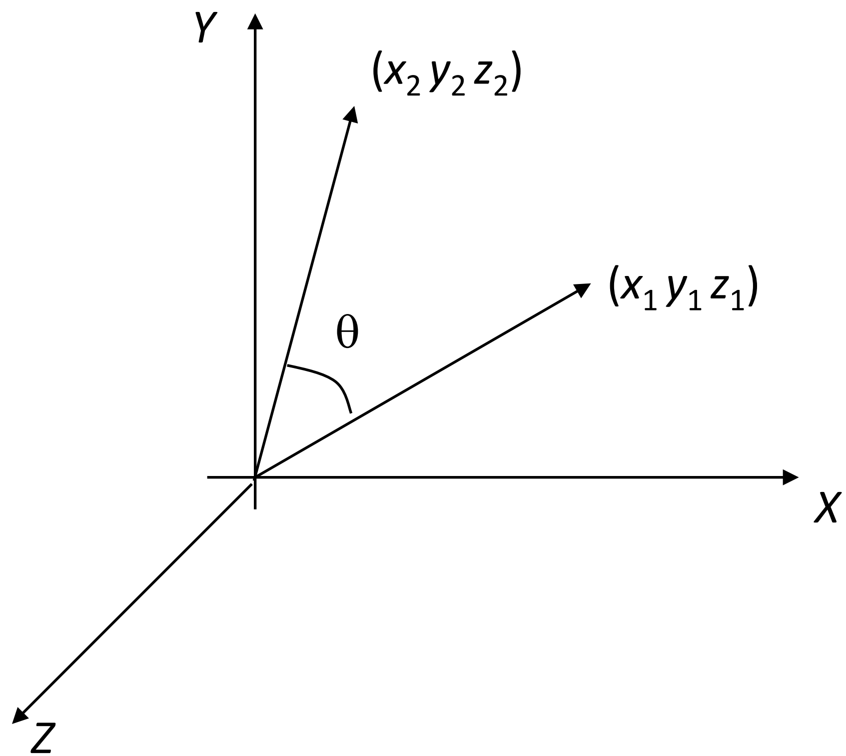
3D Transformations: Review

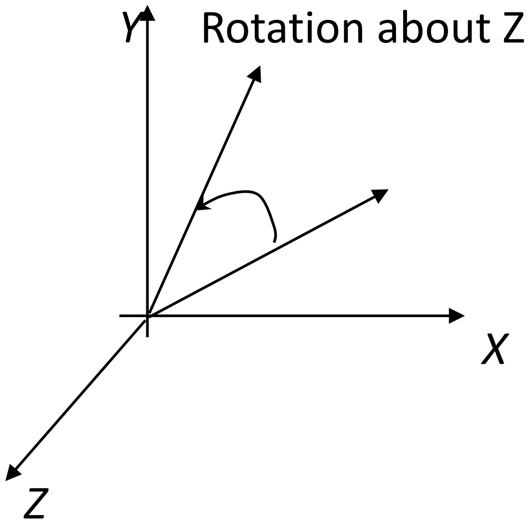
3D Rotation

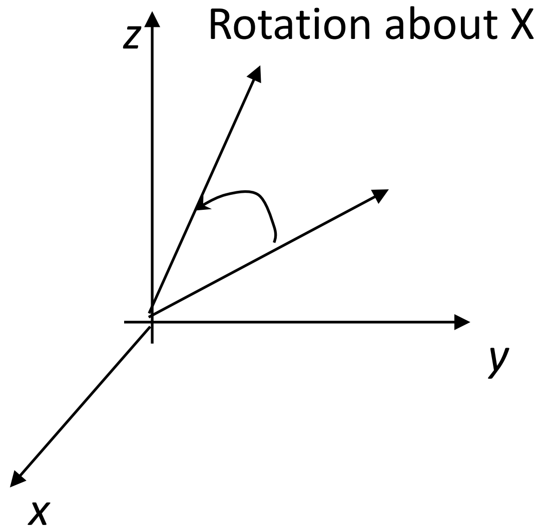
- Rotation about *Z*-axis
- Z-coordinate will not change
- Z’ = Z
- If we ignore the Z-coordinate, it is 2-D transformation in XY plane

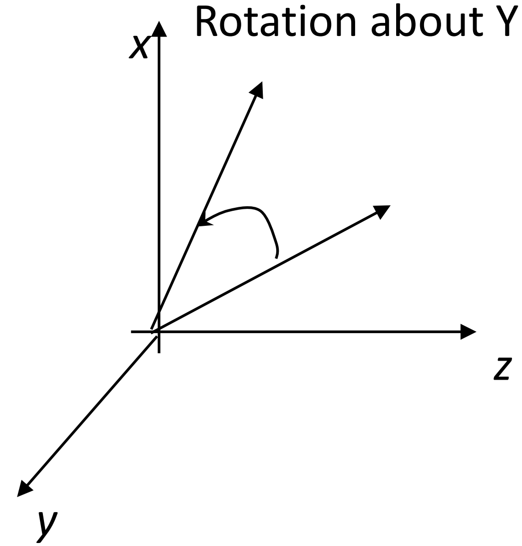


$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Rotation about Principal Axes







$X' = X\cos\alpha - Y\sin\alpha$	$Y' = Y\cos\gamma - Z\sin\gamma$	$Z' = Z\cos\beta - X\sin\beta$
$Y' = X\sin\alpha + Y\cos\alpha$	$Z' = Y\sin\gamma + Z\cos\gamma$	$X' = Z\sin\beta + X\cos\beta$
$Z' = Z$	$X' = X$	$Y' = Y$

Rotation about Principal Axes

$X' = X\cos\alpha - Y\sin\alpha$ $Y' = X\sin\alpha + Y\cos\alpha$ $Z' = Z$	$Y' = Y\cos\gamma - Z\sin\gamma$ $Z' = Y\sin\gamma + Z\cos\gamma$ $X' = X$	$Z' = Z\cos\beta - X\sin\beta$ $X' = Z\sin\beta + X\cos\beta$ $Y' = Y$

Rotation about Z

$$\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation about X

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma & 0 \\ 0 & \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotation about Y

$$\begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Concatenation of Rotations

- Rotation around X by γ followed by rotation around Y by β followed by rotation around Z by α

$$R = R_\alpha^Z R_\beta^Y R_\gamma^X$$

$$R = \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix}$$

Interpretation of Rotation Matrices

$$\begin{bmatrix} -0.8256 & 0.40388 & -0.39404 \\ -0.20084 & -0.86294 & -0.46367 \\ -0.5273 & -0.30367 & 0.79356 \end{bmatrix}$$

How do I visualize this rotation?

Properties of Rotation Matrices

- A rotation matrix transforms its own rows onto the principal axes

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} r_{11} \\ r_{12} \\ r_{13} \end{bmatrix} = ?$$

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} r_{21} \\ r_{22} \\ r_{23} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} r_{31} \\ r_{32} \\ r_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Rotation about Arbitrary Axis

- To rotate about an axis **n** by an angle θ

1. Set up rotations such that **n** rotates onto one of the principal axis [How?]
2. Rotate about that axis by θ
3. Undo the transformations in step 1

Rotation about Arbitrary Axis

- Question: Given an arbitrary 3D rotation matrix, how can we find out the axis **n** and the angle θ that represents this rotation?

$$\begin{bmatrix} -0.8256 & 0.40388 & -0.39404 \\ -0.20084 & -0.86294 & -0.46367 \\ -0.5273 & -0.30367 & 0.79356 \end{bmatrix}$$

Given R on the left, how can we tell n and θ?

Eigenvectors and Values of a Rotation Matrix

- 3D rotation matrix has eigenvalues of 1, $\cos\theta + i\sin\theta$ and $\cos\theta - i\sin\theta$
- The eigenvector associated with the real eigenvalue represents the axis of rotation [proof?]

