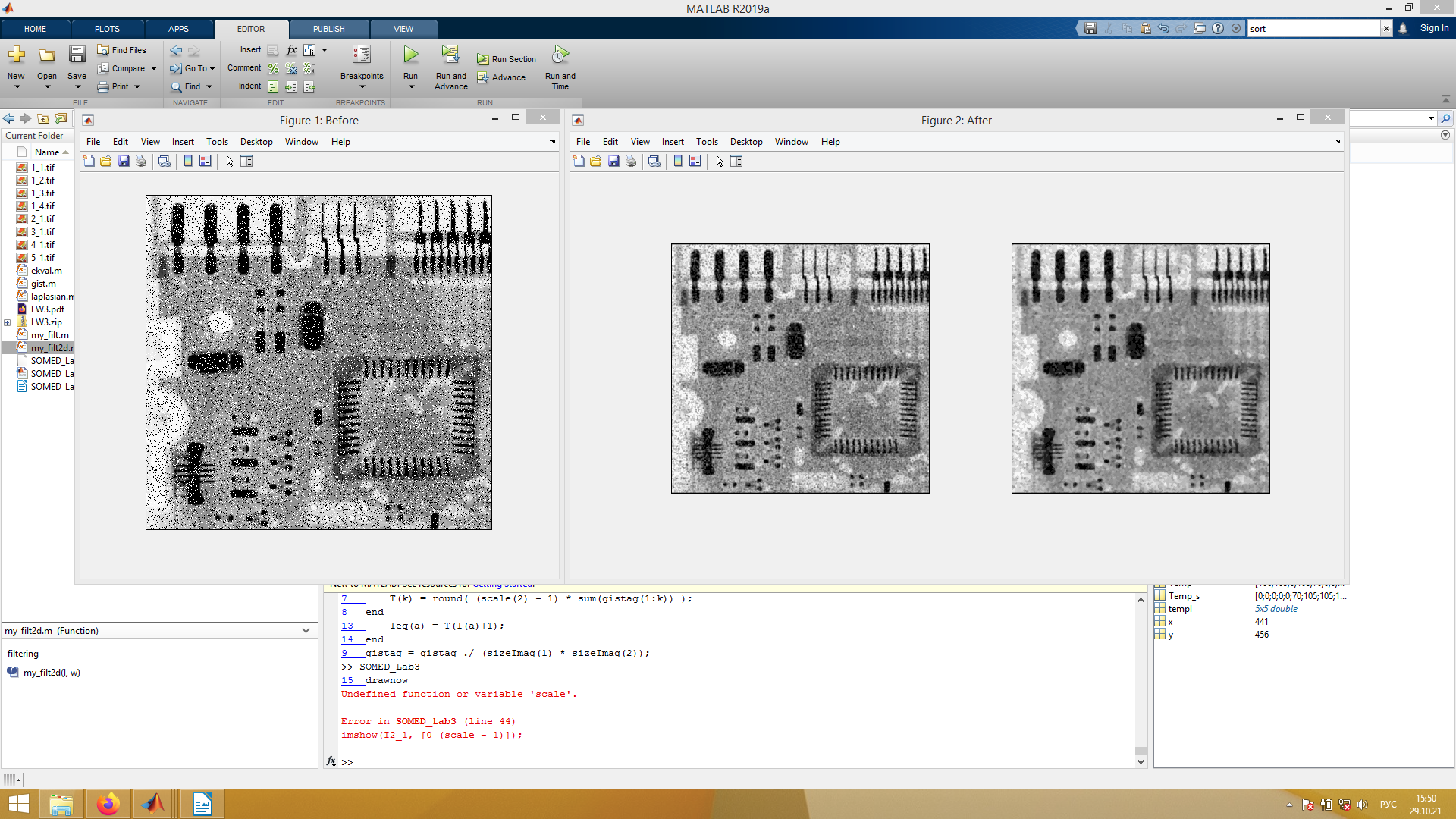


Рисунок 4 – Исходное изображение (слева) и изображения, обработанные по алгоритму сглаживания (по центру) и медианной фильтрации впоследствии (справа)

Задание на повышение резкости изображения:

a) Загрузить изображение (5\_1.tif) с помощью команды imread.

b) Преобразовать изображение в тип double. Приводить изображения к диапазону [0;1] ненужно. Задавать диапазон входных яркостей изображения от 0 до 255.

c) Рассчитать ядро лапласиана размером 3x3 по формуле (1.5).

d) Выполнить фильтрацию тестового изображения тестового изображения с помощью ядра фильтра.

e) Отобразить изображение с помощью команд imshow.

f) Добавить фон на изображение по формуле (1.6).

g) Отобразить изображение с помощью команд imshow.

