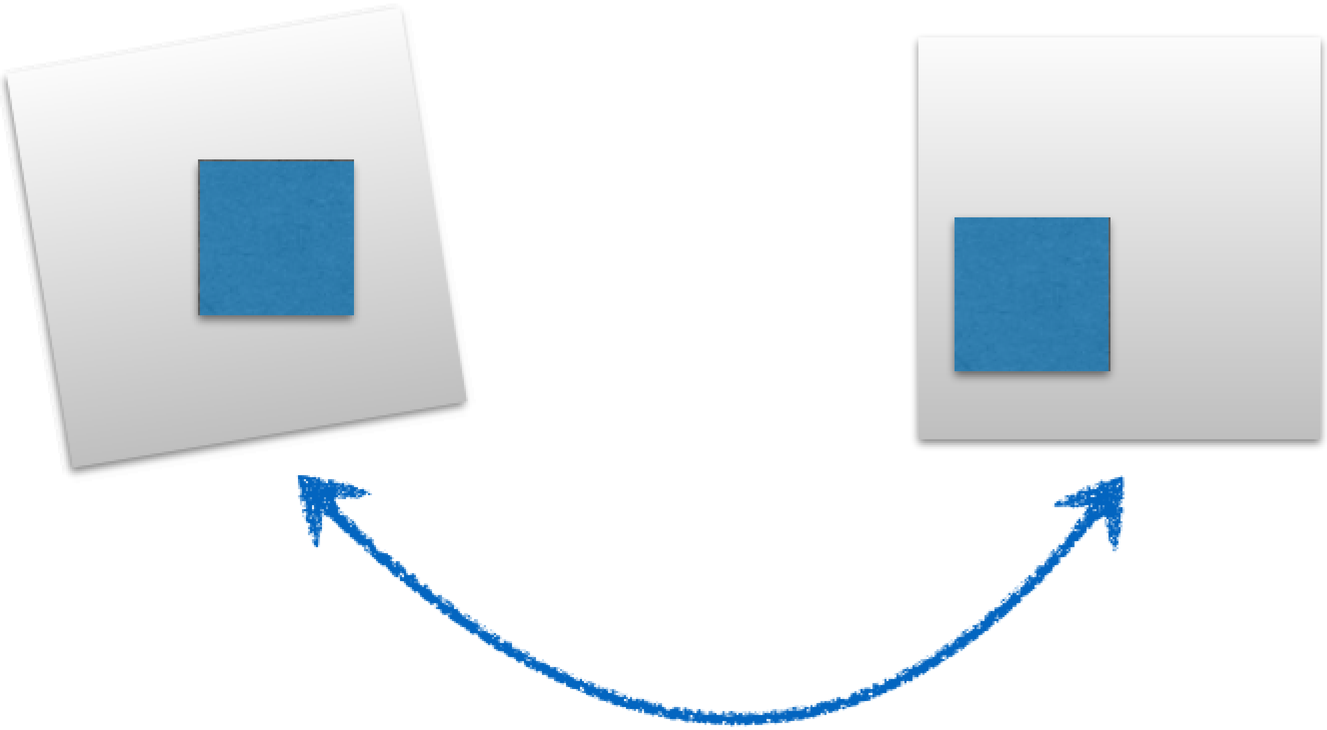
3D Data Processing

Camera model

Hyoseok Hwang

Review

- Homography



2D to 2D Transform

Today

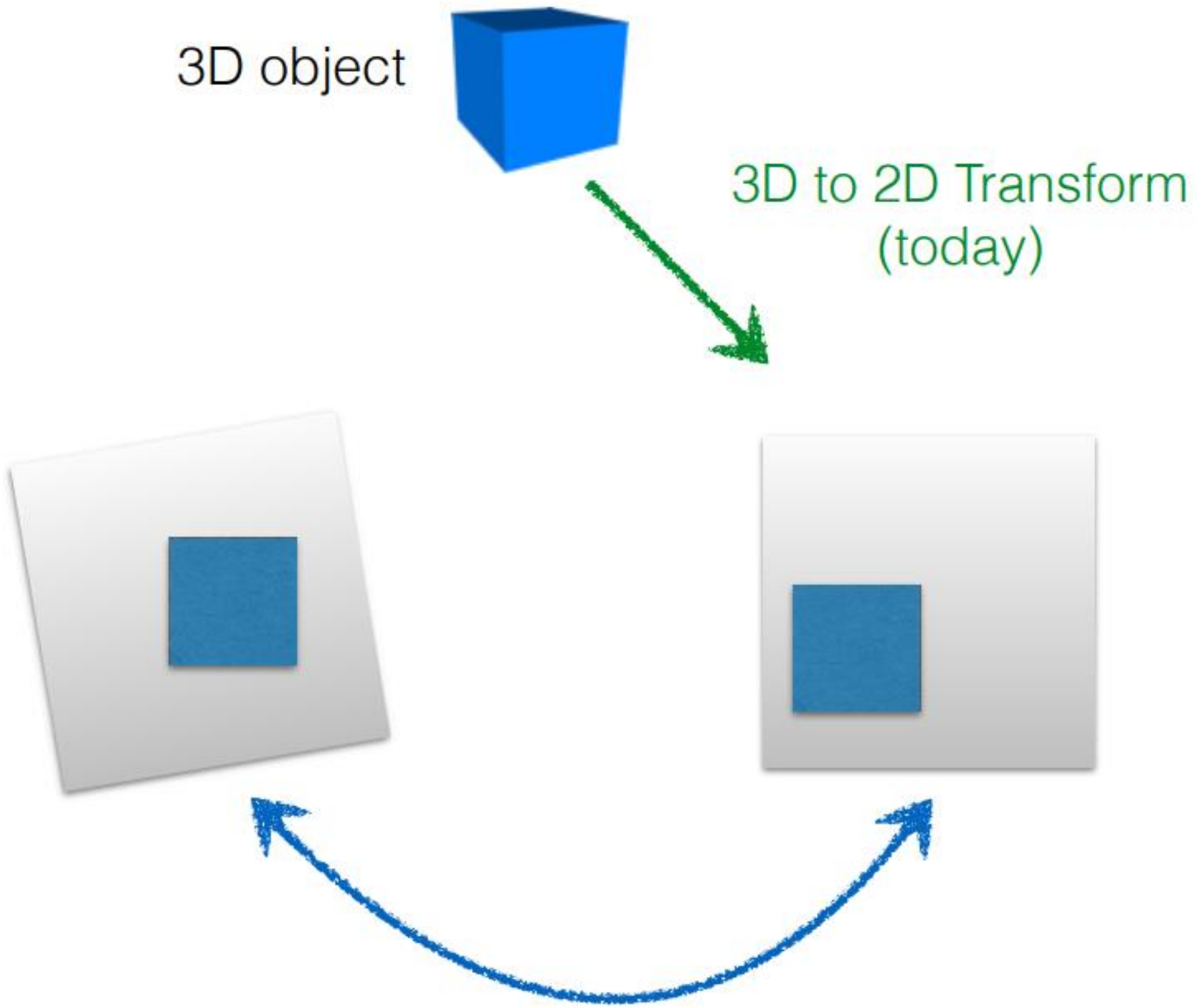
- Camera model

A camera is a mapping between

the **3D world**

and

a **2D image**



Primary concept of this class

- Where will a particular point in the world coordinate system be located in the image?

Projection matrix

- Projection Matrix

$$\boldsymbol{x} = \mathbf{PX}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \\ p_5 & p_6 & p_7 & p_8 \\ p_9 & p_{10} & p_{11} & p_{12} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