Summary: 3D Rotation Matrices

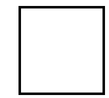
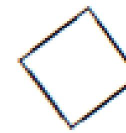
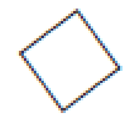
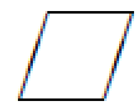
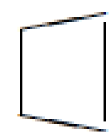
- Rotations about Principal Axes

$$\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 & 0 \\ \sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma & 0 \\ 0 & \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

About origin, in right handed coordinate system, counter clockwise when looking towards origin from positive axis

- Rotation matrix is orthonormal with Determinant of +1 and 3 dof
- Inverse of a rotation matrix is its transpose
- Concatenation of Rotations is also a rotation
- IMP: A rotation matrix transforms its own rows onto the principal axes
- Any 3D rotation matrix can be described as rotation about an axis **n** by an angle θ
- To rotate about given axis **n** by θ:
  - Rotate axes onto a principal axis
    - Two ways: by computing principal rotations or by composing appropriate matrix through cross products
  - Rotate about principal axes and then undo the earlier transformation
- To compute **n** and θ from a 3D rotation matrix
  - n is the eigenvector corresponding to the real eigenvalue of 1
  - θ can be computed by the other 2 eigenvalues, which are $\cos\theta \pm i\sin\theta$

Hierarchy of 3D Transformations

Transformation	Matrix	# DoF	Preserves	Icon
translation	$\begin{bmatrix} I \mid t \end{bmatrix}_{3\times 4}$	3	orientation	
rigid (Euclidean)	$\begin{bmatrix} R \mid t \end{bmatrix}_{3\times 4}$	6	lengths	
similarity	$\begin{bmatrix} sR \mid t \end{bmatrix}_{3\times 4}$	7	angles	
affine	$\begin{bmatrix} A \end{bmatrix}_{3\times 4}$	12	parallelism	
projective	$\begin{bmatrix} \tilde{H} \end{bmatrix}_{4\times 4}$	15	straight lines	

Summary: 2D and 3D Transformations

- Image Registration
- 2D Transformations
  - Scaling
  - Shear
  - Rotation
  - Translation
- Inverse Transformations
- Rotation about an arbitrary point
- Concatenation of transformations
- Order of transformations
- Factorization of Transformations
- Displacement Models
  - Rigid / Euclidean
  - Similarity
  - Affine
  - Projective
  - Billinear, biquadratic etc
- Recovering the best affine transformation
  - Least Squared Error solution
  - Pseudo inverse
- Image Warping
- 3D Transformations
  - Rotations about Principal Axes
  - Rotations about Arbitrary Axes
- Properties of Rotation Matrices

Camera Models: Introduction

Modeling a Camera

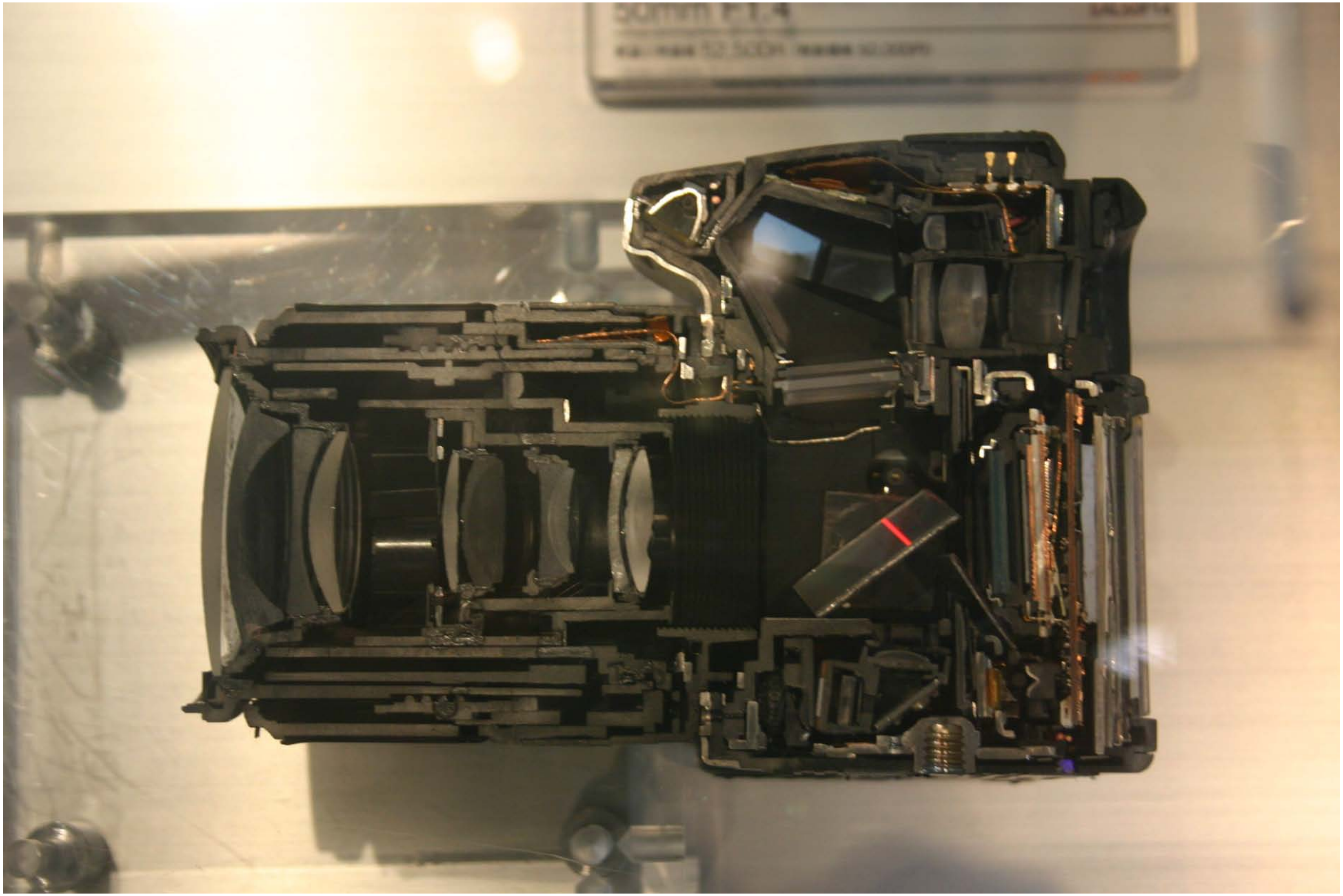
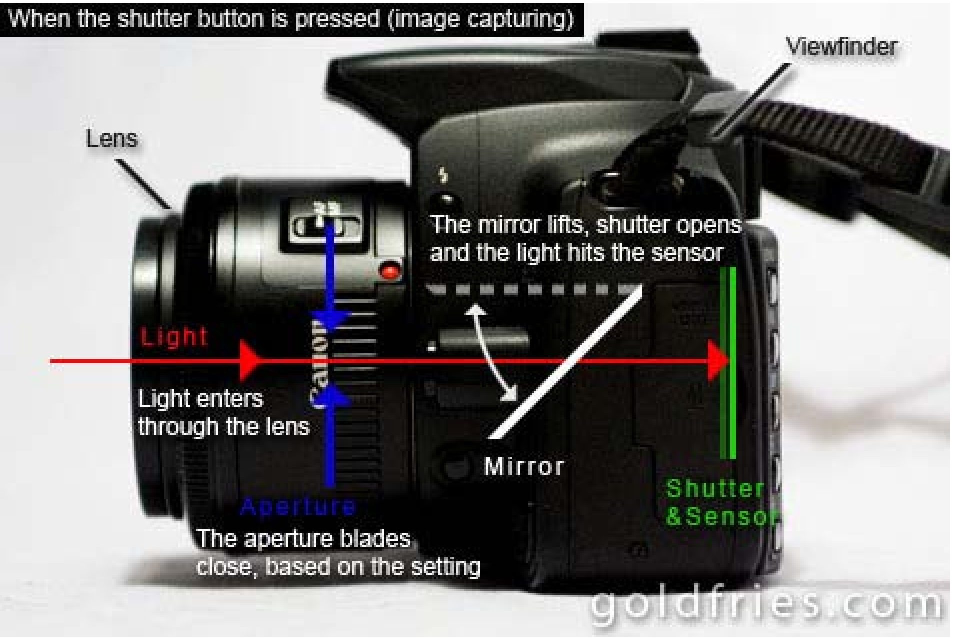
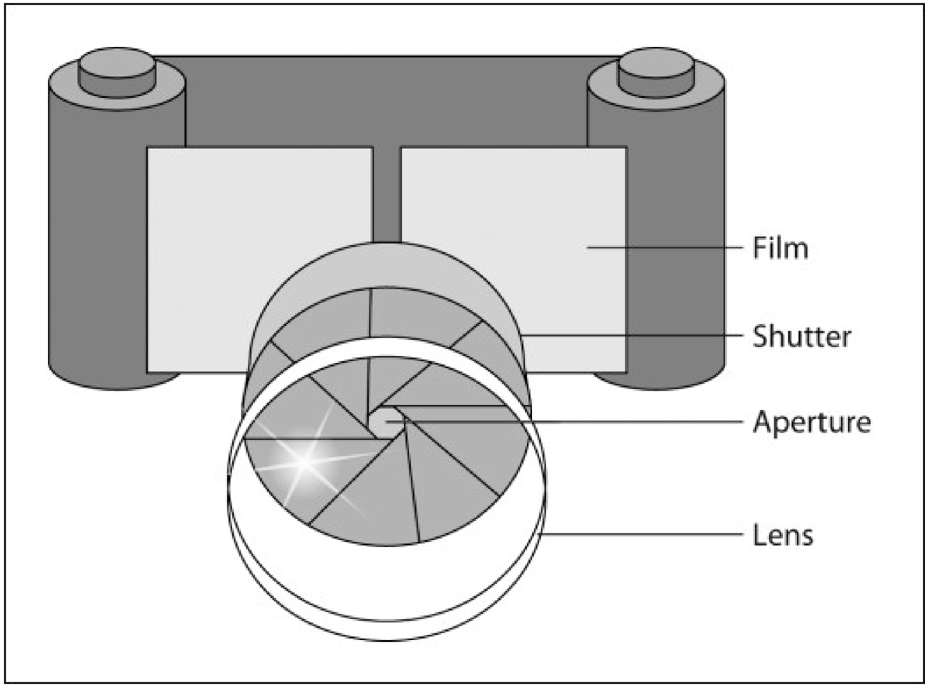


