Computer Vision

(Summer Semester 2020)

Lecture 5, Part 5

Cameras and Optics (Projective Geometry)

Pattern Recognition Lab

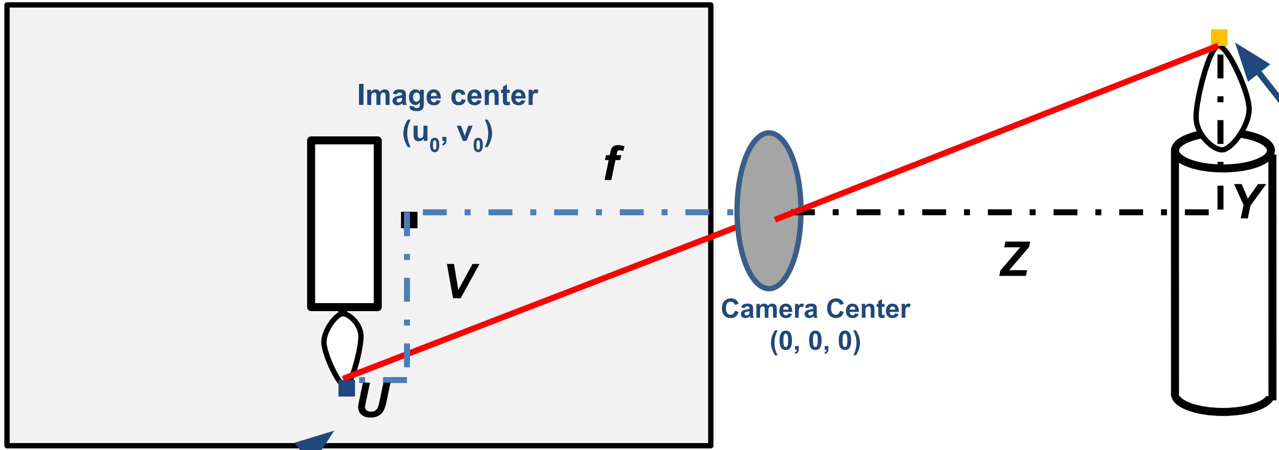
FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

Cameras and Optics (Projective Geometry)

- Pinhole Camera Model (part 4)
- Perspective Projection
- Intrinsic and Extrinsic Camera Parameters

- Note: The core of these slides stems from the class CSCI 1430: “Introduction to Computer Vision” by James Tompkin, Fall 2017, Brown University.

Projection: world coordinates → image coordinates



$$\mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$\mathbf{p} = \begin{bmatrix} U \\ V \end{bmatrix}$$

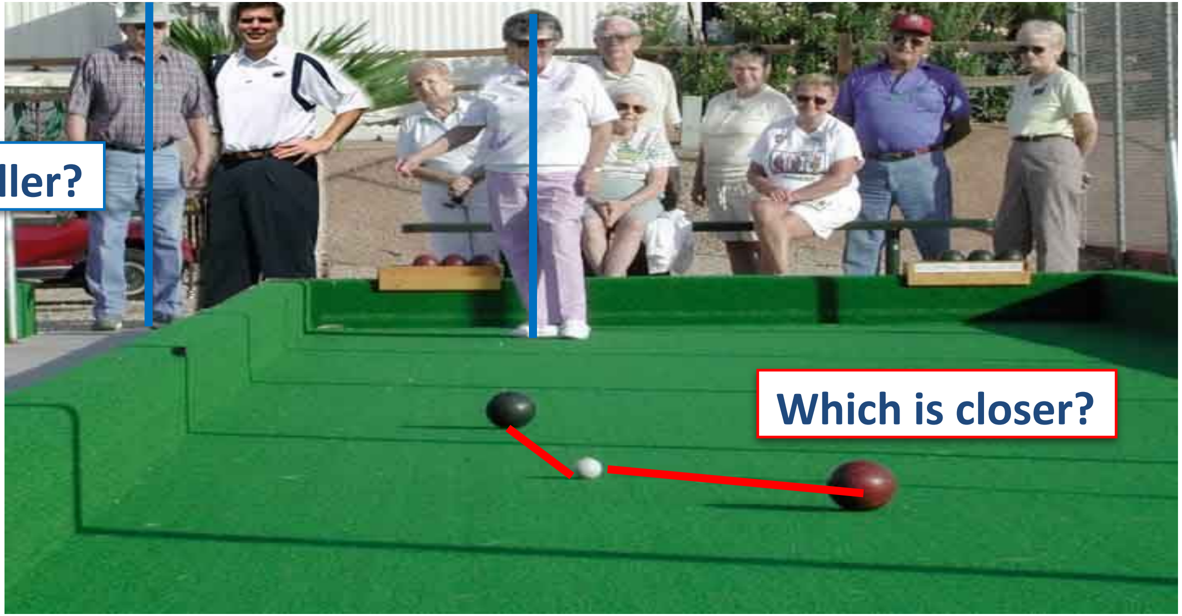
$$U = -X * \frac{f}{Z} \qquad V = -Y * \frac{f}{Z}$$