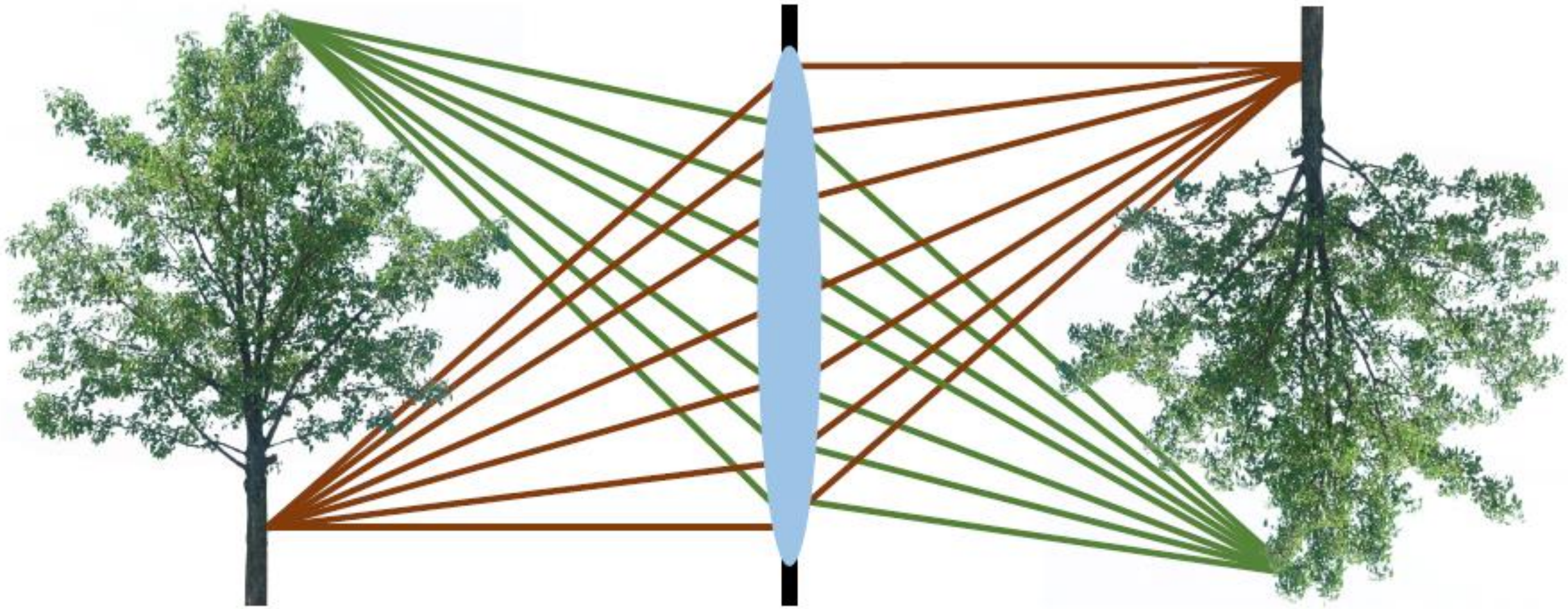
homogeneous image 3 x 1

Camera matrix 3 x 4

homogeneous world point 4 x 1

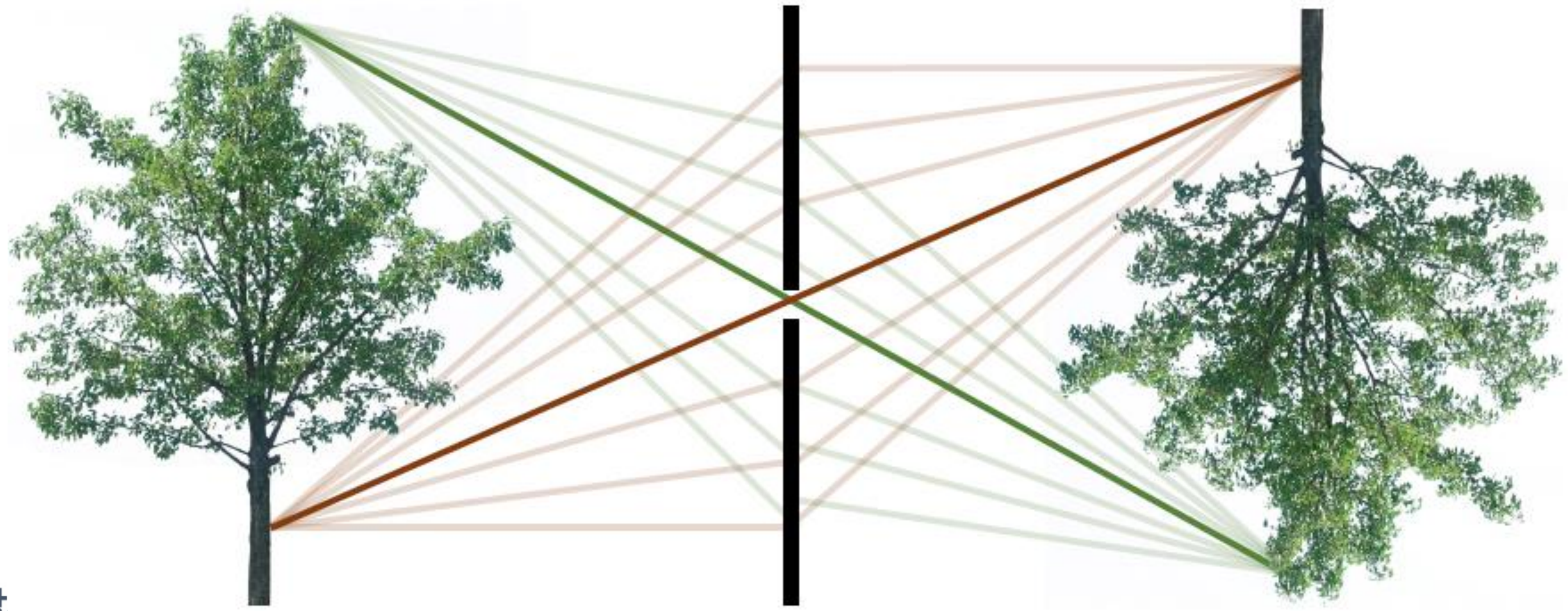
Review – pinhole camera model

- Most cameras use lenses, but we can simplify this to a pinhole model.



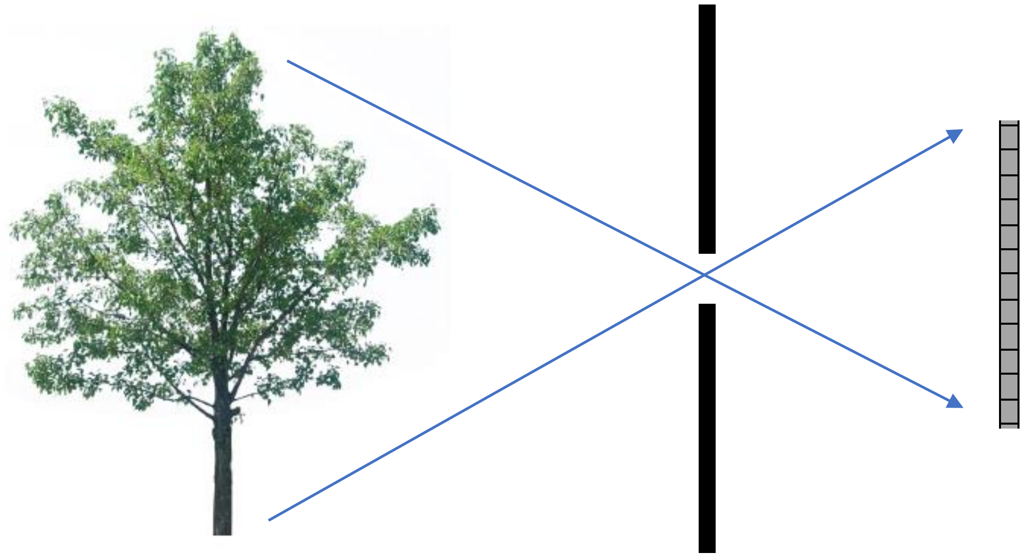
Review – pinhole camera model

- We can derive properties and descriptions that hold for both camera models if:
- We use only central rays.
- We assume the lens camera is in focus.



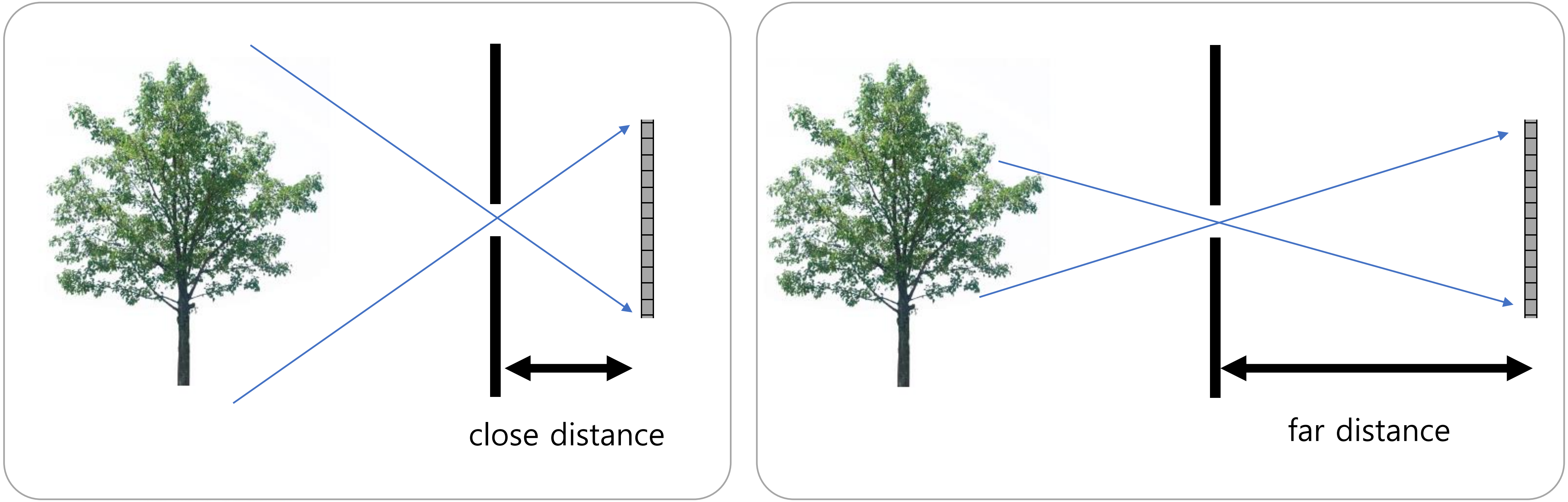
Concept of camera parameter

- Suppose that there is an image plane(e.g. an image sensor) that projects light through a pin hole. (depicted in 1D for convenience.)
- What are the factors that make the image projected on the sensor different?



Focal length

- How does the distance between the sensor and the pinhole affect the image?



