

**Министерство науки и высшего образования  
Российской Федерации**

**Федеральное государственное автономное  
образовательное учреждение высшего образования**

**«Национальный исследовательский университет  
ИТМО»**

**Факультет информационных технологий и  
программирования**

Домашнее задание № 1

Реализация алгоритма для коррекции дисторсии и проектных искажений

**Выполнил студент группы № М3302**

Суворин Ярослав Владимирович

Санкт-Петербург  
2025

## 1. Ход выполнения работы

### 1.1 Подушкообразная дисторсия

1.1.a Условия этапа — преобразовать сетку, искаженную подушкообразной дисторсией, чтобы она стала максимально приближенной к идеальной сетке.

#### 1.1.b

```
img=imread("Рис_Д3 1.2.png");
[height, width, ~] = size(img);
center_x = width / 2;
center_y = height / 2;
k1 = 0.2;
k2 = 0.5;
[X, Y] = meshgrid(1:width, 1:height);

X_norm = (X - center_x) / center_x;
Y_norm = (Y - center_y) / center_y;
R = sqrt(X_norm.^2 + Y_norm.^2);
distortion_factor = 1 + k1 * R.^2 + k2 * R.^4;
X_corrected = X_norm .* distortion_factor;
Y_corrected = Y_norm .* distortion_factor;
X_corrected = X_corrected * center_x + center_x;
Y_corrected = Y_corrected * center_y + center_y;

corrected_img = zeros(size(img), 'like', img);

for channel = 1:size(img, 3)
    corrected_img(:, :, channel) = interp2(X, Y, double(img(:, :, channel)), X_corrected, Y_corrected, 'linear', 0);
end

corrected_img = uint8(corrected_img);

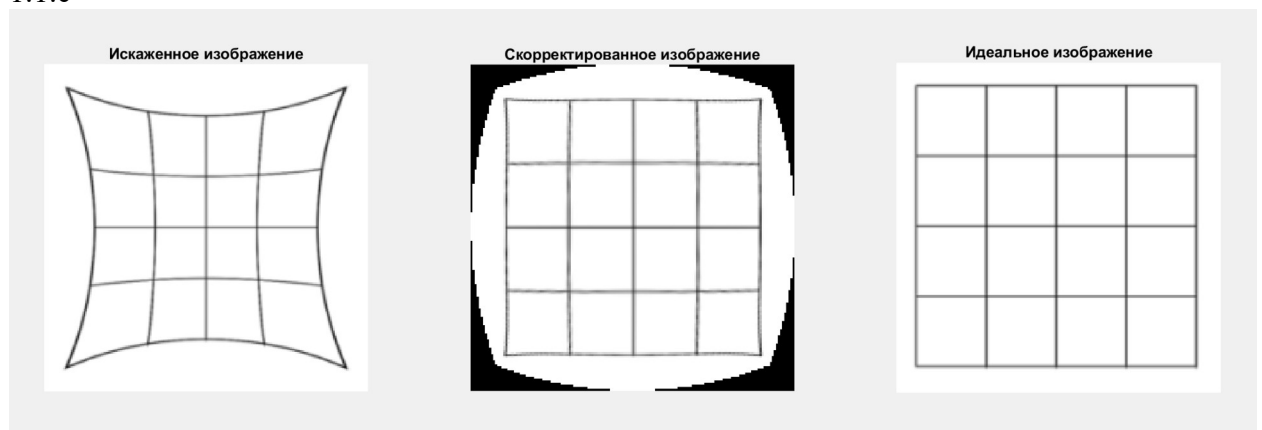
subplot(1, 3, 1); imshow(img); title('Искаженное изображение');

mask = any(corrected_img > 0, 3);
[rows, cols] = find(mask);
crop_rect = [min(cols), min(rows), max(cols)-min(cols), max(rows)-min(rows)];
corrected_img = imcrop(corrected_img, crop_rect);

subplot(1, 3, 2); imshow(corrected_img); title('Скорректированное изображение');
subplot(1, 3, 3); imshow(imread('Рис_Д3 1.1.png')); title('Идеальное изображение');
```

Подобрал коэффициенты  $k_1=0.2$  и  $k_2=0.5$

#### 1.1.c



1.2.a Условия этапа — преобразовать сетку, искаженную бочкообразной дисторсией, чтобы она стала максимально приближенной к идеальной сетке.