p = distance from image center

Intercept theorem:
https://en.wikipedia.org/wiki/Intercept_theorem

Projective Geometry

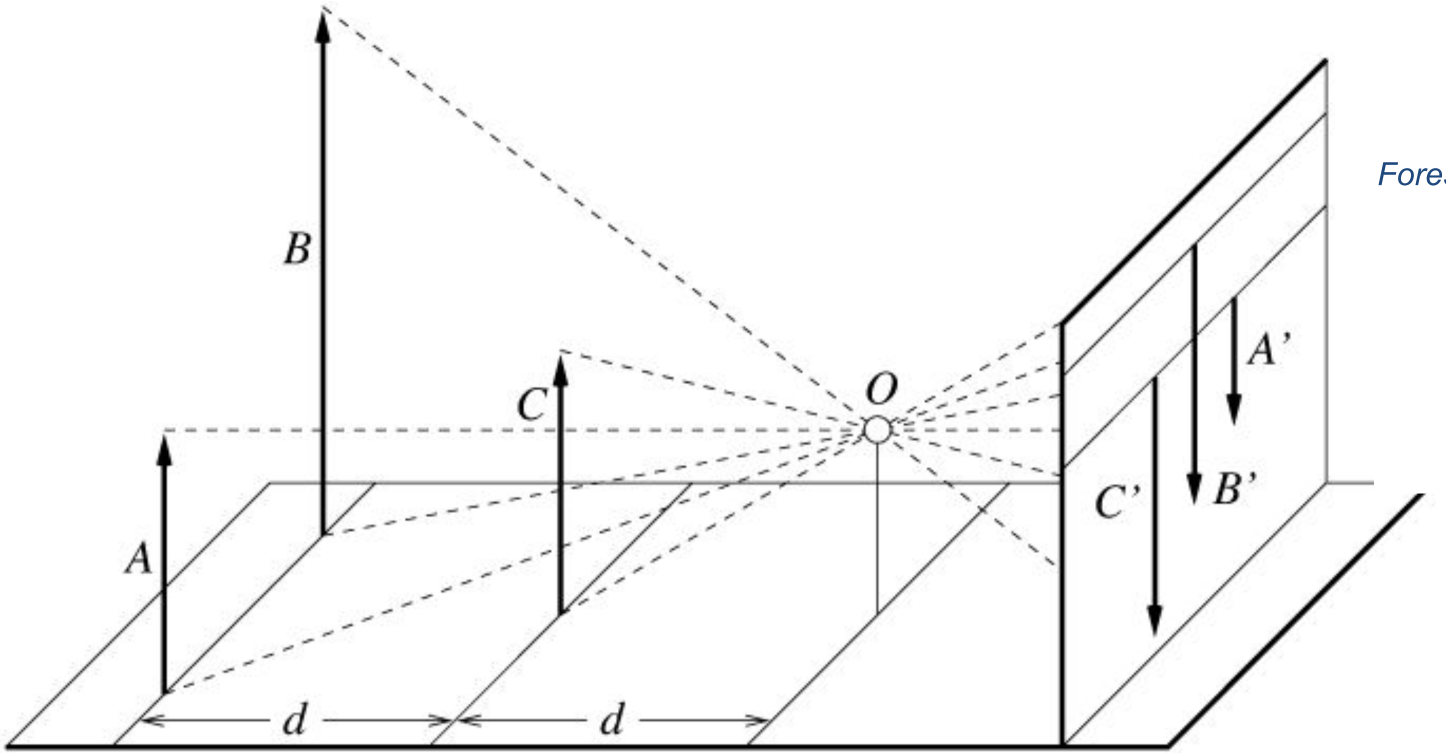
Length (and so area too) is lost.



Who is taller?

Which is closer?