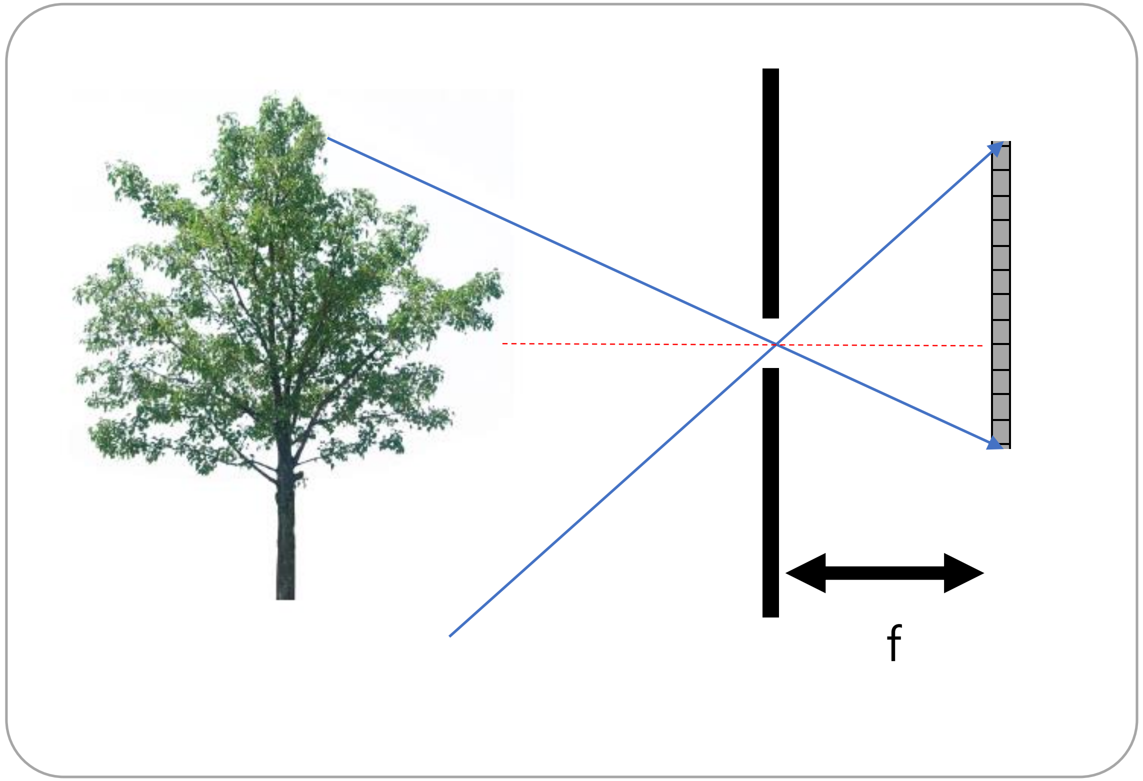
Focal length

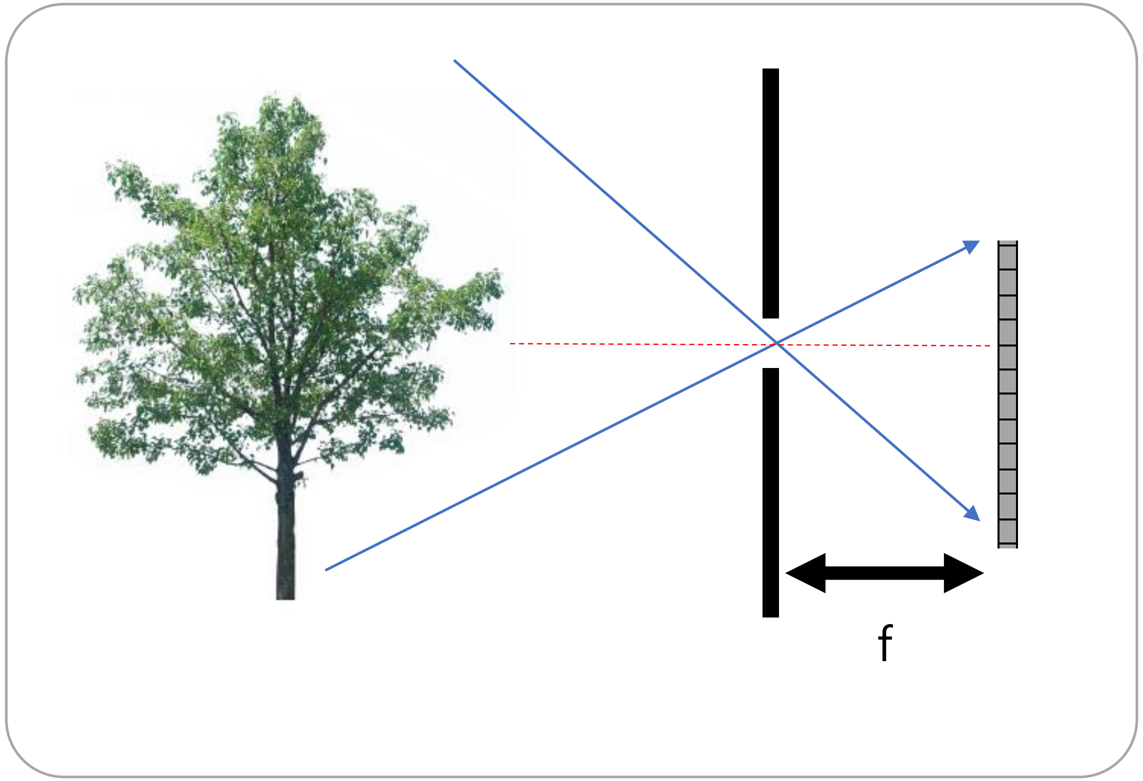
- Close distance
  - Wide field of view
  - Objects become smaller in size. (zoom-out)
- Far distance
  - Narrow field of view
  - Objects become larger in size (zoom-in)

- Focal length (f)
  - The distance from image plane and pinhole

Principal axis / points

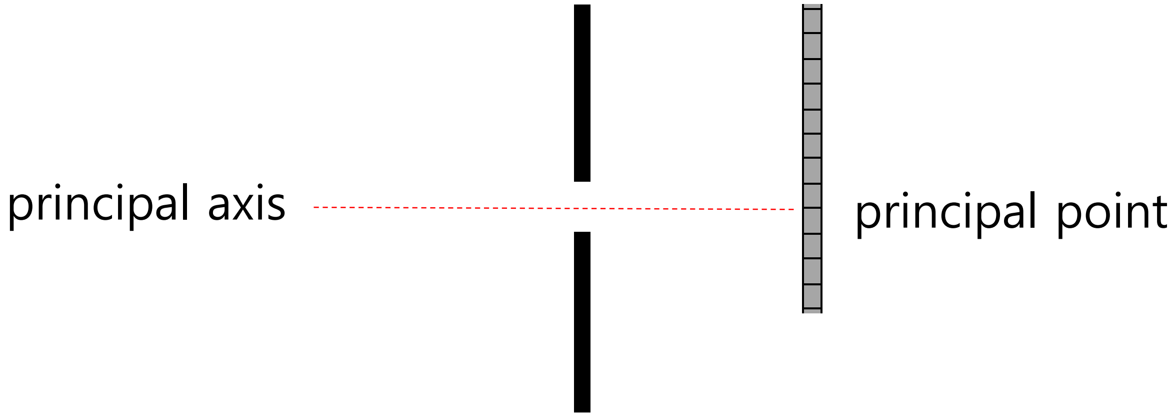
- How does the alignment between the sensor and the pinhole affect the image?





Principal axis / points

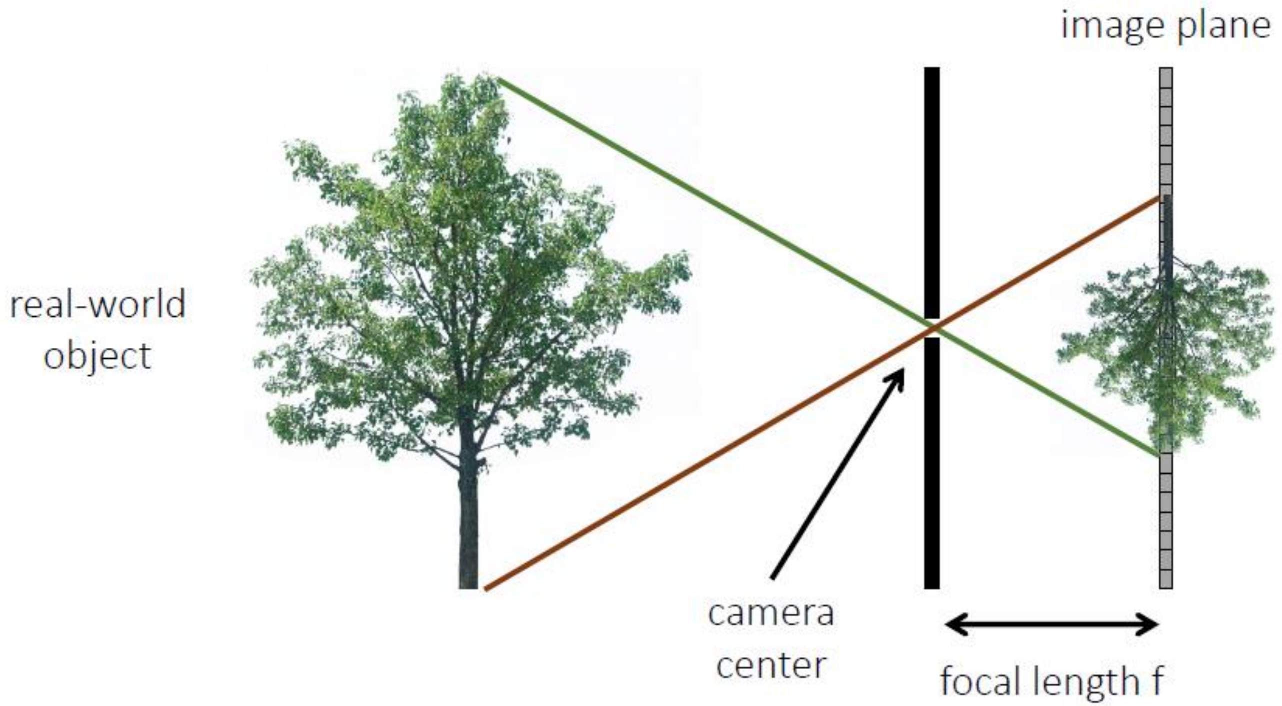
- **Principal axis:** line from the camera center perpendicular to the image plane
- **Principal point (p):** point where principal axis intersects the image plane (origin of normalized coordinate system)



- Now, what is the **camera center**?

Camera coordinate

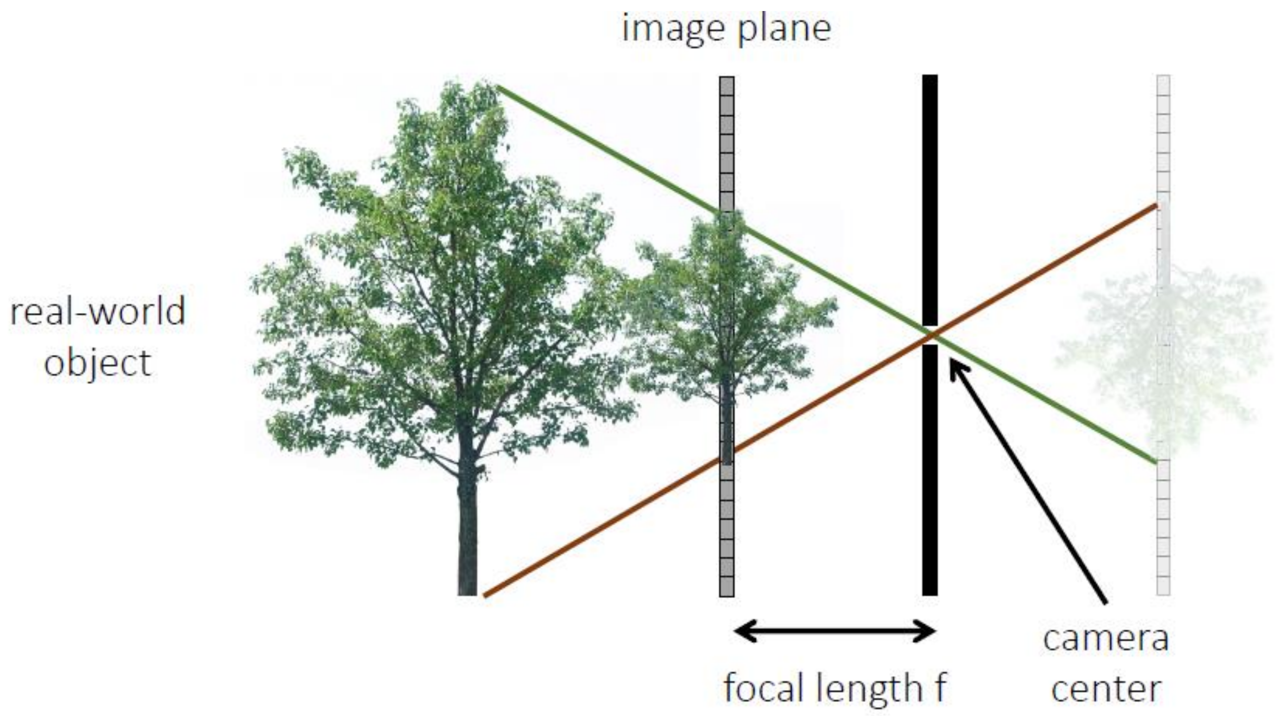
- The image plane on the other side of the pinhole can be moved point-symmetrically around the pinhole.