Image by Dr Yaser Sheikh, CMU

Modeling a Camera

- Shutter and Aperture





Aperture vs Shutter speed

- If **shutter speed** is doubled, and **aperture area** is doubled, the same amount of light should enter the camera
- Therefore, to shoot an image, there are several valid combinations of aperture and shutter speed
- High shutter speed: for fast moving objects
- Large aperture: low depth of field

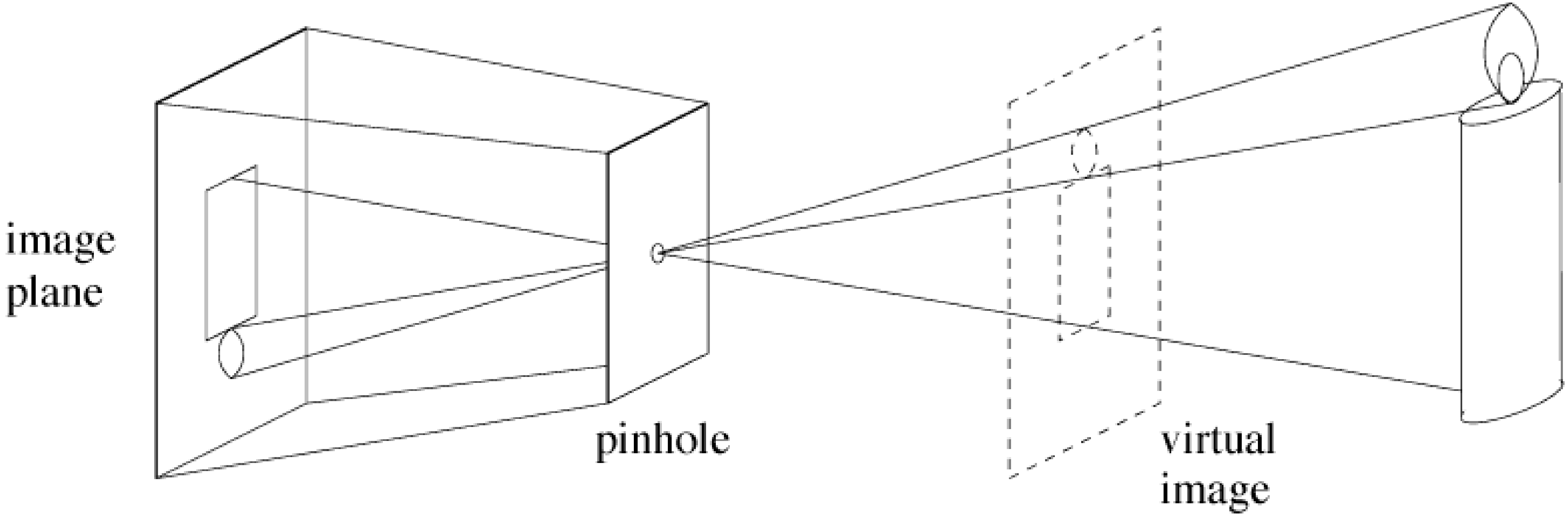
Focus

- In general, any single point on the film can have light coming from different directions
- Therefore a single point in the world may be mapped to several locations in the image
- This generates blur
- To remove blur, all rays coming from a single world point must converge to a single image point

Example of Shallow Depth of Field

Pinhole Camera

- Lens is assumed to be single point
- Infinitesimally small aperture
- Has infinite depth of field i.e. everything is in focus

