Fatemeh Darbehani [redacted]

ENSC474 – SFU – Spring 2017

Assignment 3

Task 1

- The following 10 transformation were created and displayed using grids in this task:

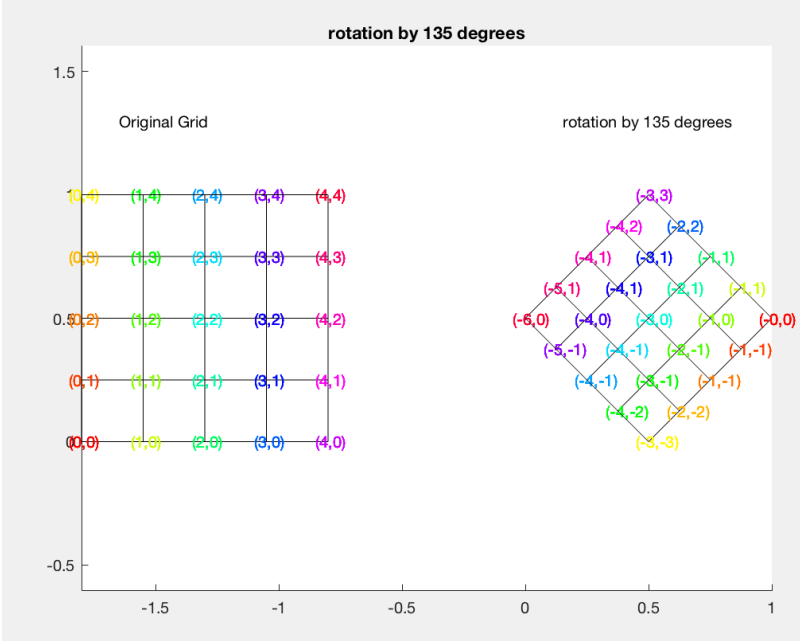

Figure 1: Pure Rotation

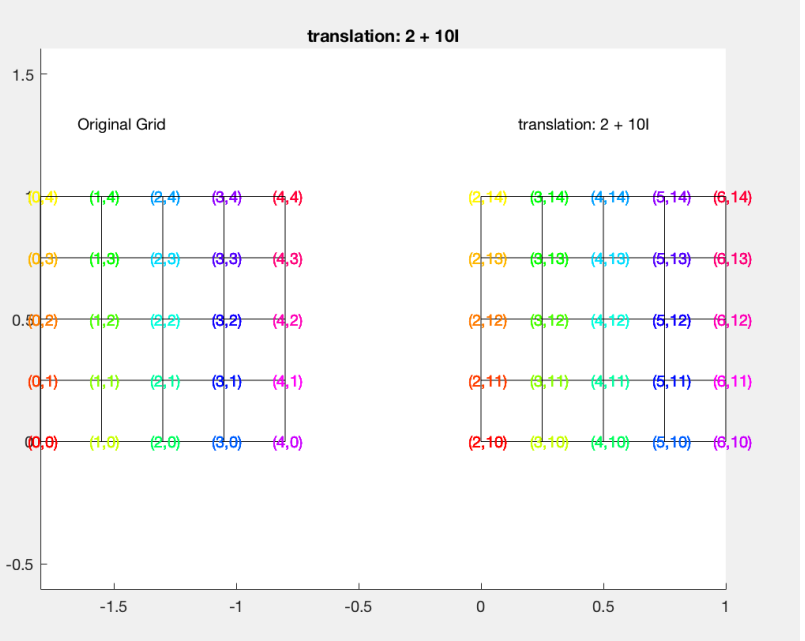

Figure 2: Pure Translation

Task1

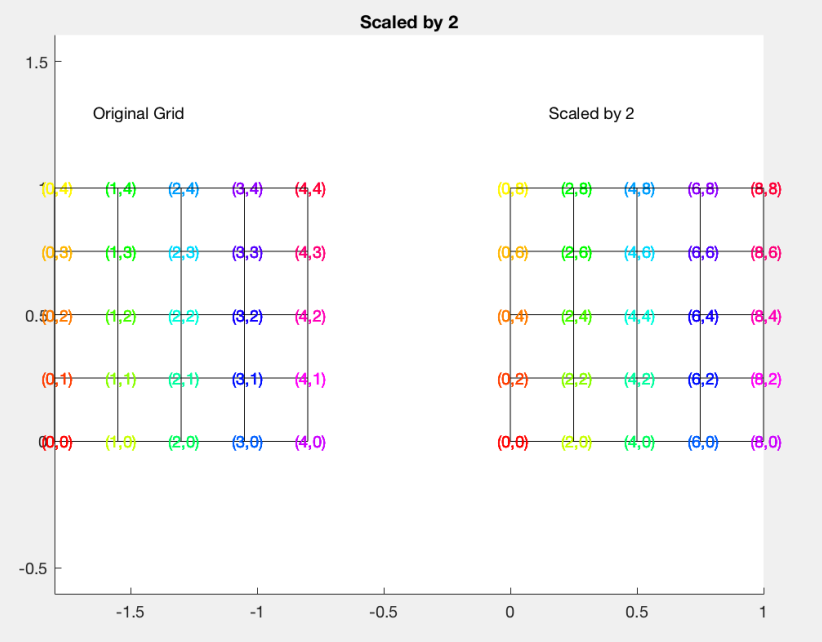

Figure 3: Pure Scale

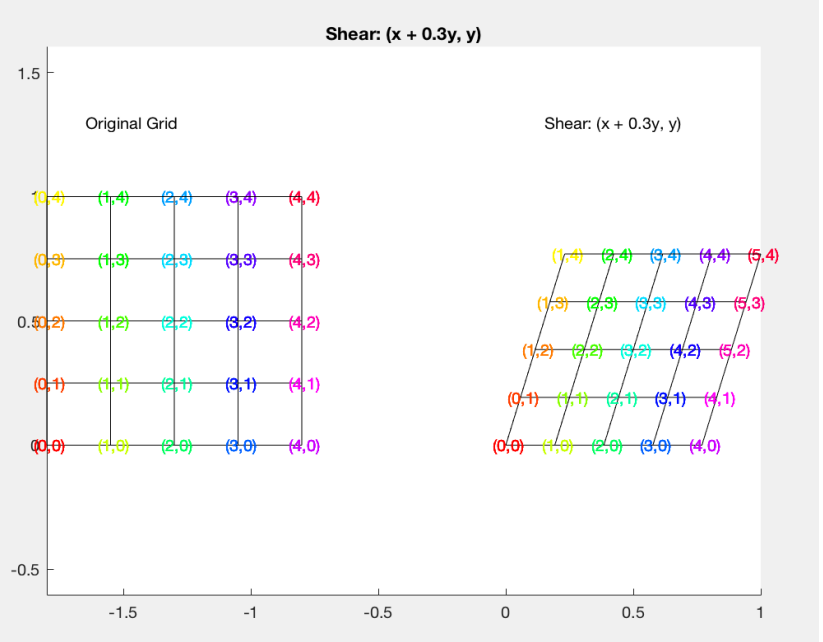

