Assignment 3 Tutorial

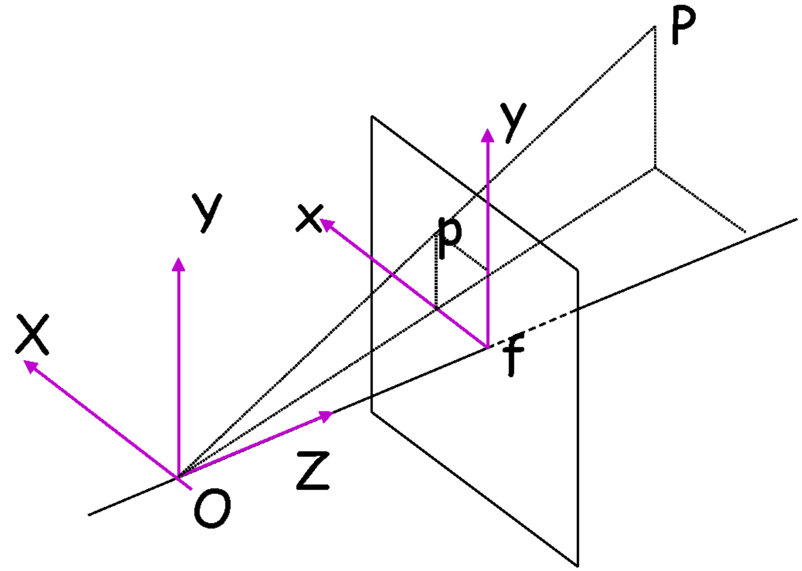
Objectives

- Get familiar with perspective and weak perspective projection techniques
- Practically compare performance of these projection methods using MATLAB

Projective Geometry

Perspective Camera Model

The 3D point P (X,Y,Z) is projected onto the image plane at 2D point p (x,y)



From similarity of triangles,

$$x = f\,\frac{X}{Z}$$

$$y = f\,\frac{Y}{Z}$$

Perspective Camera Model

$$x = f\frac{X}{Z}$$

$$y = f\frac{Y}{Z}$$

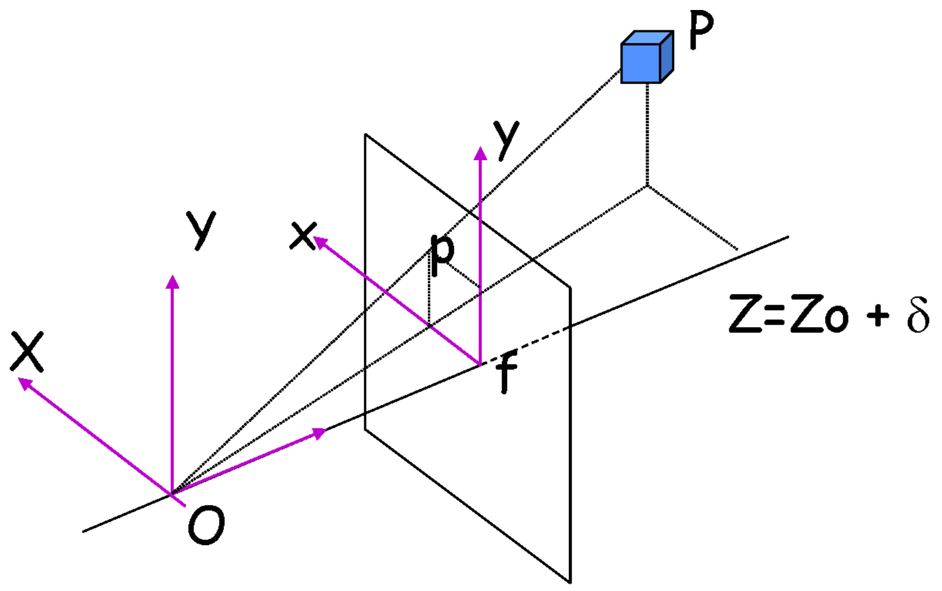
EX 1:

(-1, 0, 2)	Z=2
(1, 0, 5)	Z=5
(0, 1, 4)	Z=4
(0, -1, 3)	Z=3

Projective Geometry

Weak Perspective Camera Model

If the average depth of the scene, Z0, is much larger than relative distance between any two scene points along the optical (Z) axis, the approximation holds



$$x \approx f \frac{X}{Z_0}$$

$$y \approx f \frac{Y}{Z_0}$$

Weak Perspective Camera Model

$$x \approx f \frac{X}{Z_0}$$

$$y \approx f \frac{Y}{Z_0}$$

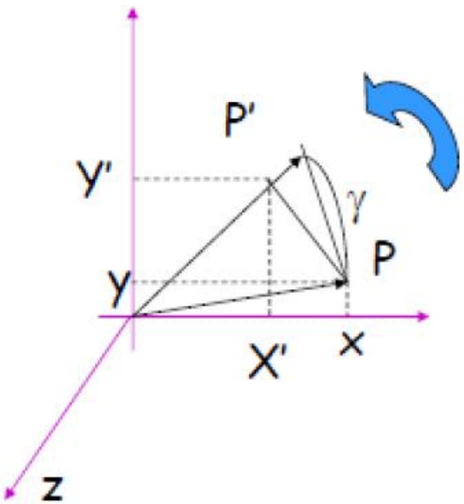
EX 1:

(-1, 0, 2)
(1, 0, 5)
(0, 1, 4)
(0, -1, 3)

$Z_0 = (2+5+4+3)/4$

3D Rotation of Points

The 3D point P (X,Y,Z) is rotated to 3D point P'(X', Y', Z')



$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \mathrm{R} \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Guidance for Assignment # 3

- **Given**:

  Perspective camera with a focal point at (0,0,0), a **focal length** of 1, and an image plane equal to the z=1 plane.