1.2.b Коэффициенты:  $k_1 = -0.55$ ,  $k_2 = -0.5$ ,  $k_3 = 1.2$ ;

```
img=imread("Рис_Д3 1.3.png");
[height, width, channels] = size(img);

border_size = 100;
img_padded = uint8(zeros(height + 2*border_size, width + 2*border_size, channels));
img_padded(border_size+1:border_size+height, border_size+1:border_size+width, :) = img;

[height, width, ~] = size(img_padded);

center_x = width / 2;
center_y = height / 2;
k1 = -0.55;
k2 = -0.5;
k3 = 1.2;
[X, Y] = meshgrid(1:width, 1:height);

X_norm = (X - center_x) / center_x;
Y_norm = (Y - center_y) / center_y;
R = sqrt(X_norm.^2 + Y_norm.^2);
distortion_factor = 1 + k1 * R.^2 + k2 * R.^4 + k3 * R.^6;
X_corrected = X_norm .* distortion_factor;
Y_corrected = Y_norm .* distortion_factor;
X_corrected = X_corrected * center_x + center_x;
Y_corrected = Y_corrected * center_y + center_y;

corrected_img = zeros(height, width, 3, 'uint8');
for channel = 1:3
    corrected_img(:, :, channel) = uint8(interp2(...
        double(img_padded(:, :, channel)), ...
        X_corrected, Y_corrected, 'linear', 0));
end

corrected_img = uint8(corrected_img);

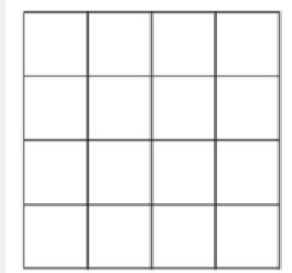
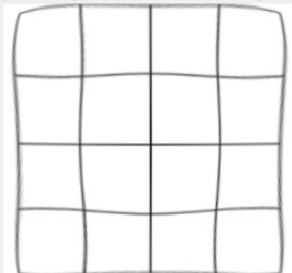
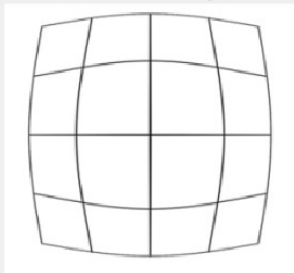
subplot(1, 3, 1); imshow(img); title('Искаженное изображение');

mask = any(corrected_img > 0, 3);
[rows, cols] = find(mask);
crop_rect = [min(cols), min(rows), max(cols)-min(cols), max(rows)-min(rows)];
corrected_img = imcrop(corrected_img, crop_rect);

subplot(1, 3, 2); imshow(corrected_img); title('Скорректированное изображение');
subplot(1, 3, 3); imshow(imread('Рис_Д3 1.1.png')); title('Идеальное изображение');
```

1.2.c

Искаженное изображение    Скорректированное изображение    Идеальное изображение



1.3.a Условия этапа — преобразовать сетку, искаженную перспективным искажением, чтобы она стала максимально приближенной к идеальной сетке.