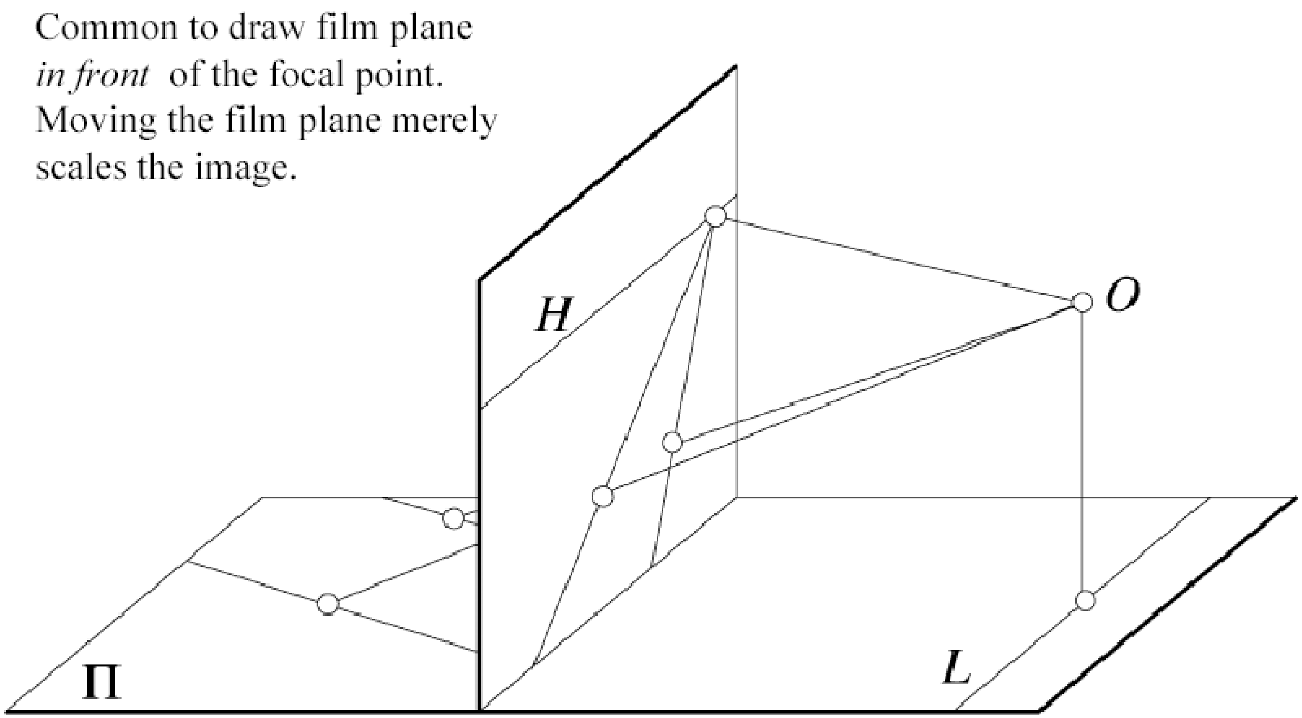


Pinhole Camera Properties: Distant objects are smaller

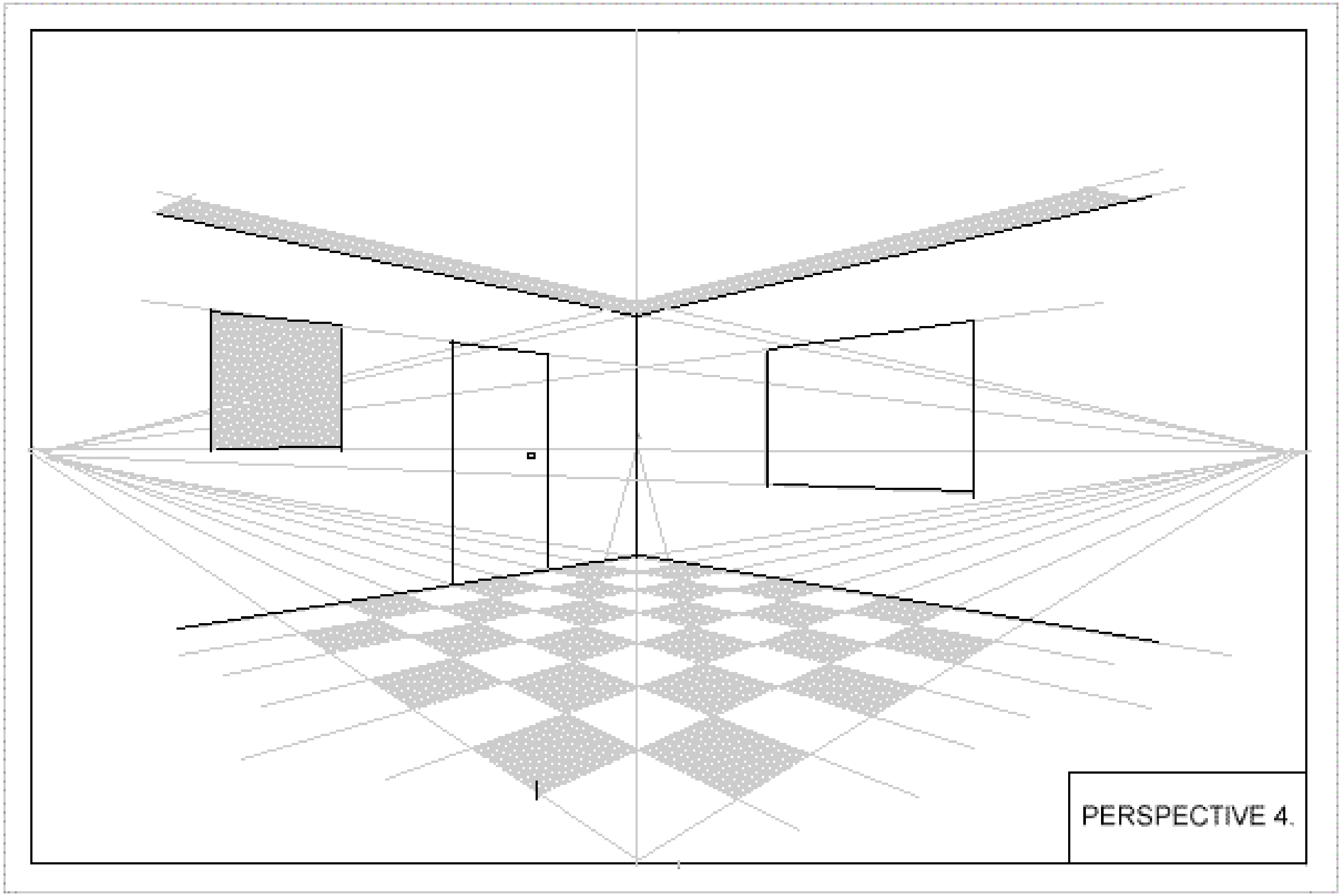
Slide Credit: Forsyth/Ponce http://www.cs.berkeley.edu/~daf/bookpages/slides.html
and Khurram Shafique, Object Video

Pinhole Camera Properties

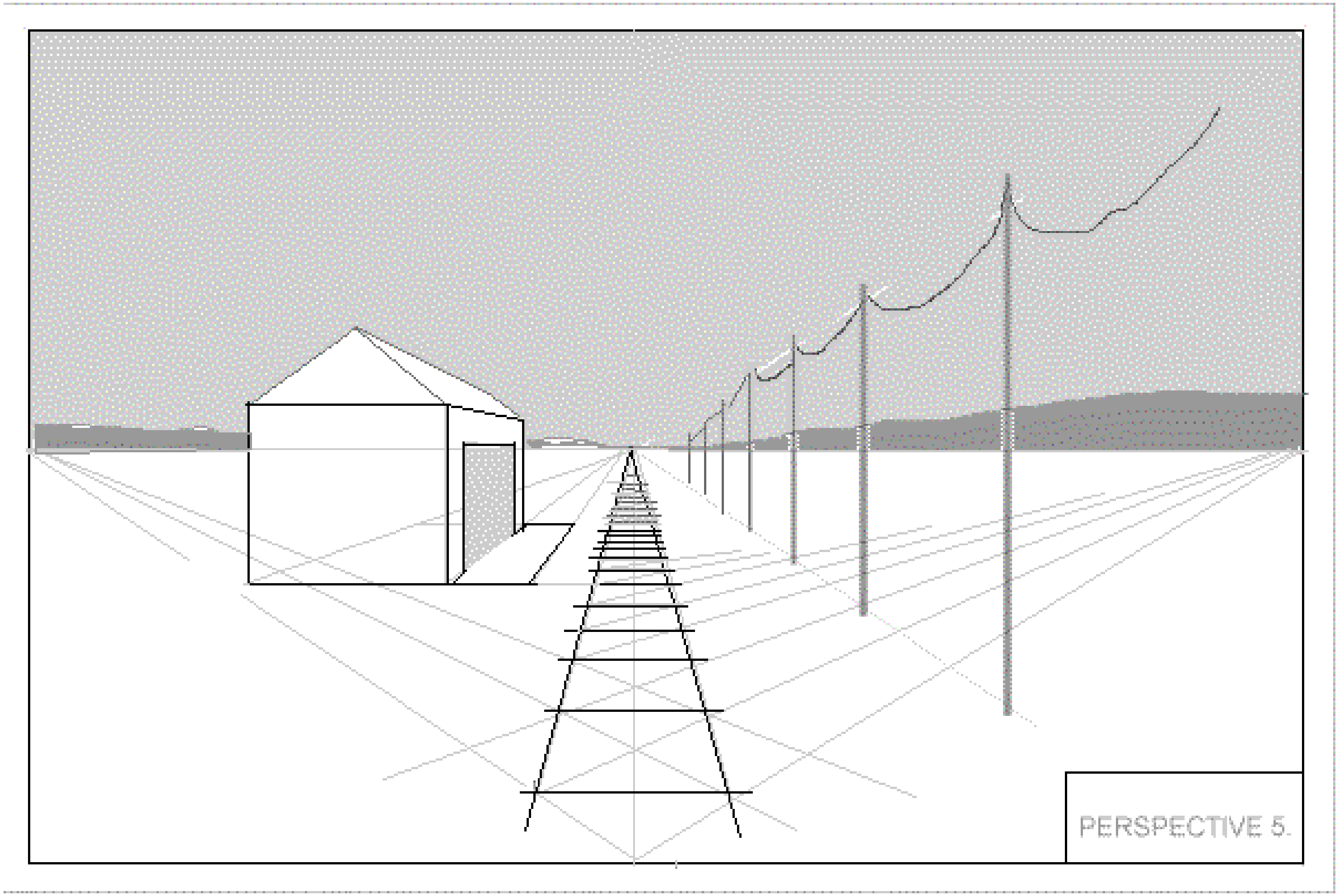
- Lines map to lines
- Polygons map to polygons
- Parallel lines meet