http://graphics.cs.cmu.edu/courses/15-463

Camera coordinate

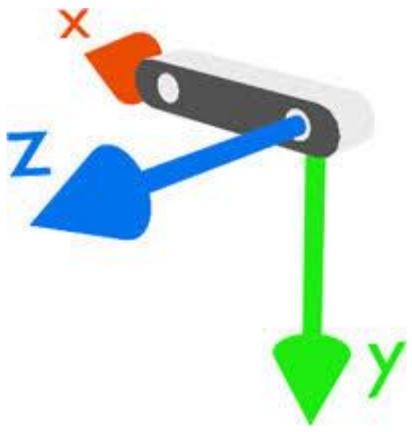
- The image plane on the other side of the pinhole can be moved point-symmetrically around the pinhole.



http://graphics.cs.cmu.edu/courses/15-463

Camera coordinate

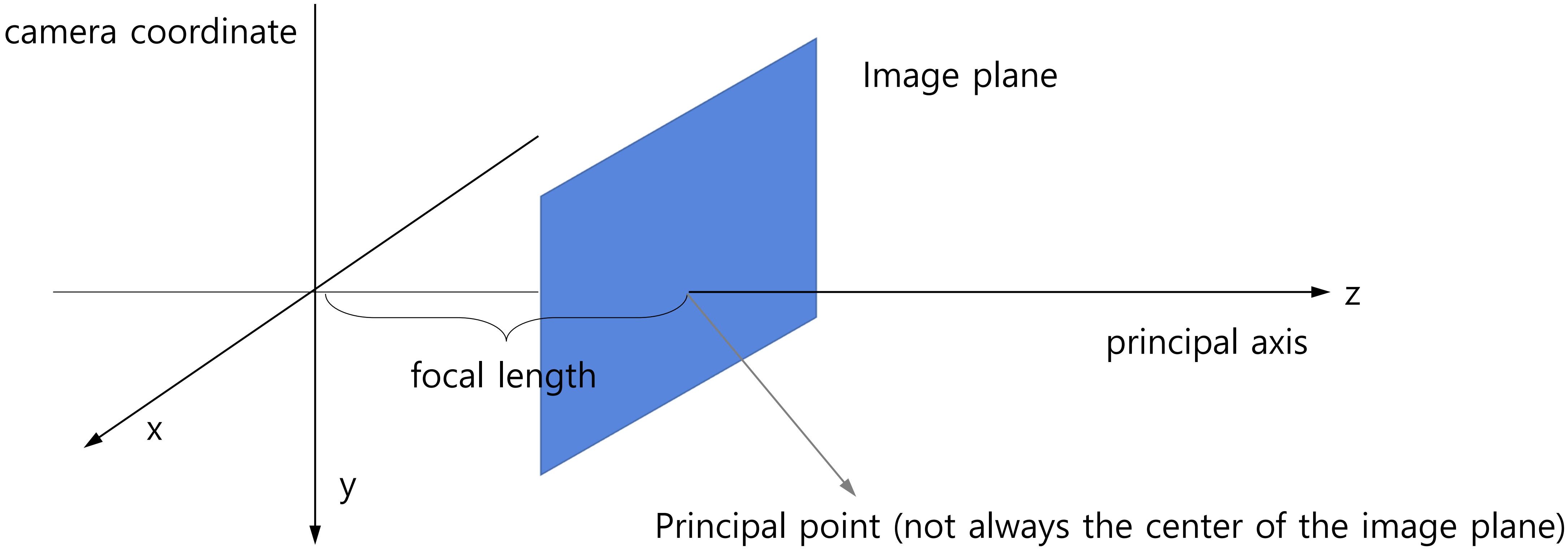
- Camera coordinate (3D)
  - Origin: pinhole
  - Axis:
    - x, y axes are same with those of the image plane
    - z axis: principal axis



- By virtue of the camera coordinate
  - We put the scene and the image plane on the same side.
  - The object on image plane is not flip upside down.

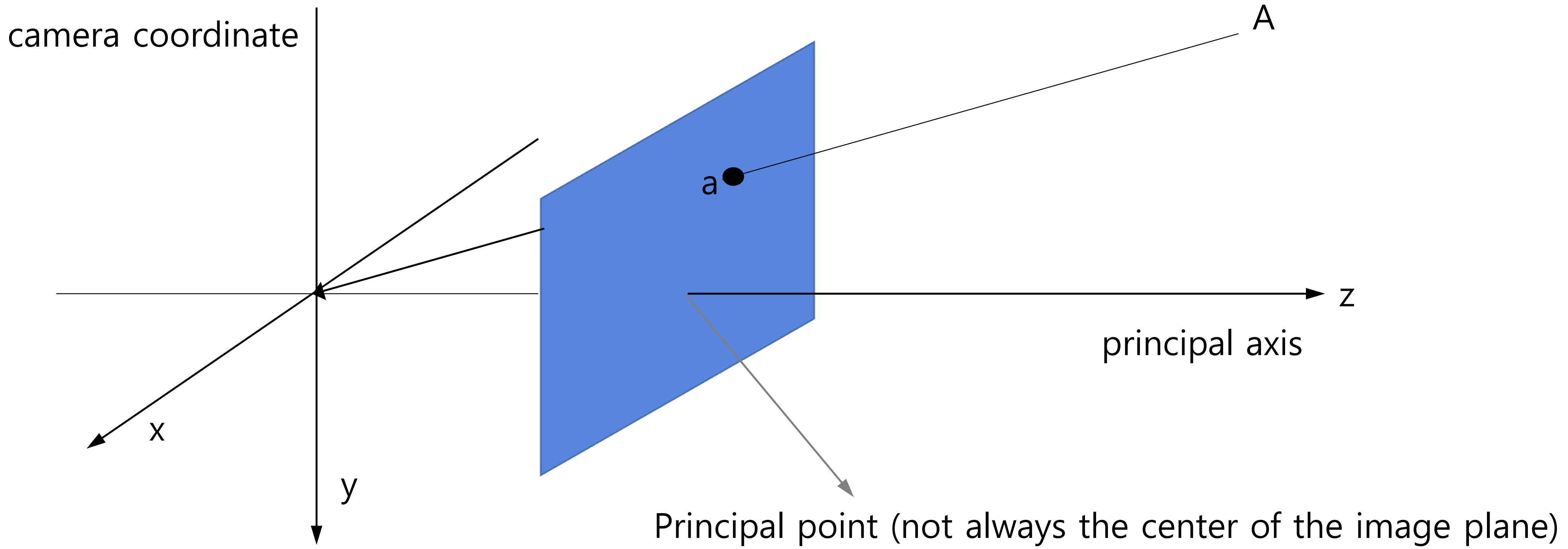
Camera coordinate

- Camera coordinate (3D)



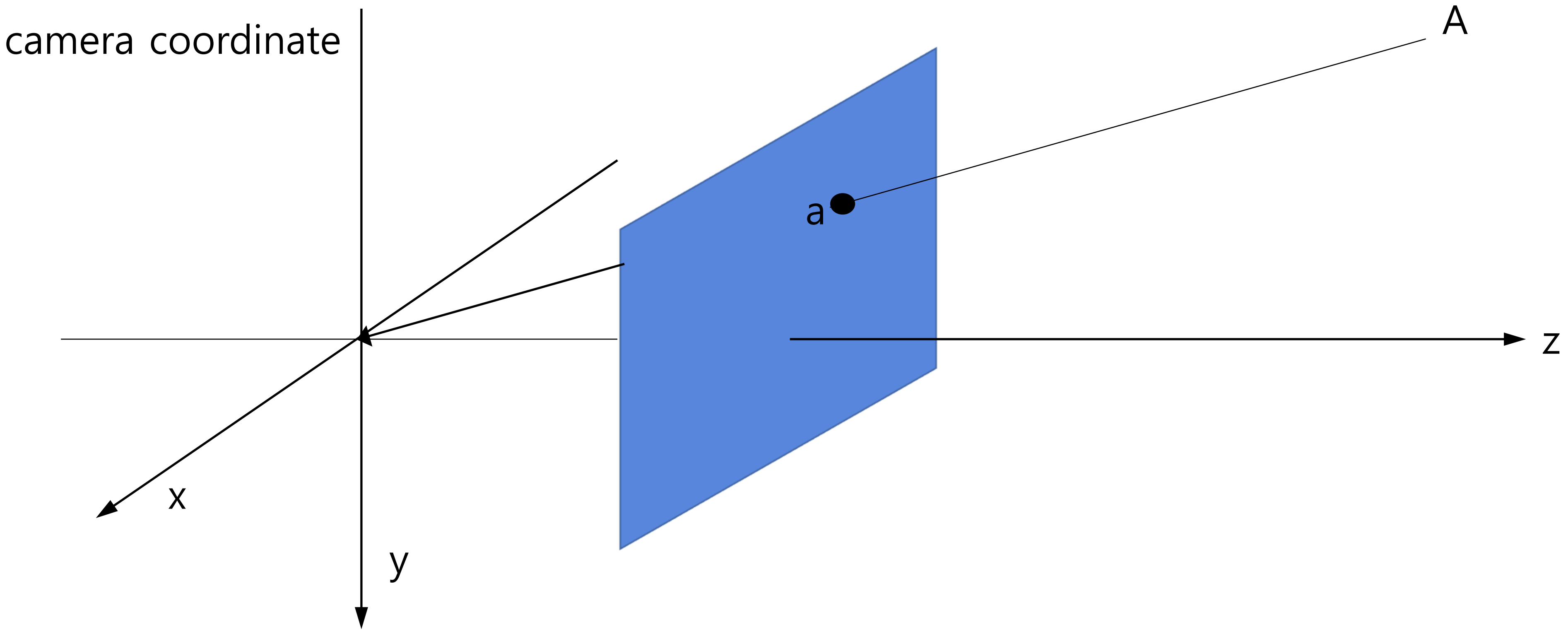
Camera coordinate

- When A in the world coordinate is projected to a on the image plane
- How can we know the position of “a” on the image plane?