Figure 4: Shear

Task1

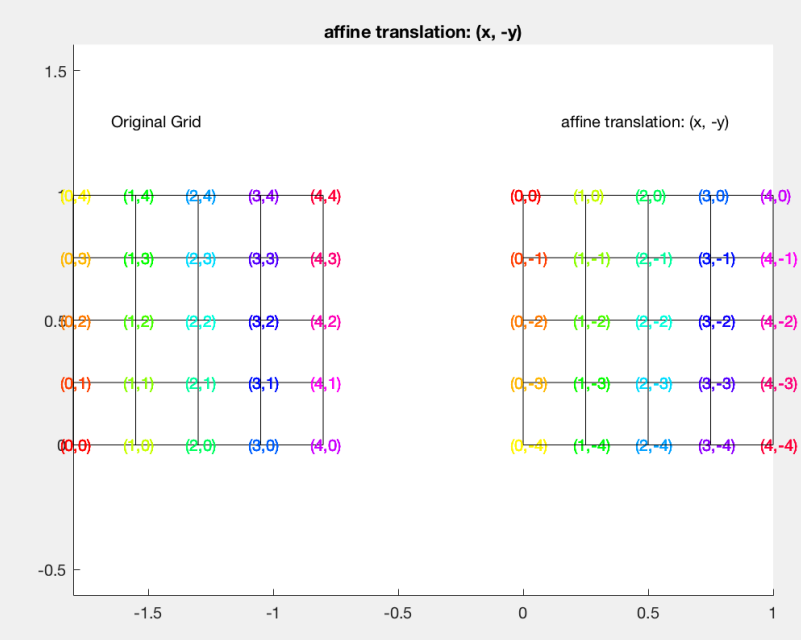

Figure 5: Affine Transformation

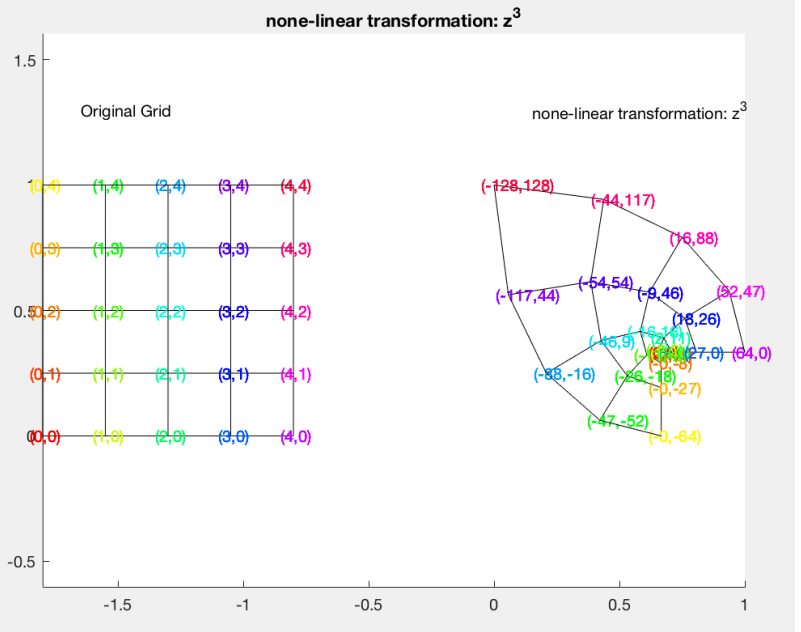

Figure 6: Non-linear Transformation

Task1

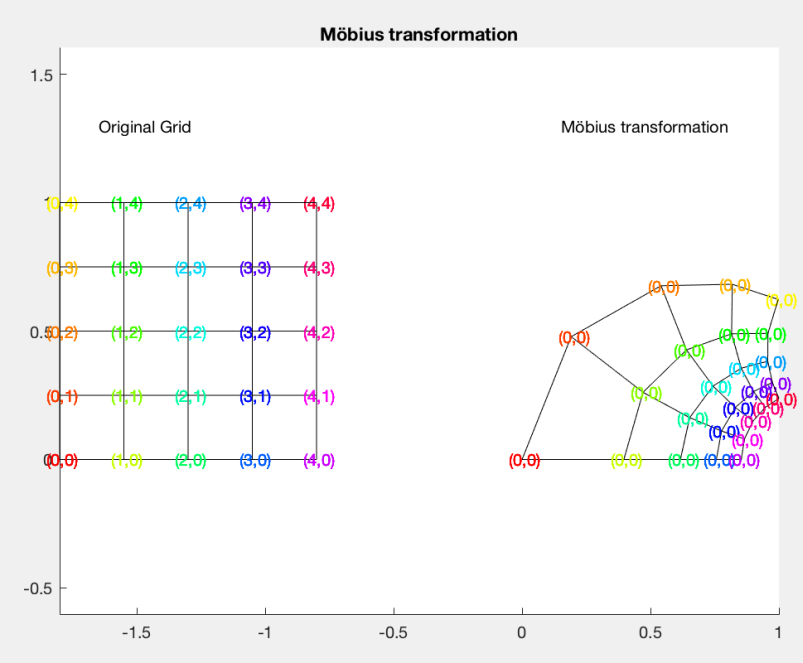

Figure 7: Mobius Transformation

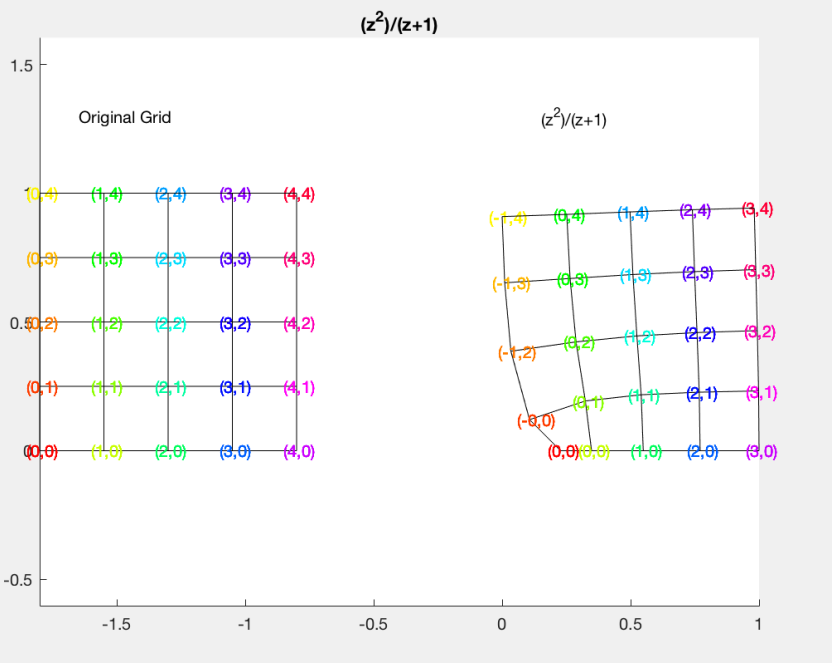

Figure 8: Non-linear Transformation

Task1

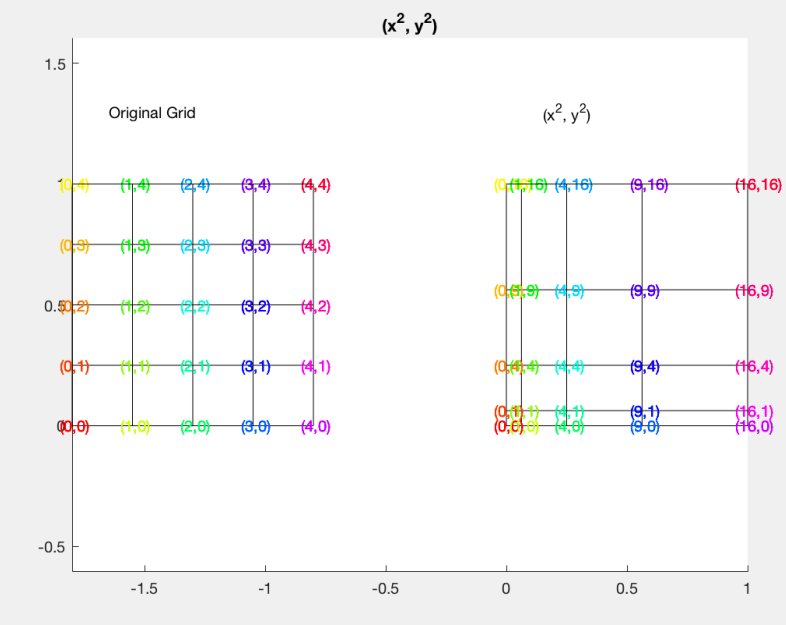