Pinhole Camera Properties: Parallel Lines Converge



Pinhole Camera Properties: Parallel Lines Converge

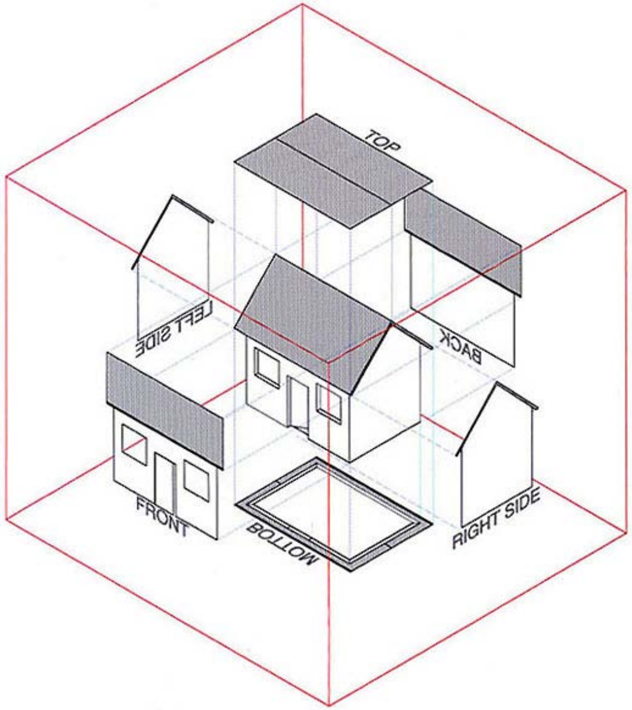


Pinhole Camera

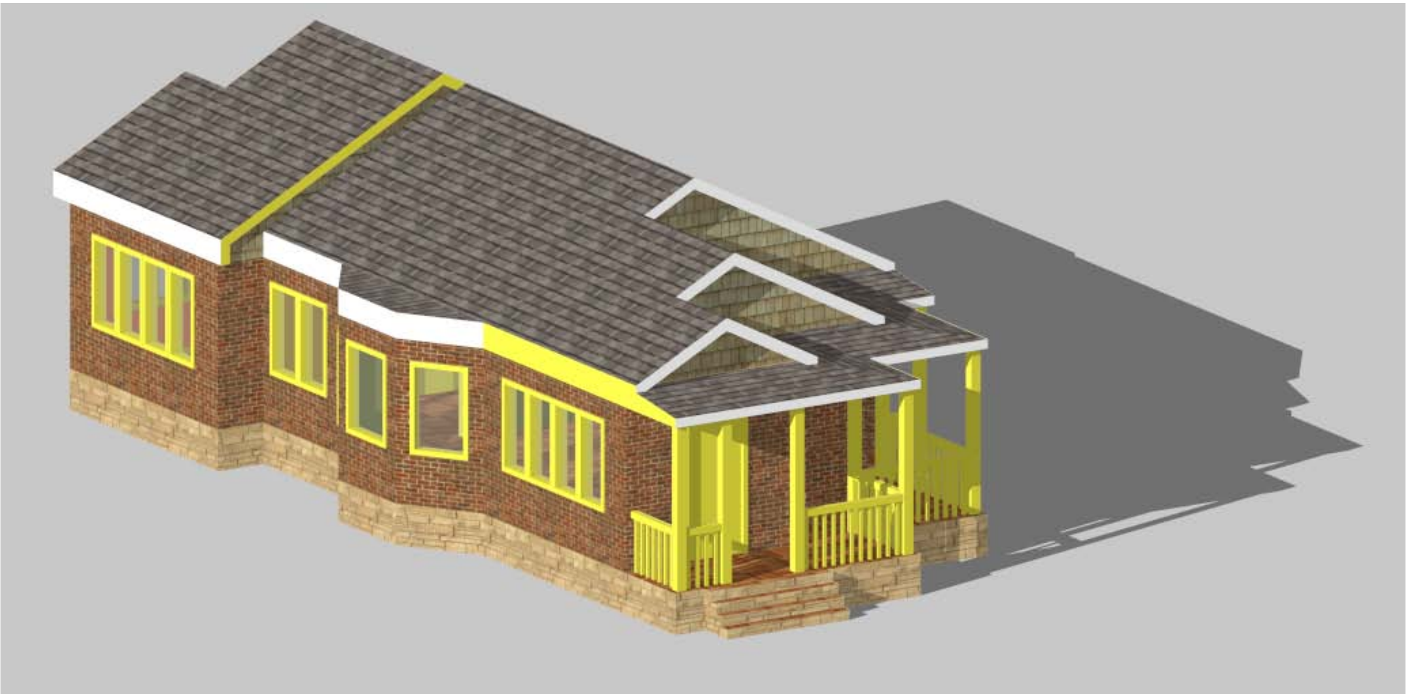
- Advantage
  - Because of small aperture, everything is in focus (infinite depth of field)
  - Simple construction
- Disadvantage
  - Small aperture requires high exposure time, often too long for practical purposes

Another Type of Camera: Orthographic Camera

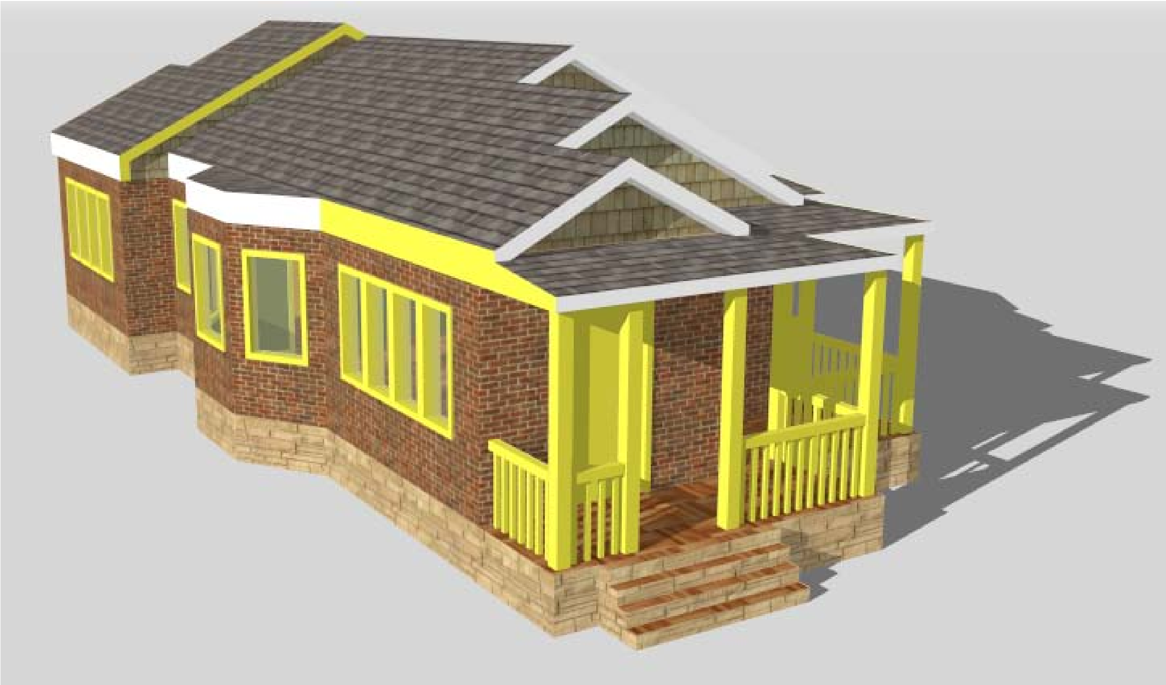
- Parallel Lines remain parallel and do not converge
  (also termed parallel projection)