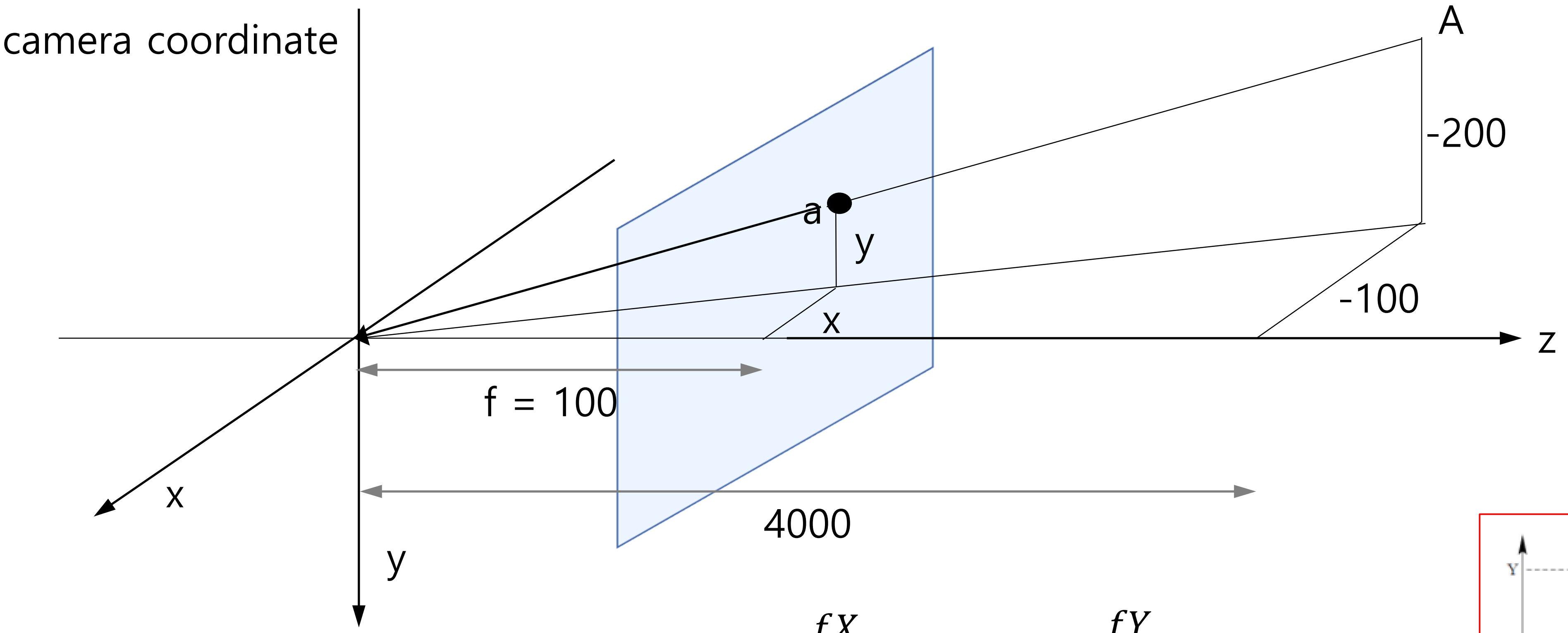
Camera coordinate

- The location of A in camera coordinate is (-100, -100, 4000)

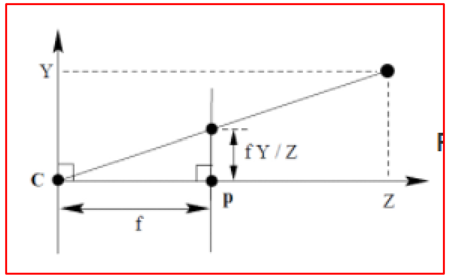


Camera coordinate

- The location of A in camera coordinate is (-100, -200, 4000)
- The location of a in the camera coordinate is (x, y, 100)



$$x = \frac{fX}{Z} = -2.5 \quad y = \frac{fY}{Z} = -5.0$$



Camera coordinate

- The a(x,y) is not a location of image coordinate but a location of camera coordinate
- This means a(x,y) is somewhere on the image plane
- How to convert camera coordinate to image coordinate?

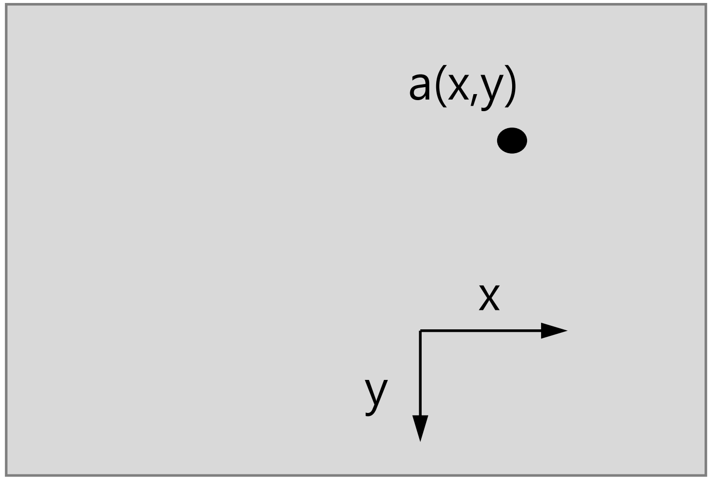


image plane
unit: mm

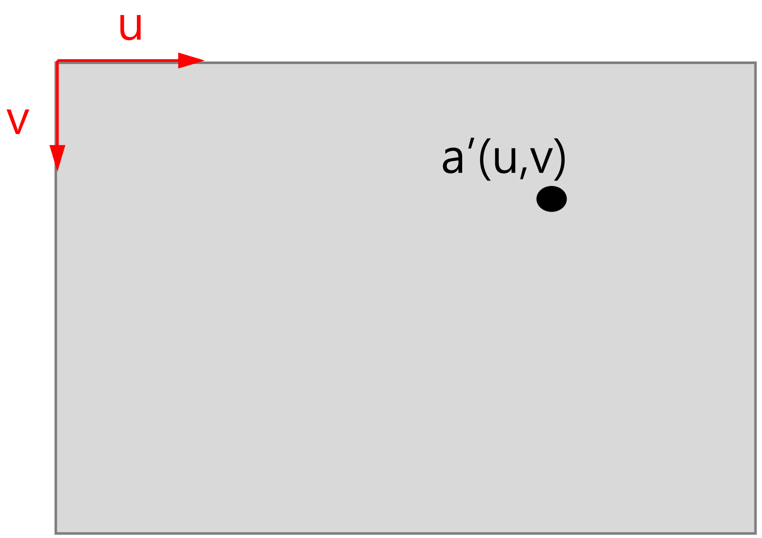
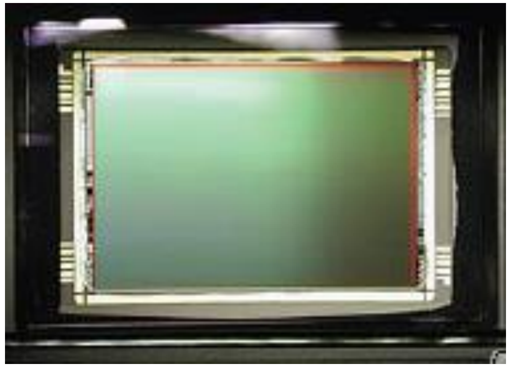


Image
unit: pixel

Camera coordinate

- Step 1: unit conversion
  - Convert mm to pixel
  - $(x, y) = (fX/Z, fY/Z)$
  - $u' = s_x x, v' = s_y y$



s_x:pixel per mm in x direction

s_y:pixel per mm in y direction

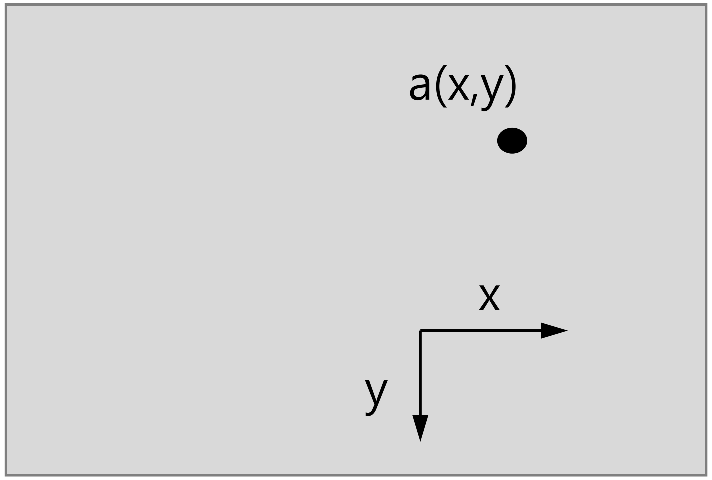


image plane
unit: mm

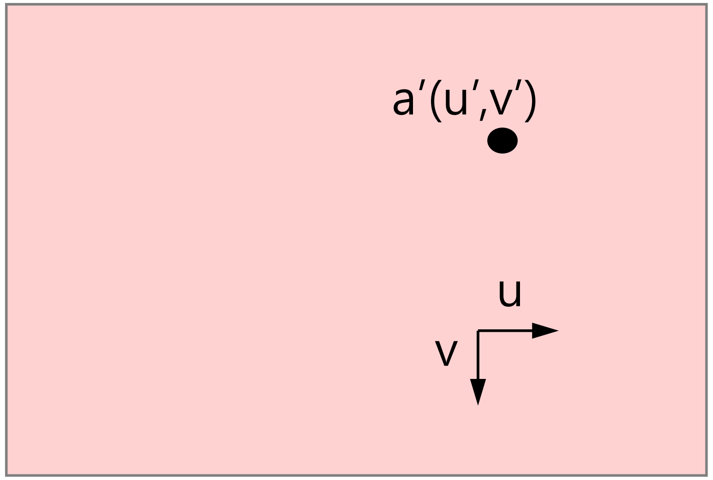


Image
unit: pixel

Camera coordinate

- Step 2: translation
  - Add $p(p_x, p_y)$ to converted a'
  - Then, $u = u' + c_x, v = v' + c_y$