Figure 9: Non-linear Transformation

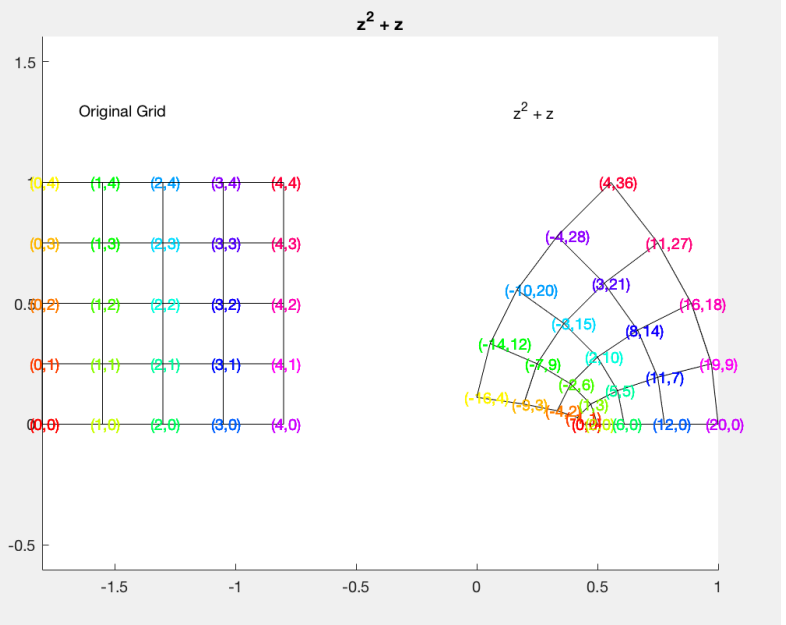

Figure 10: Non-linear Transformation

Task2

- The transform function was calculated using the following formula and displayed using grids:

$$\varphi(x) = R\,x + T$$

Figure 11: Matlab implementation

```
%% Finding Phi
alpha = 45;
cost = cos(alpha*pi/180); sint = sin(alpha*pi/180);
R = cost + sint*I;
T = 3 + 5*I;
phi = R .* Z + T;