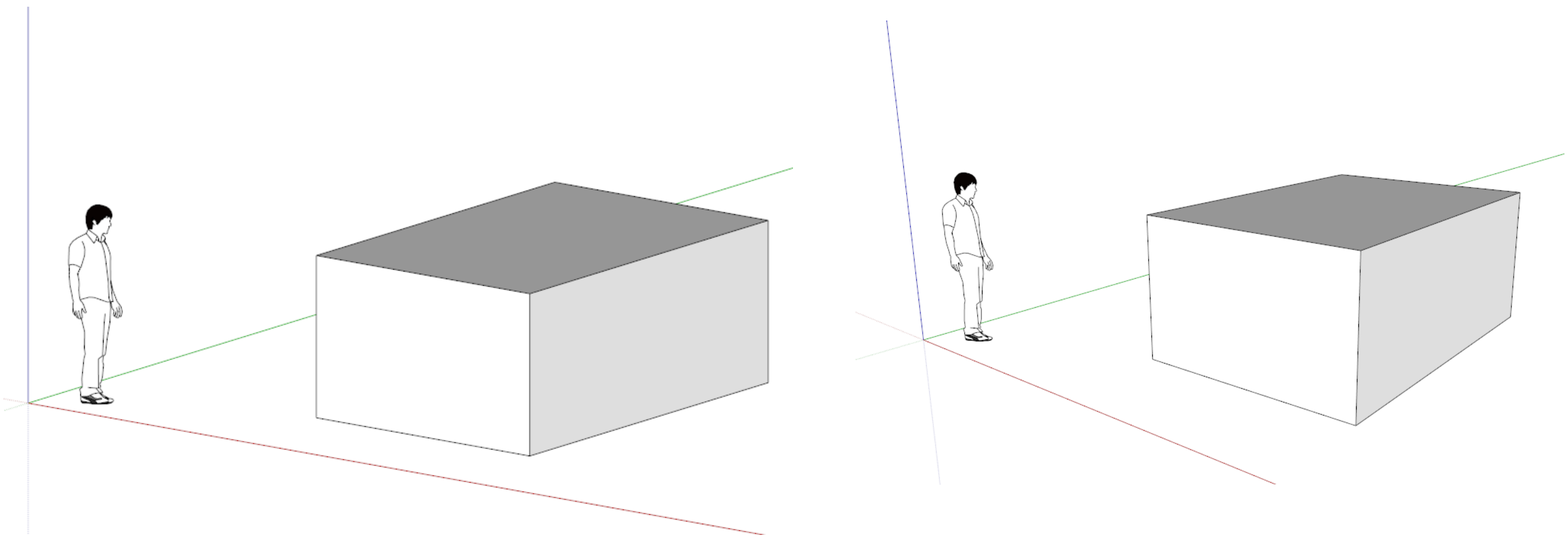
Type of Projection?



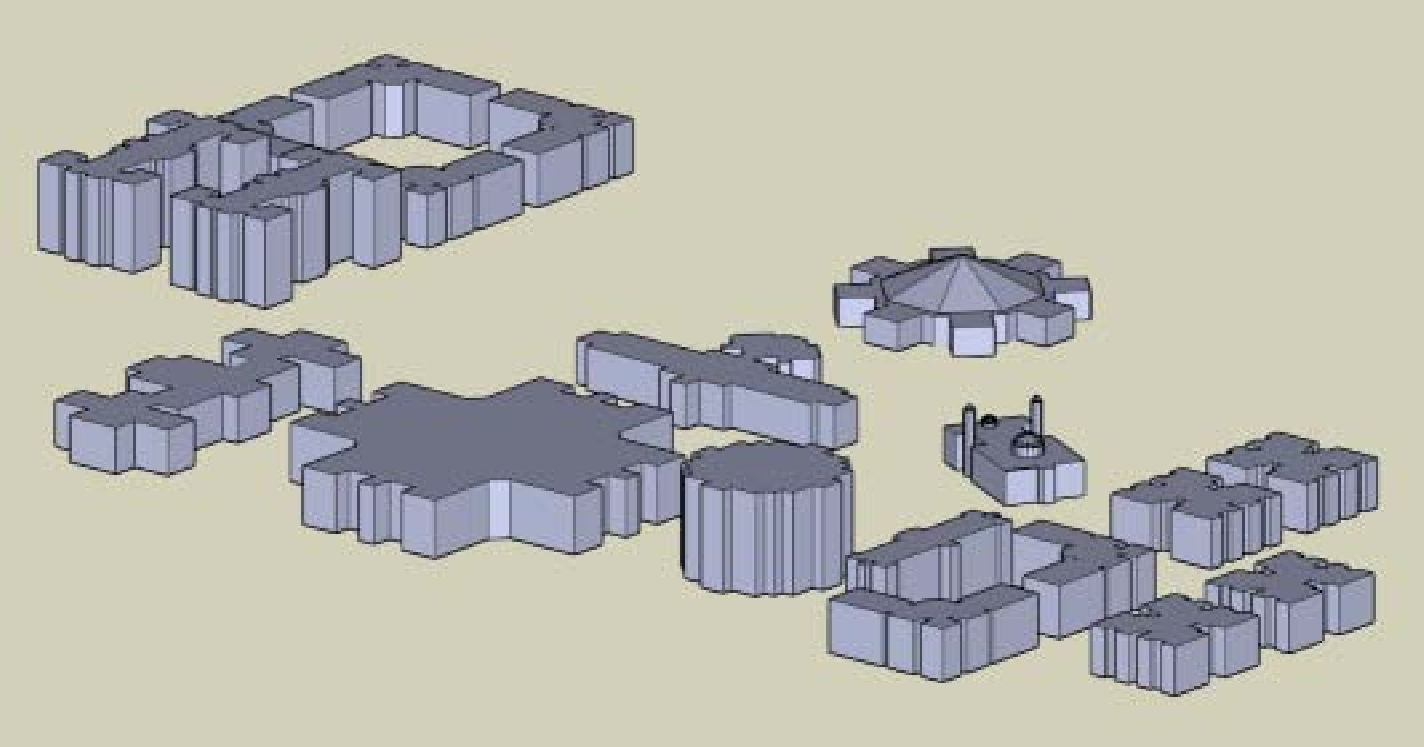
Type of Projection?



Type of Projection?



Type of Projection



Type of Projection?