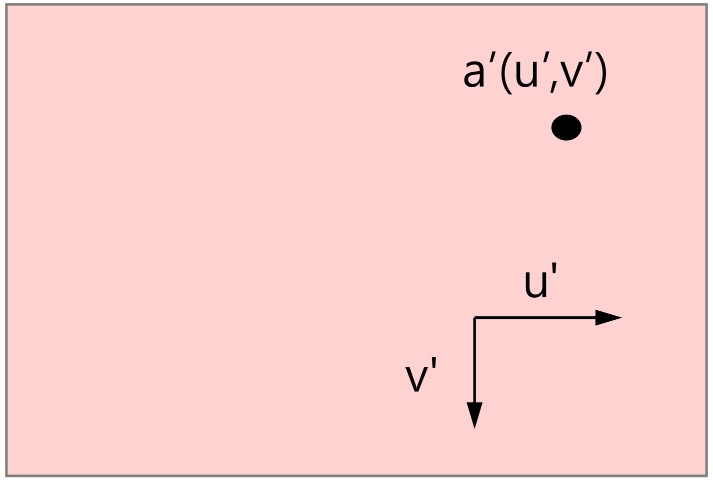


Image
unit: pixel

Origin of pixel coordinate
or image coordinate

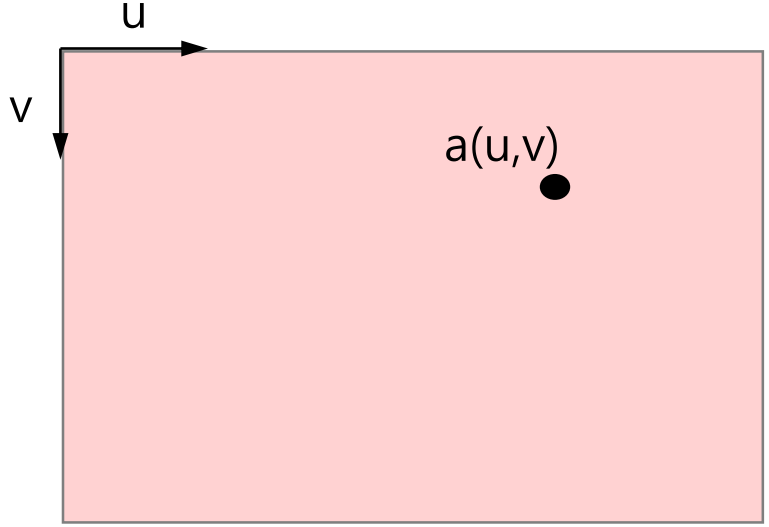


Image
unit: pixel

Intrinsic parameter

- Represented by 3x3 Matrix

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & \beta & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} s_x f & \beta & c_x \\ 0 & s_y f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} f_x & \beta & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

Position on the image coordinate is (u, v)
Because, (u, v) = $(u, v, 1) = (uk, vk, k)$

β is skew parameter, but we can set this value to 0

Intrinsic parameter

- Represented by 3x3 Matrix

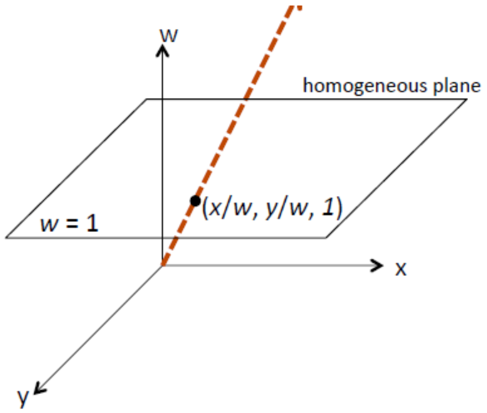
$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

- Therefore, four parameters are important
  - f_x : focal length (including pixel size) in x axis
  - f_y : focal length (including pixel size) in x axis
  - c_x : x value of principal point on the image coordinate
  - c_y : y value of principal point on the image coordinate

Homogeneous coordinates

- Represent a 2D point (x,y) by a 3D point (x′,y′,z′) by adding a "fictitious" third coordinate
- By convention, we specify that given (x′,y′,z′) we can recover the 2D point (x,y) as

$$x = \frac{x'}{z'} \qquad y = \frac{y'}{z'}$$



- Note: (x,y) = (x,y,1) = (2x, 2y, 2) = (kx, ky, k)
  for any nonzero k (can be negative as well as positive)

Projection

- A point in 3D coordinate A(X, Y, Z, 1) is projected to a(u, v) on the image according to the following:

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} s_x f & \beta & c_x \\ 0 & s_y f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Intrinsic parameter

- Represented by 3x3 Matrix

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

- What $[X_c, Y_c, Z_c]^T$ means?
- What if we don't know the location for the camera coordinates?

  → convert to camera coordinates

Projection matrix decomposition

- The projection matrix

$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = \begin{bmatrix} s_x f & \beta & c_x \\ 0 & s_y f & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

is represented as

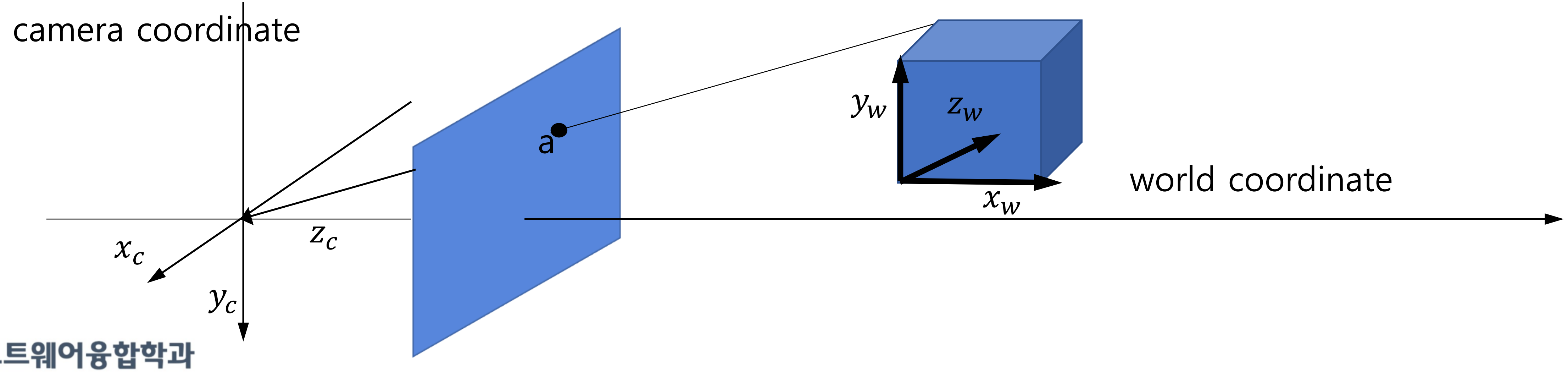
$$\begin{bmatrix} uk \\ vk \\ k \end{bmatrix} = K[R \mid t] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

, where **K is camera intrinsic parameter** and **R, t are extrinsic parameters** which represent rotation and translation between the camera coordinate and the world coordinate (or the object coordinate).

Extrinsic parameter

- World coordinate?
  - Up to convert the location to image coordinate, we have to set all coordinate values based on the camera coordinates.
  - In some cases, we can only know the position of a point on an object relative to the object(world) coordinate system.

A_w(0, 100, 100) in object coordinate
A_c(?, ?, ?) in the camera coordinate?

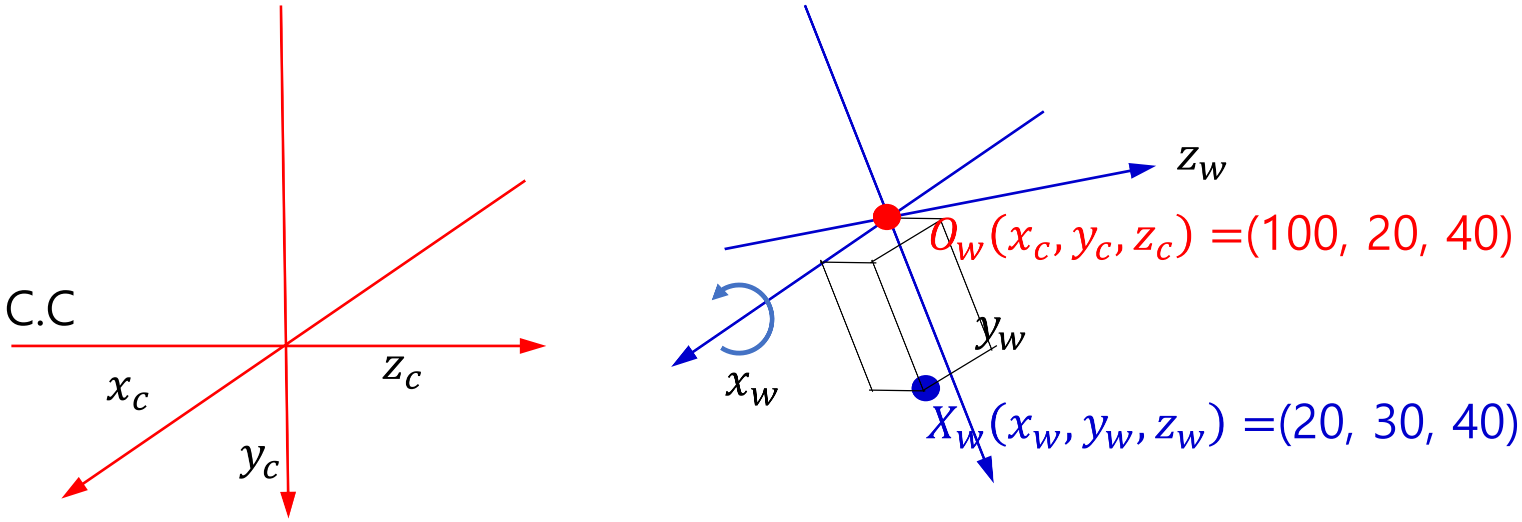


Extrinsic parameter

- World coordinate to camera coordinate
  - A point in the world coordinate is converted to the point int the camera coordinate by multiplying R, t
  - R: rotation matrix
  - t: translation matrix
- Example
  - When the origin of W.C. is (100, 20, 40) in the camera coordinate.
  - The W.C. is rotate 30deg in x axis
  - The position of $X_w(20, 30, 40)$
  - How can be this point located in the camera coordinate?