phi_x = real(phi); phi_y = imag(phi);
```

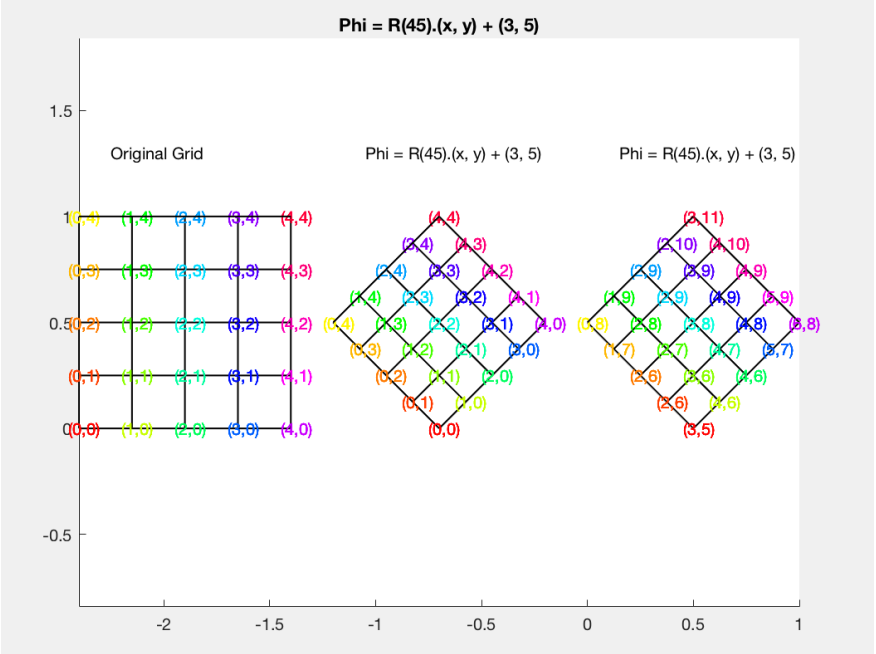

Figure 12: Phi Transformation

Task2

- The inverse transform function was calculated using the following formula and displayed using grids:

$$\varphi^{-1} = R^{-1}(x - T)$$

Figure 13: Matlab implementation

```
%Finding Phi -1
R_inv = (cost - sint*I);
phi_inv = R_inv.*(Z - T);
phi_x_i = real(phi_inv); phi_y_i = imag(phi_inv);
```