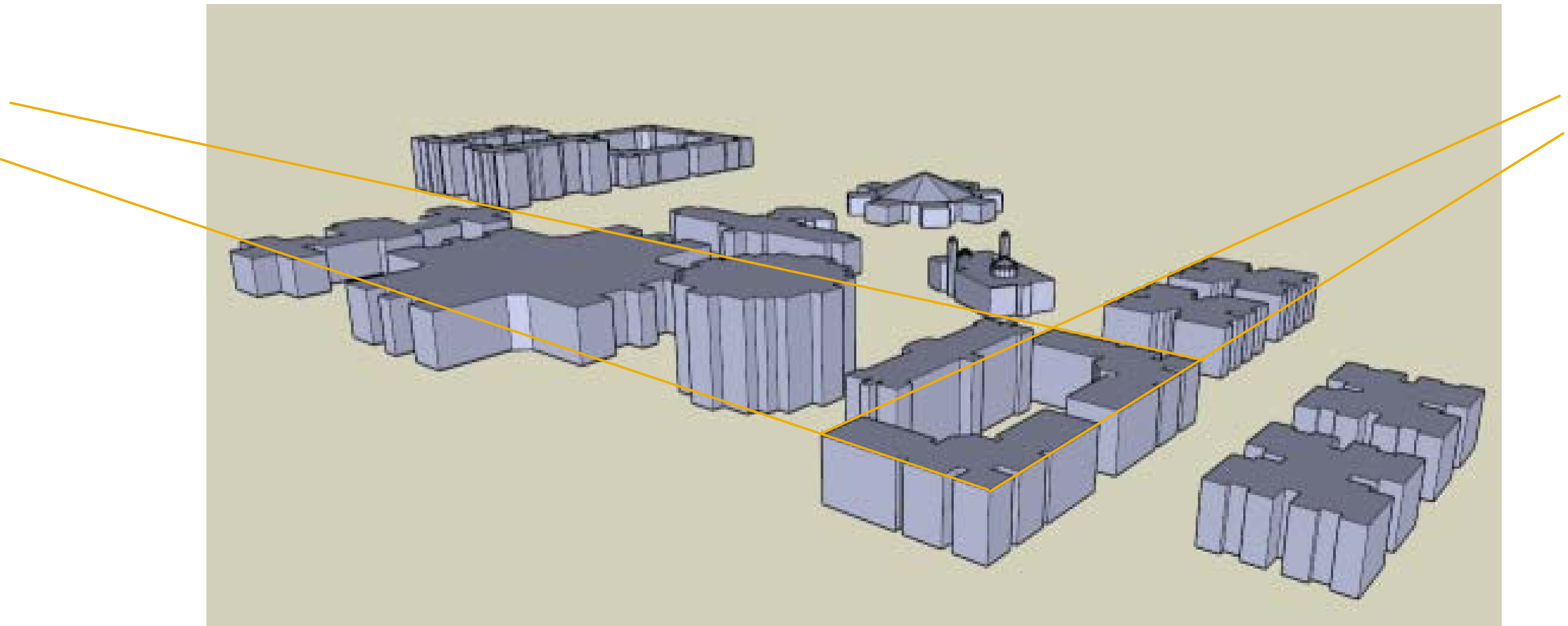
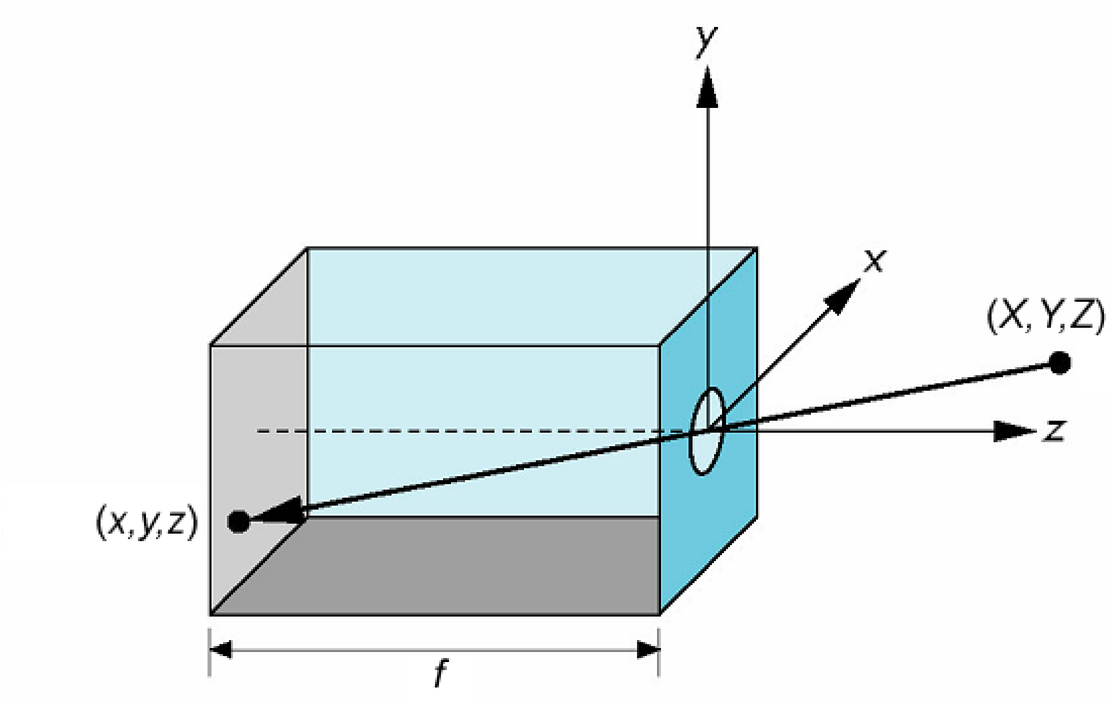


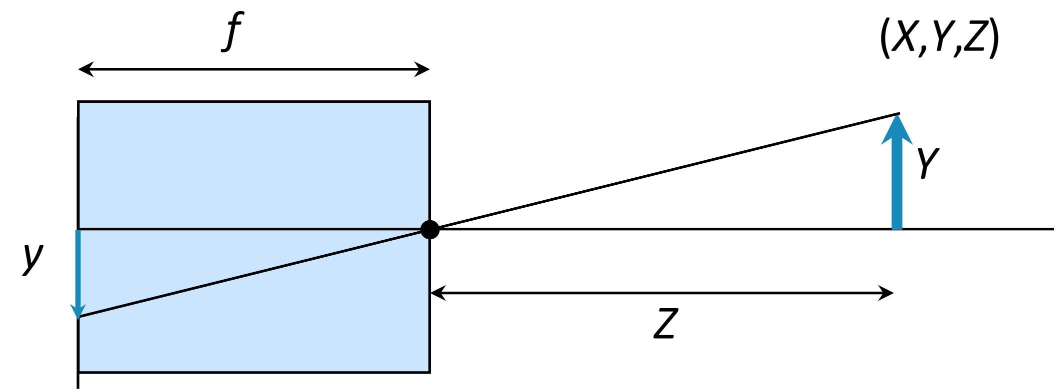
Image Formation: The Pin-Hole Camera

- Orient along z-axis
- World point (X,Y,Z) [in world coordinates]
- Image point at (x,y,z) [in real world coordinates]



Perspective Transform

Equation relating world coordinate and image coordinate?



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

$$y = -\frac{fY}{Z} \qquad x = -\frac{fX}{Z}$$

It is customary to use a negative sign to indicate that the image is always formed upside down

Perspective Transform

- This relates the camera frame to the real image frame
- Example:
  - I take the image of a person (2m tall) standing 4m away from the camera, with a 35 mm camera using the geometry shown previously. How high will be the image?
  - Answer: y = -(35)(2000)/4000 = -17.5mm
  - i.e, the image will be formed inverted of length 17.5 mm
- How to convert to pixel frame (i.e. what will be the coordinates of the head of the person in the image?