Extrinsic parameter

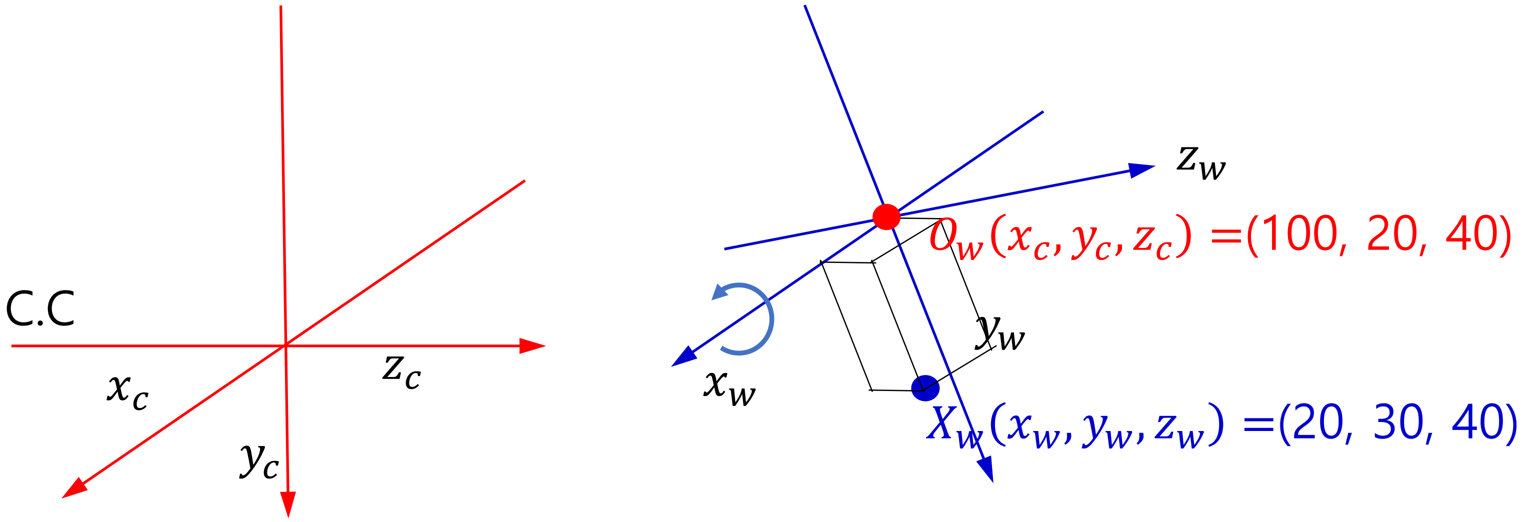
- Example
  - When the origin of W.C. is (100, 20, 40) in the camera coordinate.
  - The W.C. is rotate 30deg in x axis
  - The position of X_w(20, 30, 40)
  - Where will be this point located in the camera coordinate?



Extrinsic parameter

- Solution
  - 1. Rotate W.C. to align with C.C (rotate 30 deg in x axis)

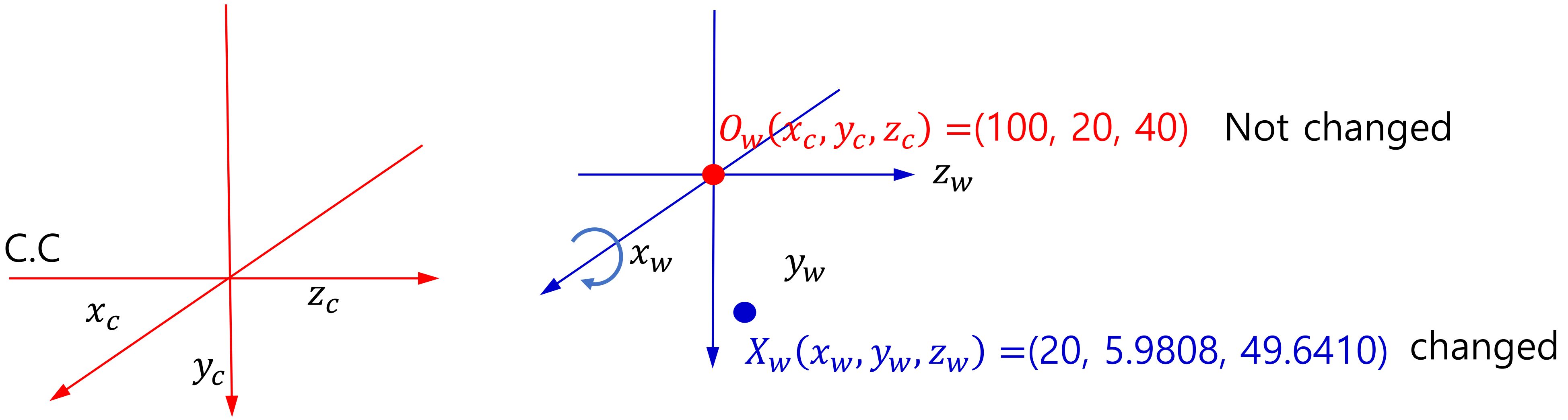
$$R = \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta &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



Extrinsic parameter

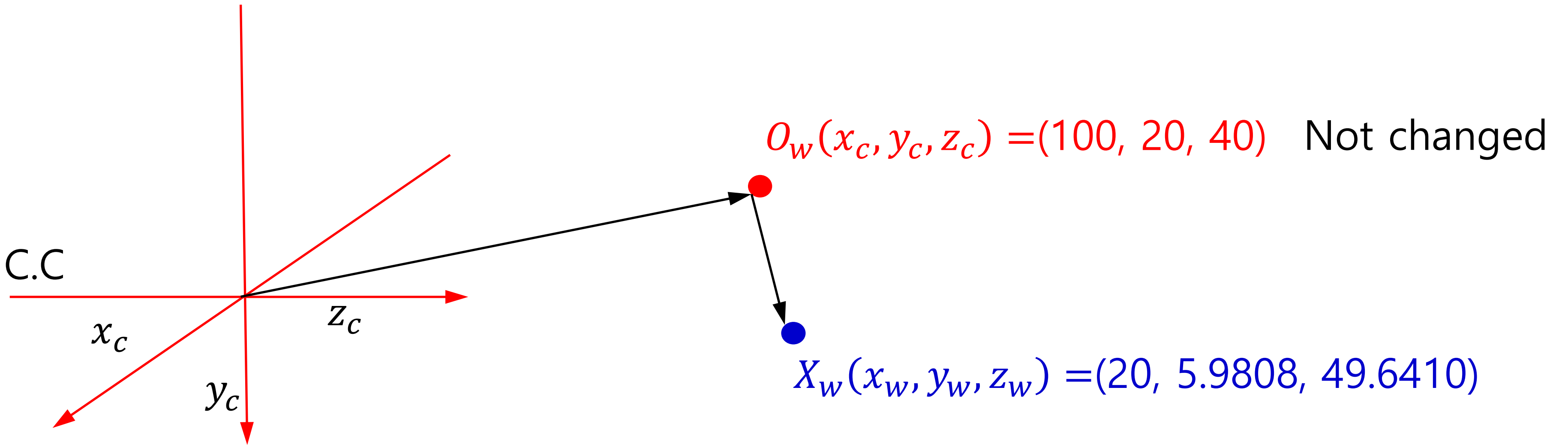
- Solution
  - 1. Rotate W.C. to align with C.C (rotate 30 deg in x axis)

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



Extrinsic parameter

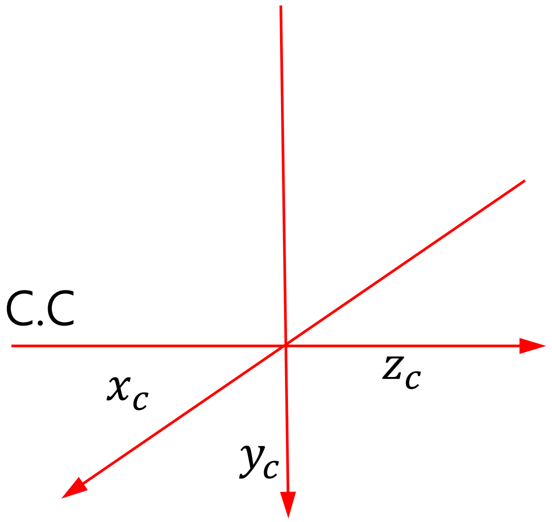
- Solution
  - 2. Translate The coordinate
  - $X_c = (O_w + X_w)$



Extrinsic parameter

- Solution
  - Briefly,

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + t \quad \Rightarrow \quad \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = [R \mid t] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$



$X_c(x_c, y_c, z_c)$ =(120, 25.9808, 89.6410)

Extrinsic parameter

- Simple code

```
import numpy as np
import cv2

R_vec = np.array([30 * np.pi / 180.0, 0, 0])
t_vec = np.array([[100],[20],[40]])

Xw = np.array([[20],[30],[40]])

R_mat = np.zeros((3,3))
cv2.Rodrigues(R_vec, R_mat)

print(np.dot(R_mat, Xw) + t_vec)
```

