**Photogrammetric Computer Vision WiSe2024**

**Assignment 1**

**Group 30**

Muhammad Farzad Ali(126559)

Cesar Fernando Gamba Tiusaba(127138)

Chun Ting Lin(127130)

Parya Miraki(127187)

1. **Points and Lines in the Plane**
2. You would like to compute the connecting line between two 2D points. What happens if the two points are identical?

The line will be undefined, because if we want to connect 2 points with coordinates (X1,Y1) and (X2,Y2), the line equation between these point will be

If both points have the same coordinates the slope of the line will be 0/0 that is not possible

1. Where does the general line x cos φ + y sin φ= d intersect the line (0, 0, 1)T given in homogeneous coordinates? How can this point be interpreted?

We do the cross product between the general line and the line (0,0,1)T

This intersection point (, ,0) lies on the line at infinity (since the third coordinate is zero), which indicates that the two lines intersect at a point on the horizon. Geometrically, this point can be interpreted as the direction vector of the original line in Euclidean space, representing the "vanishing point" of parallel lines with the same direction as L.

1. Show that the horizon is a straight line by showing that three points on the horizon are always collinear. (Hint: use projective geometry.)

We have three points, with the next coordinates

Where C1, C2, C3, are 0 meaning they all lie on the line at infinity.

Following the duality principle that says that

Since the determinant is zero, P1​, P2, and P3​ are indeed collinear, meaning that any three points on the horizon lie on a single straight line. Therefore, the horizon is itself a straight line.

1. **First Steps in MATLAB**

**2.1** the MATLAB file is attached to the zip file as” *assignment\_1\_2.1\_Task*”.

% Given points

x = [2; 3; 1]; % Adding 1 for homogeneous coordinates

y = [-4; 5; 1]; % Adding 1 for homogeneous coordinates

% Compute line l as the cross product of x and y

l = cross(x, y);

disp('Connecting line l between points x and y:');

disp(l);

% Translation vector

t = [6; -7];

% Rotation angle in radians

phi = 15 \* (pi / 180);

% Scaling factor

lambda = 8;

% Translation: move x and y by vector t

x\_trans = [x(1) + t(1); x(2) + t(2); 1];

y\_trans = [y(1) + t(1); y(2) + t(2); 1];

% Rotation matrix

R = [cos(phi), -sin(phi); sin(phi), cos(phi)];

% Rotate the translated points

x\_rot = R \* x\_trans(1:2);

y\_rot = R \* y\_trans(1:2);

% Scale the rotated points

x\_prime = lambda \* x\_rot;

y\_prime = lambda \* y\_rot;

disp('Transformed point x\_prime:');

disp(x\_prime);

disp('Transformed point y\_prime:');

disp(y\_prime);

% Homogeneous transformation matrix for translation

T = [1, 0, t(1); 0, 1, t(2); 0, 0, 1];

% Homogeneous transformation matrix for rotation

R\_hom = [cos(phi), -sin(phi), 0; sin(phi), cos(phi), 0; 0, 0, 1];

% Homogeneous transformation matrix for scaling

S = [lambda, 0, 0; 0, lambda, 0; 0, 0, 1];

% Combined transformation matrix

transformation = S \* R\_hom \* T;

% Apply transformation to line l

l\_prime = transformation' \* l;

disp('Transformed line l\_prime:');

disp(l\_prime);

% Convert x\_prime and y\_prime back to homogeneous coordinates

x\_prime\_hom = [x\_prime; 1];

y\_prime\_hom = [y\_prime; 1];

% Check if x' and y' lie on the line l'

is\_x\_on\_l = abs(l\_prime' \* x\_prime\_hom) < 1e-10;

is\_y\_on\_l = abs(l\_prime' \* y\_prime\_hom) < 1e-10;

if is\_x\_on\_l

disp('Transformed point x\_prime lies on transformed line l\_prime');

else

disp('Transformed point x\_prime does not lie on transformed line l\_prime');

end

if is\_y\_on\_l

disp('Transformed point y\_prime lies on transformed line l\_prime');

else

disp('Transformed point y\_prime does not lie on transformed line l\_prime');

end

**2.2** If x′ and y′ do not lie on l′, it could be due to:

1. Numerical Inaccuracies: Small errors from rotation and scaling transformations.

2. Inconsistent Transformations: Slight mismatches in how transformations are applied to the points versus the line.

3. Order of Transformations: If the order (translation, rotation, scaling) differs between the points and line, it can lead to misalignment.