Perspective Transform

- Suppose I know that the size of the film is 8cm x 6cm, and that the resolution of the camera is 640 x 480 pixels
- Implies, the center of the image is at 4cm x 3cm from the corner, and is at location (240, 320)
- Image will first be made right side up
- 17.5mm out of 60mm is 140 out of 480 pixels
- Hence the coordinates of the head will be (240-140 in x, same in y) = (100, 320)

Perspective Transform

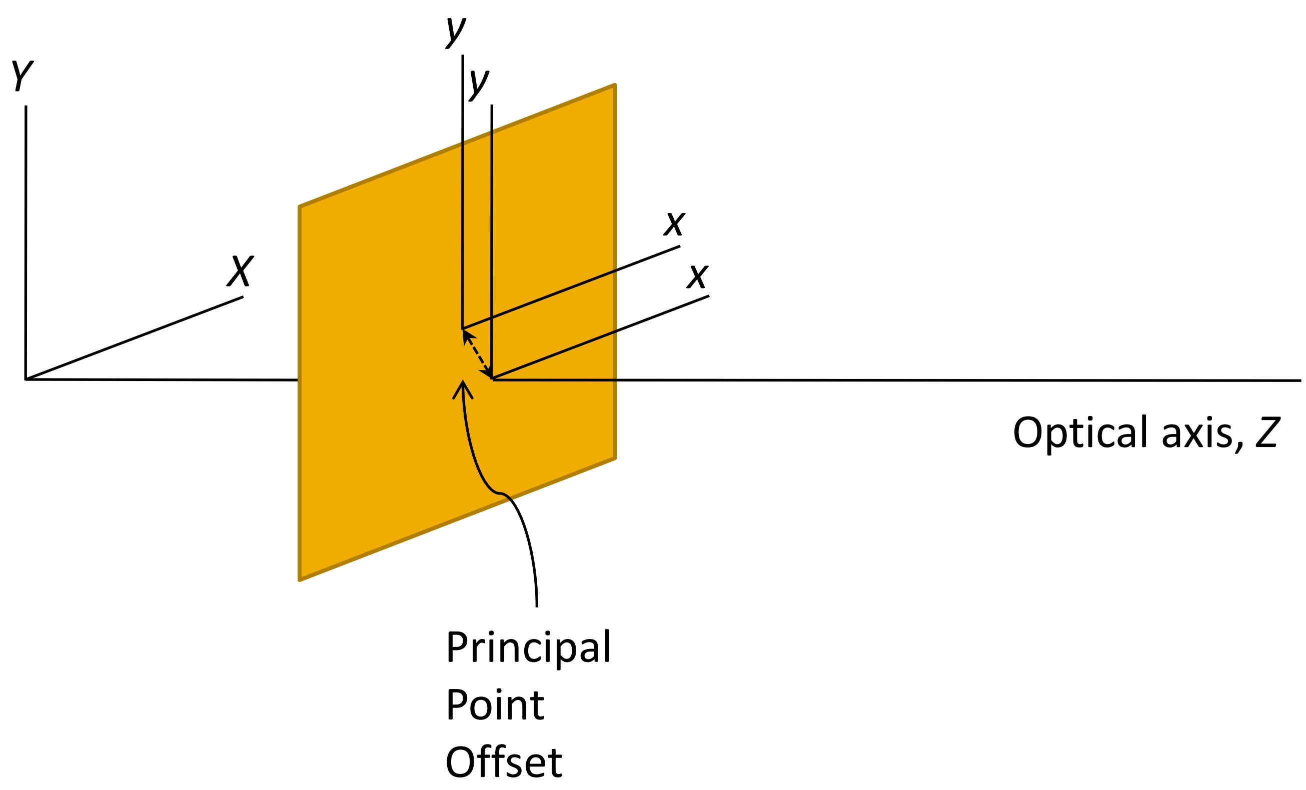
- We can write this as a matrix using the homogeneous coordinates

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

More General Perspective Camera Model

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x & 0 \\ 0 & m_y f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

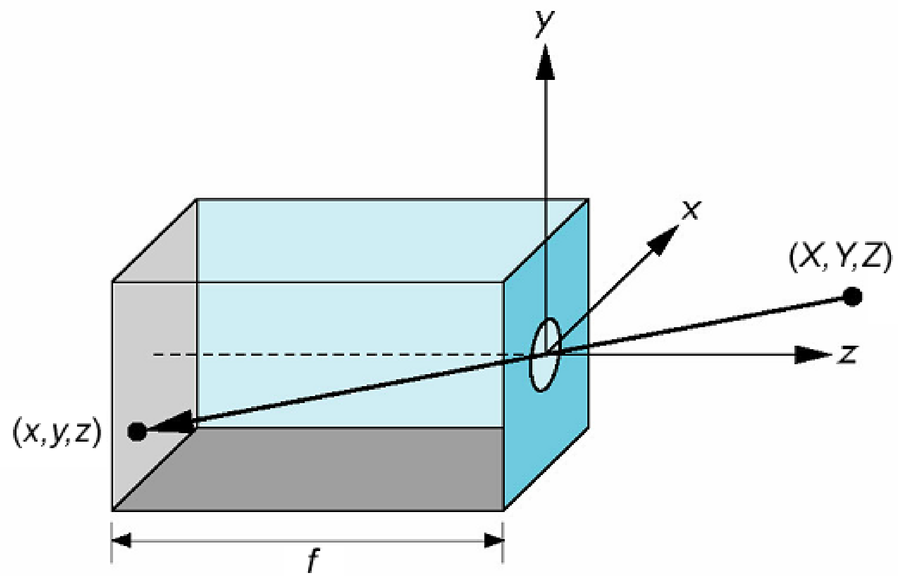
- m_x and m_y are scaling, to convert to pixels
  - m_x = #of pixels in x direction / size of CCD array in x direction
  - m_y = #of pixels in y direction / size of CCD array in y direction
- p_x and p_y are principal point offset


y
y
Y
X
x
x
Optical axis, Z
Principal
Point
Offset

Perspective Transform

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x & 0 \\ 0 & m_y f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- Perspective Camera Model for the case when camera axes and world axes are aligned



Perspective Transform

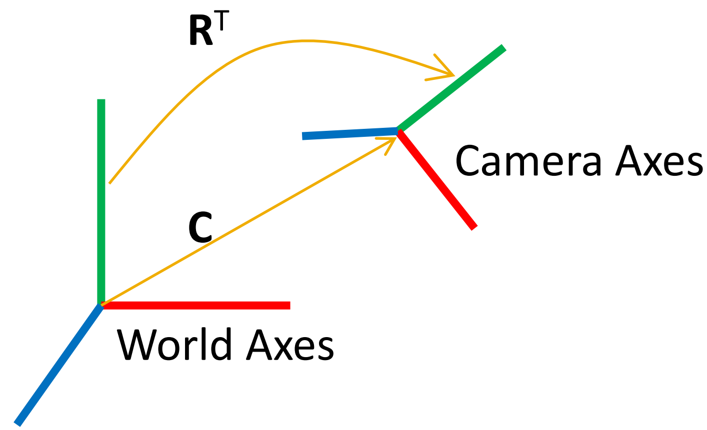
- This is for the case when the camera's optical axis is aligned with the world z-axis
  - Or: it relates camera frame to real image frame
- What if that is not the case?

Camera Model

- If the camera is moved **C** from the origin, we should move the world point by -**C**
- Then the perspective transform equation will be applicable
- Same holds for rotations

Perspective Transform

- In general, the camera center is at a rotation of R^T, followed by a translation of C from the world origin
- Then



$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x & 0 \\ 0 & m_y f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \left(\begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -C_X \\ 0 & 1 & 0 & -C_Y \\ 0 & 0 & 1 & -C_Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \right)$$

Perspective Transform

- Commonly used form for canonical view

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x \\ 0 & m_y f & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$h\bar{\mathbf{x}} = \mathbf{K}\left[\mathbf{I}_{3\times 3} \middle| \mathbf{0}_{3\times 1}\right]\mathbf{X}$$

Perspective Camera Model

Canonical View

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x \\ 0 & m_y f & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

General View

$$\begin{bmatrix} hx \\ hy \\ h \end{bmatrix} = \begin{bmatrix} m_x f & 0 & p_x \\ 0 & m_y f & p_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_X \\ r_{21} & r_{22} & r_{23} & t_Y \\ r_{31} & r_{32} & r_{33} & t_Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\mathbf{x} = \mathbf{K}[\mathbf{R} \,|\, \mathbf{T}]\mathbf{X}$$

$$\mathbf{x} = \mathbf{KR}\,[\mathbf{I} \,|\, -\mathbf{C}]\mathbf{X}$$

x – image point
X – world point
K – 3x3 matrix of internal camera parameters
[**R** | **T**] – 3x4 matrix of external camera parameters
  R – rotation needed to align camera to world axes
  T – Translation needed to bring camera to world origin
    T = -RC where **C** is the vector to camera center