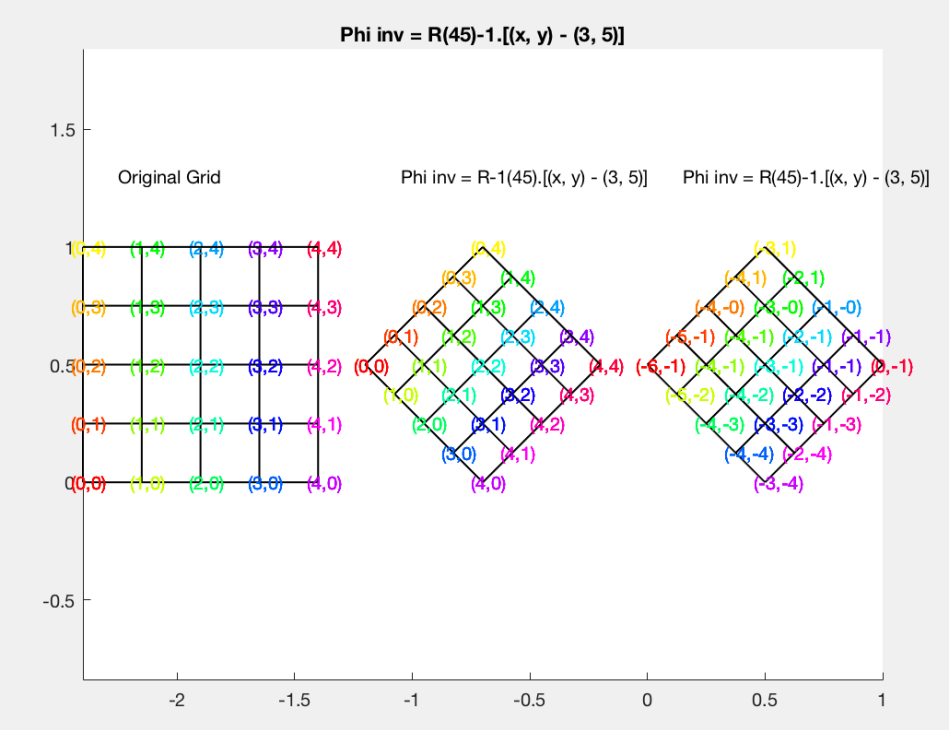

Figure 14: Phi inverse Transformation

Task 2

- The transformation was applied to the mugshot grayscal image:
  - The transform was applied to find the new frame (size) of the image.
  - The inverse transformation was applied to find the coordinates of the original image to transform it.
  - User defined valueAt() function was called on each coordinate of the original image to interpolate the values
  - **Note**: Matlab image coordinates are different form the Cartesian coordinates so the following rotation matrices were used for rotation around the origin(top left corner of the image).

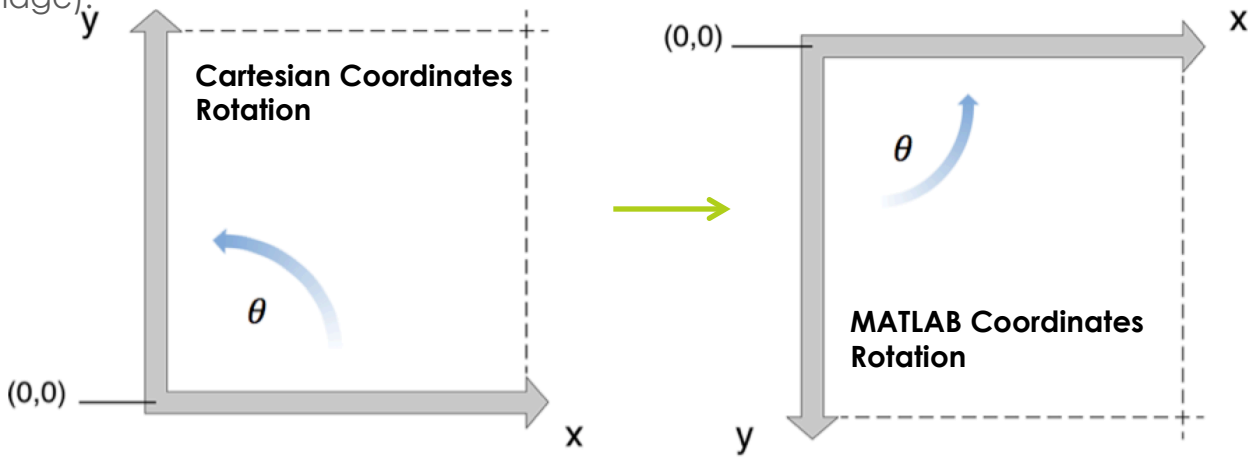

Figure 15: Cartesian vs. Matlab Coordinates

$$\begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \rightarrow \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix}$$

