Image Analysis and Object Recognition

Assignment 1

Points and Lines in the Plane and Introducing Octave

Gruppe 04
Members:
Balkis Saidi (124785)
Ahsan Qadeer (123892)
Simon André Ruth

Task1:

a)
We have x = (3,6,3) and y = (6,12,6) two identical points
After normalization: x = (1,2,1) and y = (1,2,1)
The cross product of x and y will compute the connecting line between them.
The cross product gives us the result (0,0,0). This doesn't represent a line.

b)
The general line x cos φ + y sin φ = d
We consider L1 = (cos φ, sin φ, - d)
We have L= (0, 0, 1)T, a line at infinity.
The result of the cross product of L1 and L is (sin φ, -cos φ, 0). The intersection of L1 and L is at infinity. We interpret that the intersection of any general line with a line at infinity is always at infinity.

c)
The horizon is at infinity. we consider x1, x2, x3 three points on the horizon.
x1= (u1, v1, 0)T
x2= (u2, v2, 0)T
x3= (u3, v3, 0)T
We compute the determinant of x1, x2, x3

$$\det[x1, x2, x3] = \det \begin{matrix} u1 & v1 & 0 \\ u2 & v2 & 0 \\ u3 & v3 & 0 \end{matrix}$$

det [x1, x2, x3] = 0
we conclude that x1, x2, and x3 are collinear and lie on a straight line.

Task3:

a)

```
% Part a

% Points represented in homogenous coordinate system
x = [2; 3; 1];
y = [-4; 5; 1];

% The cross product of x and y will give us the connecting line between them
l = cross(x, y);
disp('Connecting line between the two points:'), disp(l);

% ==============================================================================
```

b)

```
% Part b

% The translation matrix mT for movement in the direction (6, -7)
mT = [
1 0 6;
0 1 -7;
0 0 1
];

% Multiplying the two points by the translation matrix:
x1 = mT * x;
y1 = mT * y;

disp('The points x and y after translation are:');
disp('x1:'), disp(x1);
disp('y1:'), disp(y1);

% The rotation matrix mR for rotation by 15 degrees
phi = deg2rad(15);
mR = [
cos(phi) -sin(phi) 0;
sin(phi) cos(phi) 0;
0 0 1;
];

% Multiplying the points x1 and y1 by the rotation matrix:
x2 = mR * x1;
y2 = mR * y1;

disp('The points x1 and y2 after rotation are:');
disp('x2:'), disp(x2);
disp('y2:'), disp(y2);
```

```
% The scale matrix mS for scaling by 8 units
lambda = 8;
mS = [
8 0 0;
0 8 0;
0 0 1;
];

% Multiplying the points x1 and y1 by the rotation matrix:
x3 = mS * x2;
y3 = mS * y2;

disp('The points x2 and y3 after scaling are:');
disp('x3:'), disp(x3);
disp('y3:'), disp(y3);

% ===============================================================================
```

c)

```
% Part c

% Performing transformations with one matrix
mTransformation = mS * mR * mT;

% Taking the inverse and then the transpose of the transformation matrix
mTransformationInv = inv(mTransformation);
mTransformationInvTrans = transpose(mTransformationInv);

% Multiplying mTransformationInvTrans with the line
lTransformation = mTransformationInvTrans * l;

xIntercept = (-lTransformation(3)/lTransformation(1));
yIntercept = (-lTransformation(3)/lTransformation(2));

disp('The transformation applied to the line results in:');
disp(lTransformation);

disp('X intercept of the line:');
disp(xIntercept);

disp('Y intercept of the line:');
disp(yIntercept);

% ===============================================================================
```

d)

```
% Part d

% Getting the connecting line between the transformed points

l2 = cross(x3, y3);
disp('Connecting line between the two points x3 and y3:'), disp(l2);

xIntercept2 = (-l2(3)/l2(1));
yIntercept2 = (-l2(3)/l2(2));

disp('X intercept of the new line:');
disp(xIntercept2);

disp('Y intercept of the new line:');
disp(yIntercept2);

% ==============================================================================
```

Since both intercepts are same, both the set of transformed points lie on the same transformed line.

e)

```
% Part e

plot([x(1) y(1)], [x(2) y(2)], 'go-', [x3(1) y3(1)], [x3(2) y3(2)], 'b*-');
xlabel('x');
ylabel('y');
```