Код программы:

m = 3;

step = (m-1)/2;

I5\_1 = wextend('2d', 'zpd', double(imread('5\_1.tif')), step);

figure('Name','Before');

imshow(I5\_1, [0 (scale - 1)]);

Ifiltr5\_1 = laplasian(I5\_1); % рассчёт лапласиана 3 x 3 для оригинального изображения

figure('Name','After');

subplot(1, 2, 1); imshow(Ifiltr5\_1, [0 (scale - 1)]);

Ifiltr5\_1 = I5\_1 - Ifiltr5\_1; % добавление фона на изображение

subplot(1, 2, 2); imshow(Ifiltr5\_1, [0 (scale - 1)]);

Код функции laplasian:

function [Iret] = laplasian(I)

%LAPLASIAN Compiting laplassian 3 x 3

m = 3;

step = (m-1)/2;

Iret = zeros(size(I));

for x = (1 + step):(size(I, 1) - step)

for y = (1 + step):(size(I, 2) - step)

Iret(x, y) = I(x+1, y) + I(x-1, y) + I(x, y+1) + I(x, y-1) - 4\*I(x, y);

end

end

end

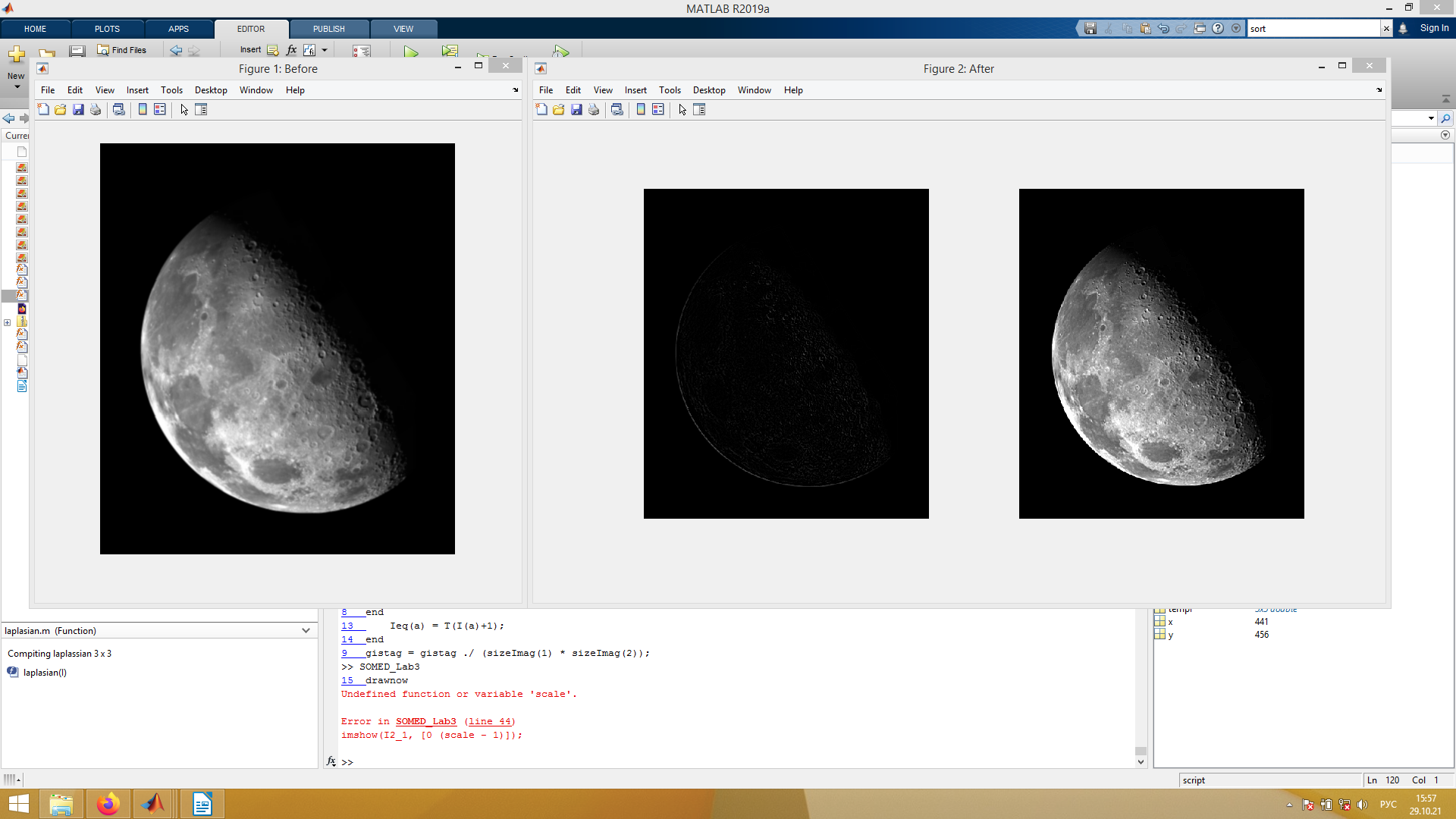


Рисунок 4 – Исходное изображение (слева) и изображения, обработанные по алгоритму лапласиана размером 3x3 (по центру) и после добавления фона на изображение (справа)

# Вывод

В ходе лабораторной работы были произведены расчёт гистограммы изображений, их глобальная и локальная эквализации, пространственная пороговая и медианная фильтрации изображений, увеличение резкости изображения при помощи оператора Лапласа, были подробно изучены данные алгоритмы.