Task 2

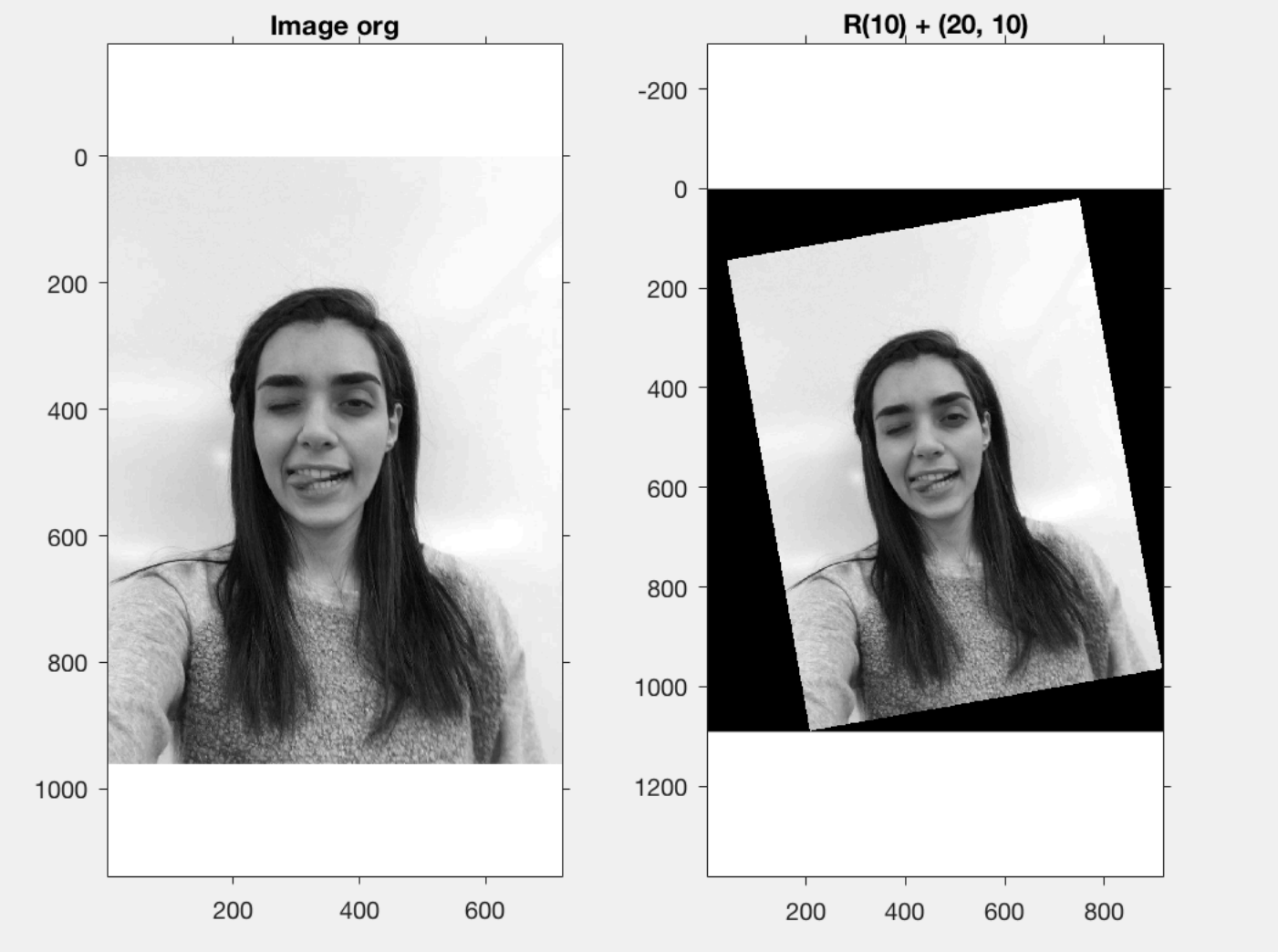

Figure 16: Original image vs. Rotation(10) + (20, 10)