1.3.b

```
img = imread('рис.png');
figure; imshow(img); title('Исходное изображение');
hold on;
[x, y] = ginput(4);
plot(x, y, 'ro', 'MarkerSize', 10);

width = 500;
height = 500;
target_points = [1, 1; width, 1; width, height; 1, height];
source_points = [x, y];
H = fitgeotrans(source_points, target_points, 'projective');

output_view = imref2d([height, width]);
corrected_img = imwarp(img, H, 'OutputView', output_view);
[height, width, ~] = size(corrected_img);

[Y, X] = meshgrid(1:width, 1:height);
scale_factor_Y = 1 + 0.4*(Y/height);
scale_factor_X = 1 + 0.5*(X/width);

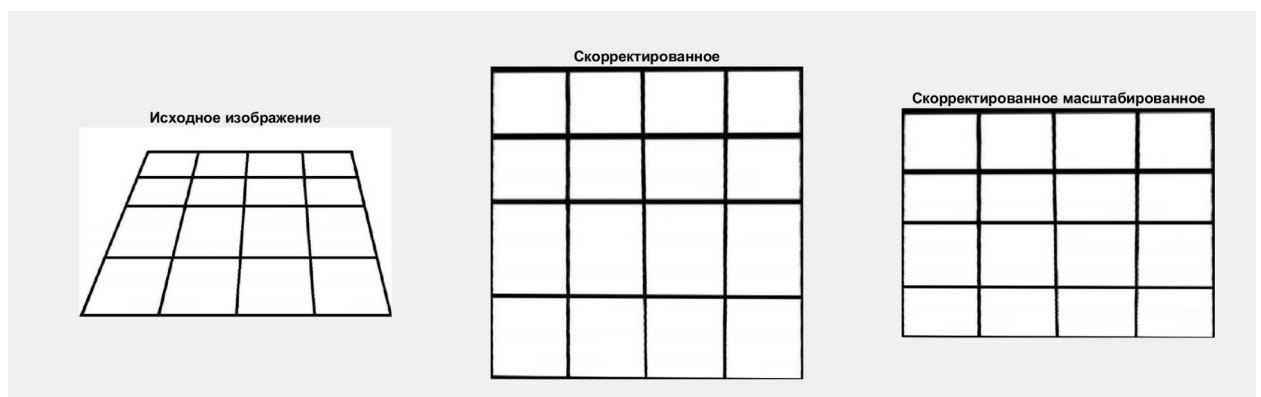
scaled_img = corrected_img;
for channel = 1:3
    scaled_img(:, :, channel) = interp2(double(corrected_img(:, :, channel)), Y.*scale_factor_Y, X.*scale_factor_X, 'linear', 0);
end

mask = any(scaled_img > 0, 3);
[rows, cols] = find(mask);
crop_rect = [min(rows), min(cols), max(rows)-min(rows), max(cols)-min(cols)];
scaled_img = imcrop(scaled_img, crop_rect);

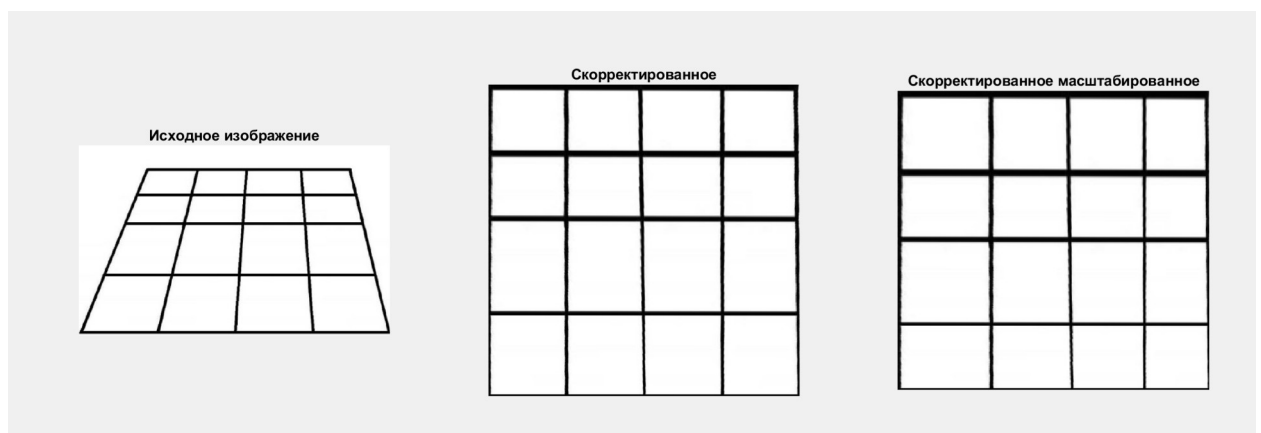
subplot(1,3,1); imshow(img); title('Исходное изображение');
subplot(1,3,2); imshow(corrected_img); title('Скорректированное');
subplot(1,3,3); imshow(scaled_img); title('Скорректированное масштабированное');
```

1.3.c

X: 0.5, Y: 0



X: 0.5, Y: 0.4



## 2. Вывод по работе

В результате проделанной работы попытался применить на практике алгоритмы для исправления изображений. Результаты получились противоречивые — с одной стороны, подушкообразную дисторсию получилось исправить практически идеально. С другой стороны, у бочкообразной дисторсии так и не получилось добиться подходящего результата — в процессе экспериментов, выходило либо выравнивание краёв, но при этом середина оставалась неисправленной, либо же середина исправлялась, но края очень сильно искажались подушкообразно. В результате я вывел вариант, который, с одной стороны, усреднил полученные результаты, но, с другой стороны, бросаются в глаза неровности и кривые линии. Основные проблемы этого этапа: верхняя линия подвергается искажению слабее, чем нижняя и боковые, а также неравномерно искажались края и центр.

Последний этап также нельзя назвать идеальным — несмотря на то, что в общем и целом сетку получилось привести в что-то похожее на оригинальное изображение, все еще видны неровности — неравные размеры ячеек, как относительно их сторон, так и относительно друг друга. Основная проблема этого этапа: специфический наклон сетки.

Общий вывод можно сделать такой — несмотря на теоретическую эффективность алгоритмов, на реальных случаях могут возникать большие проблемы, во многом из-за того, что сами искажения на изображениях неравномерные и даже перебор коэффициентов не всегда приведет к успеху.