  Object O with four 3D co-ordinates:

  {(-1, 0, 2), (1, 0, 5), (0, 1, 4), (0, -1, 3)}

Tips:

- Use matrices in MATLAB.
- Make a function that combines four points and provides the output as a matrix

Guidance for Assignment # 3

Task 1:

- Write a function that will take as input a set of 3D points, and returns as output a set of 2D points that are the projection of the 3D points with the **perspective camera**.

Tips:

- Use the equations for calculating 2D coordinates from 3D coordinates in Projective Transformation

Guidance for Assignment # 3

Task 2:

- Write a second function to project the points using weak perspective projection. Note that the scale factor should be based on the average distance to all points.

Tips:

- ❑ First, compute the average of the distances of the points from camera along the optical axis
- ❑ Use the equations for Projective Transformation applicable for Weak Perspective Model

Discussion of Assignment *contd*…

Task 3:

- **Plot** all the perspective and weak perspective points.

Tips:

```
% coordinate of perspective camera projections
xc = ctwoD(:,1);
yc = ctwoD(:,2);

% coordinate of weak perspective camera projections
xw = wtwoD(:,1);
yw = wtwoD(:,2);

% Plot the transformed points from 3D to 2D
plot(xc,yc,'-ro',xw, yw,'-*b','MarkerSize',10)
hleg1 = legend('__________');
xlabel('_____')
ylabel('_____')
title('_______')
```

Discussion of Assignment *contd*…

Task 4:

- Image Difference: Write a function to compute the **sum of square differences** (SSD) between two sets of image points. That is, for point sets: $(p_1,\dots,p_n),\ (q_1,\dots,q_n)$

  compute: $$\sum_{i=1}^{n} \left\| p_i - q_i \right\|^2$$

Test it on the images generated in parts 2 and 3.

Tips:

- ❑ The function should take two sets of image points as input, and provide the SSD as output

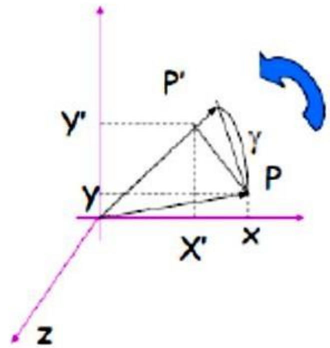
```
function difference = mySSD(p, q)

difference = sum(sum(abs(p-q)));
```

Discussion of Assignment *contd*…

Task 5:

- Write a function to **rotate the object** o **counter clock** wise by **45 degrees around the Z-axis**. We call the object after rotation as R.



$$R_x(\alpha)=\begin{bmatrix}1 & 0 & 0\\ 0 & \cos\alpha & -\sin\alpha\\ 0 & \sin\alpha & \cos\alpha\end{bmatrix}$$

$$R_y(\beta)=\begin{bmatrix}\cos\beta & 0 & \sin\beta\\ 0 & 1 & 0\\ -\sin\beta & 0 & \cos\beta\end{bmatrix}$$

$$R_z(\gamma)=\begin{bmatrix}\cos\gamma & -\sin\gamma & 0\\ \sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{bmatrix}$$

Repeat part **task 3 and task 4** for object **R**.

Tips:

$R = \begin{bmatrix}cos\theta & -sin\theta & 0\\ sin\theta & cos\theta & 0\\ 0 & 0 & 1\end{bmatrix}$, for P= (X, Y, Z), the rotated point P'= (X', Y', Z') is equal to $\begin{bmatrix}X'\\ Y'\\ Z'\end{bmatrix} = R \times \begin{bmatrix}X\\ Y\\ Z\end{bmatrix}$

```
function rotationMatirx = myrotz_cc(ang)
ang = _________________ % converting to radians
% the rotation matrix respect to z-axis
r = [________________________]; rotationMatirx = r;
```

Discussion of Assignment *contd*…

Task 5 Cont…

- **Now Plot** all the perspective and weak perspective points of the rotated 3D points (after Euclidian transformation) .
- Also you have to calculate the SSD

Tips:

```
figure
% coordinate of perspective camera projections
xc = rotated_c2D(:,1);
yc = rotated_c2D(:,2);
% cordinate of weak perspective camera projections
xw = rotated_w2D(:,1);
yw = rotated_w2D(:,2);
plot(xc,yc,'-ro',xw,yw,'-*b','MarkerSize',10)
hleg1 = legend('Perspective','Weak');
xlabel('X')
ylabel('Y')
title('2D Projection of the Object R','FontSize',12)
```