Task 2

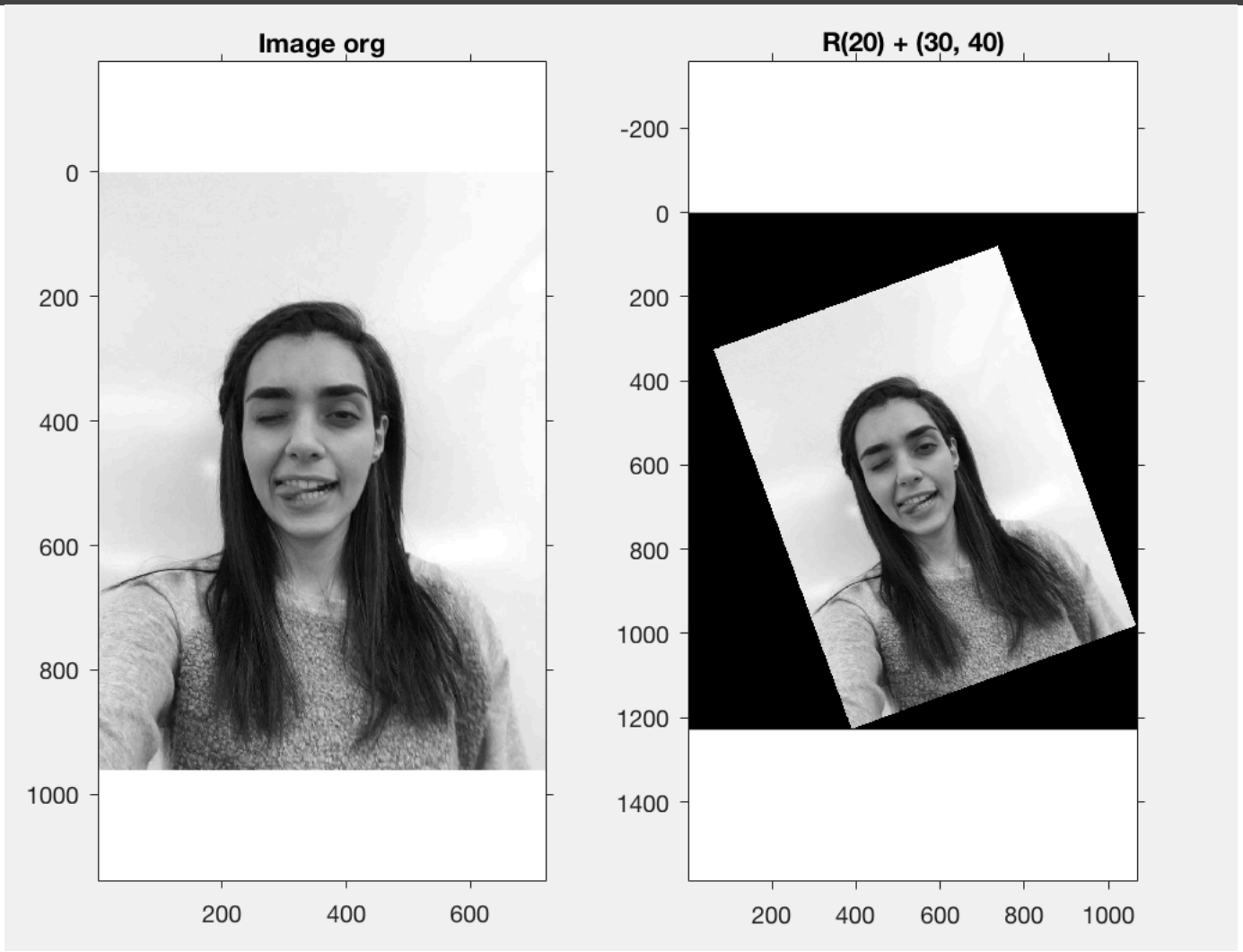

Figure 17: Original image vs. Rotation(20) + (30, 40)