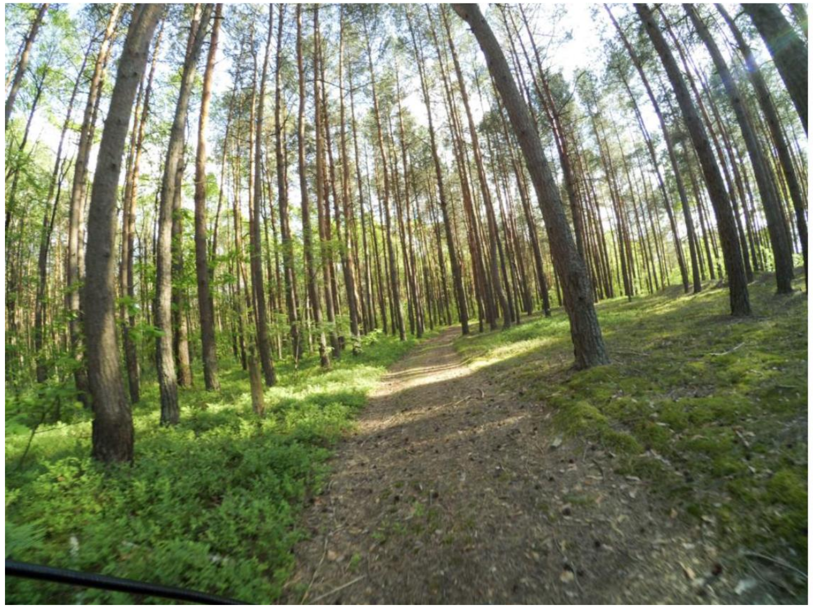
Extrinsic parameter

- When we already know the 3D position in the camera coordinate,
  - We set rotation matrix to identity matrix
  - We set translation vector to null vector

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix}$$

More Intrinsic parameter

- Radial distortion
  - We assume that the basic camera model follows pinhole model.
  - Unfortunately, cameras do use lens



distorted image

Undistorted image

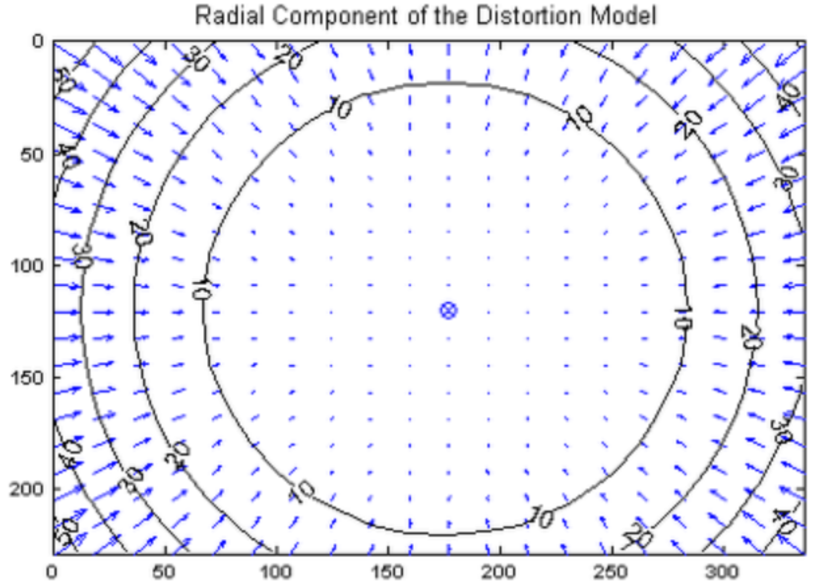
https://research.mapillary.com/publication/cvpr19d/

More Intrinsic parameter

- Zero at the optical center(principal point)
- Increase as moving toward the periphery
  - Coefficient estimation (k1, k2, k3)

$$x_u = x + \underbrace{(x - x_c) \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + \cdots)}_{\text{radial terms}}$$

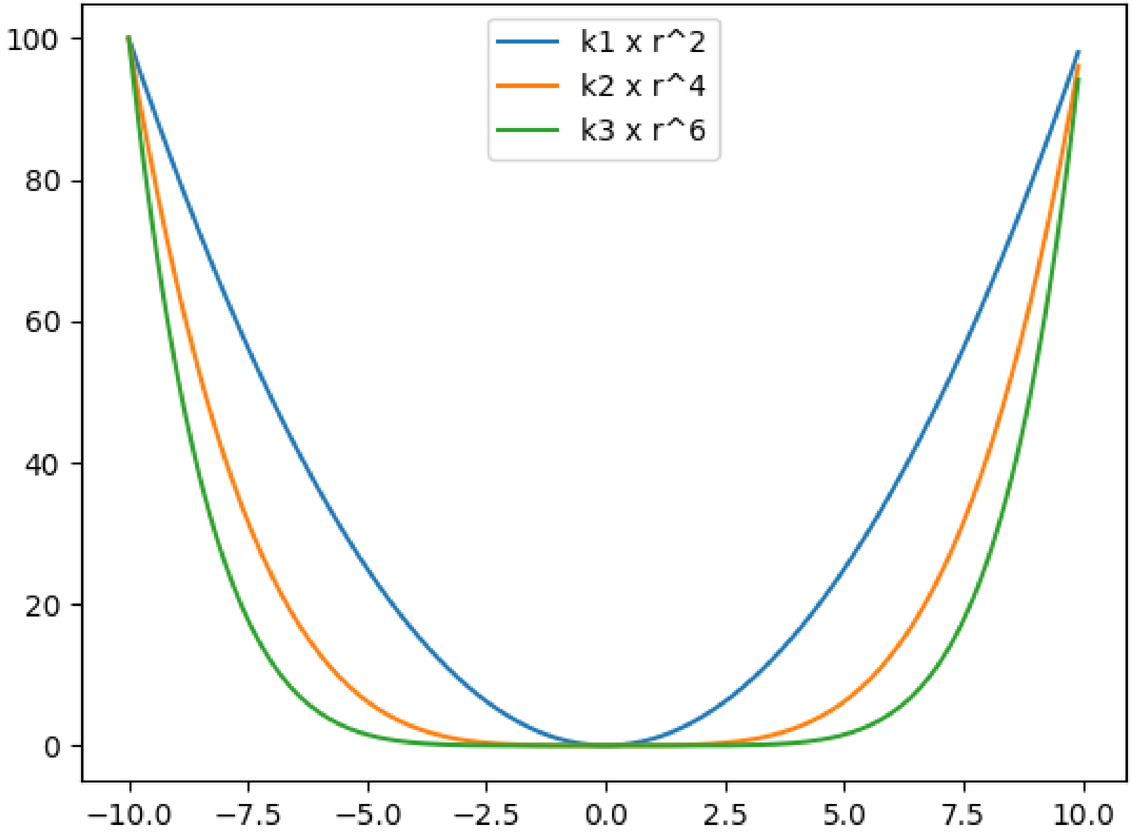
$$y_u = y + \underbrace{(y - y_c) \cdot (k_1 \cdot r^2 + k_2 \cdot r^4 + \cdots)}_{\text{radial terms}}$$



Pixel error	= [0.2688, 0.277]	
Focal Length	= (181.995, 164.699)	+/- [0.4468, 0.4092]
Principal Point	= (175.5, 119.5)	+/- [0, 0]
Skew	= 0	+/- 0
Radial coefficients	= (-0.289, 0.08213, -0.01014)	+/- [0.002255, 0.001728, 0.0003671]
Tangential coefficients	= (-0.0002611, -0.0002235)	+/- [0.0002153, 0.0001831]

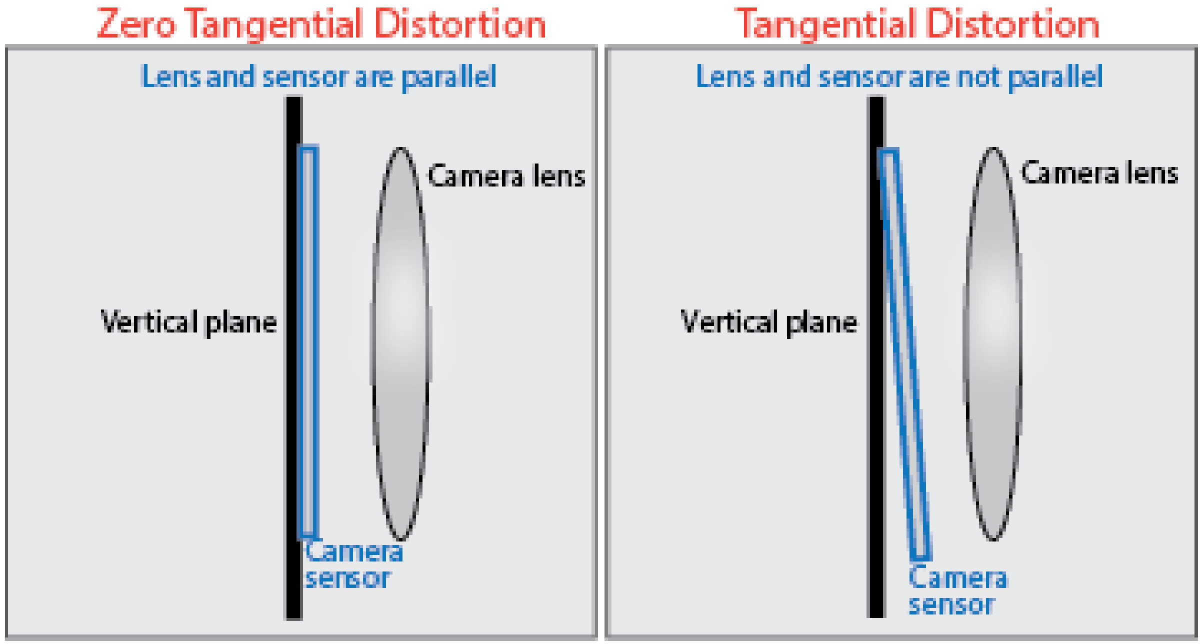
More Intrinsic parameter

- Zero at the optical center(principal point)



More Intrinsic parameter

- Tangential distortion
  - Tangential distortion occurs when the lens and image plane are not parallel.



Calibration

- Camera calibration is to estimate both
  - **Camera intrinsic parameter**
  - Extrinsic parameter (pose estimation)

- Why the calibration is necessary?
  - If we know the origin direction of ray,
  - We can back projection → 3D reconstruction

Thank you