Task 2

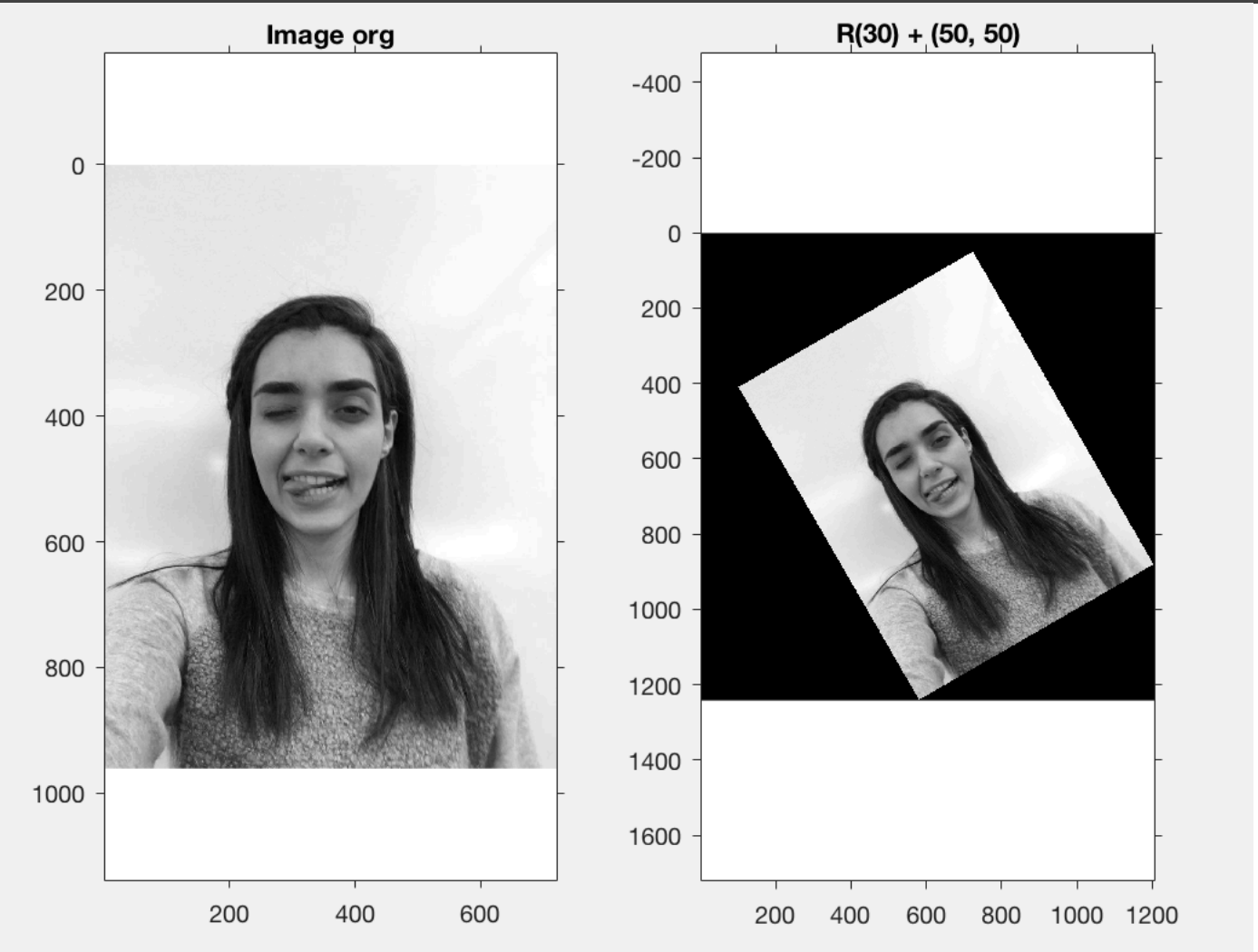

Figure 18: Original image vs. Rotation(30) + (50, 50)

Task 2

Figure 19: Original image vs. Rotation(50) + (100, 150)

Task 2

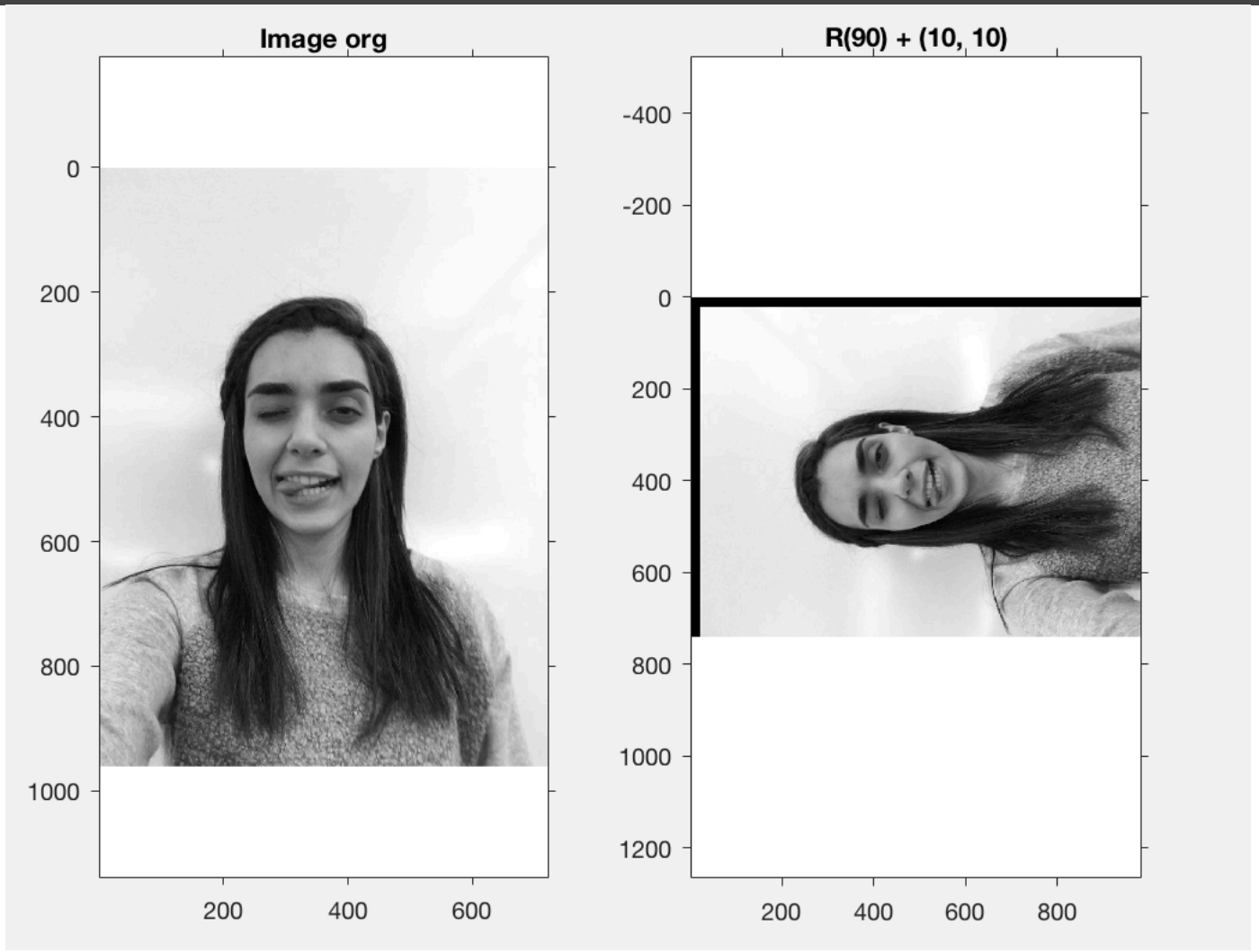

Figure 20: Original image vs. Rotation(90) + (10, 10)