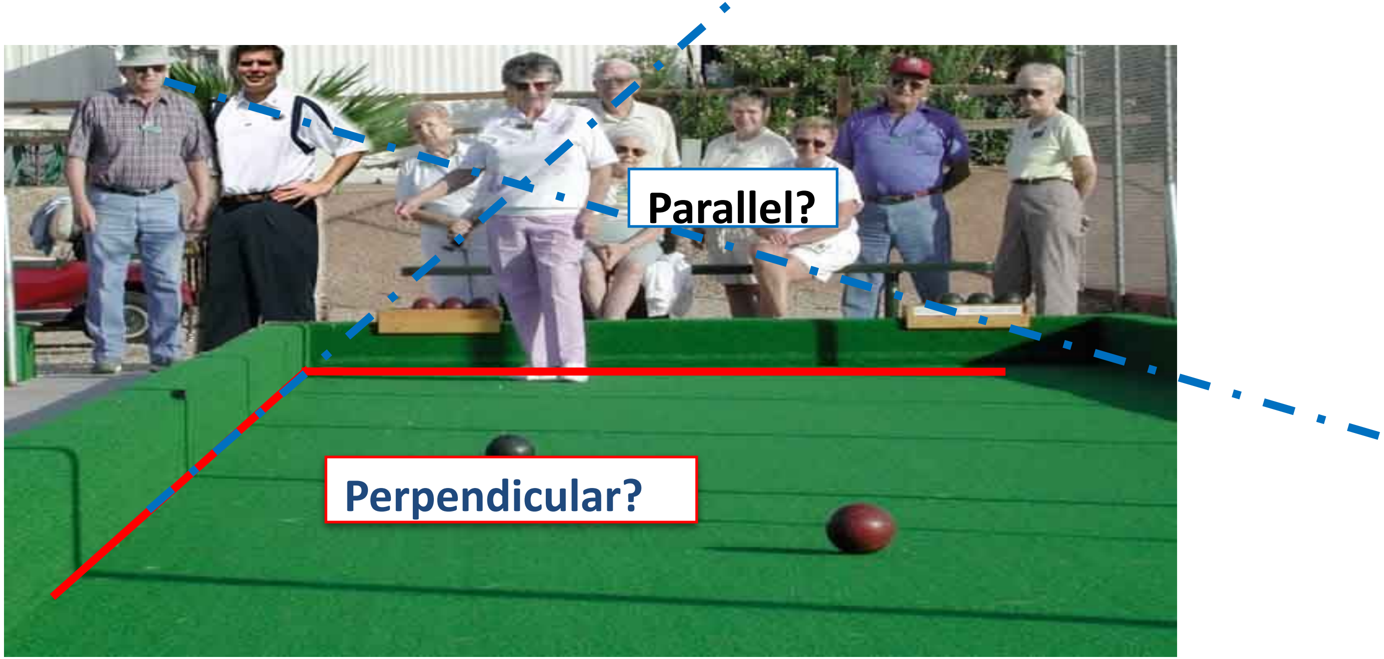
Length and area are not preserved



Foreshortening (ex. selfies)

Projective Geometry

Angles are lost.



Projective Geometry

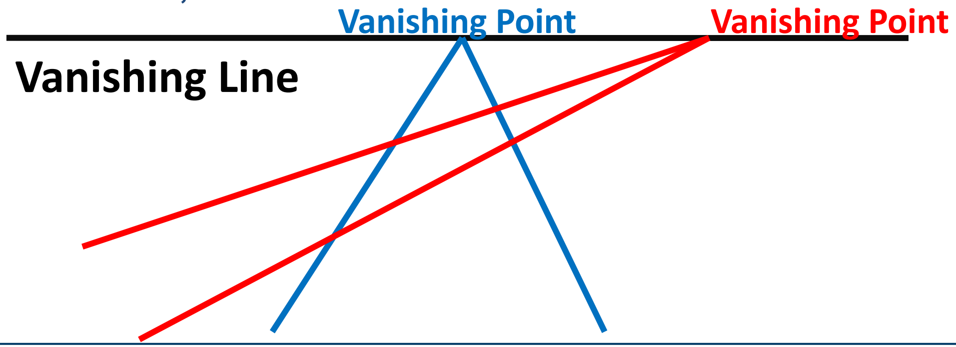
What is preserved?

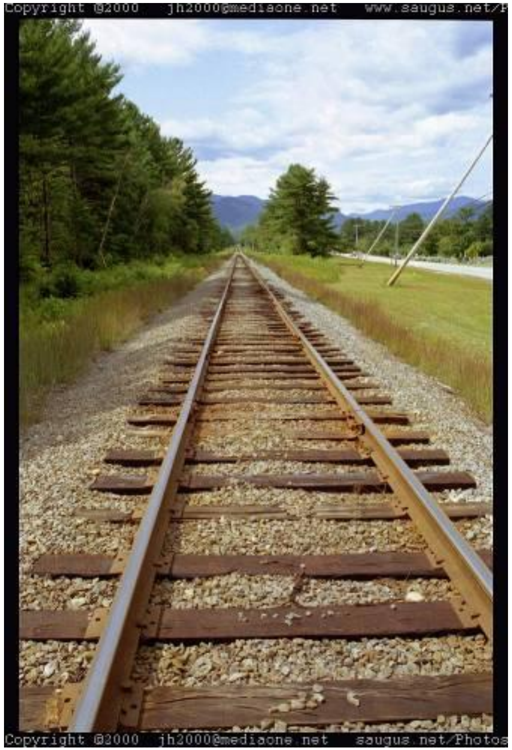
Straight lines are still straight.

Vanishing points and lines

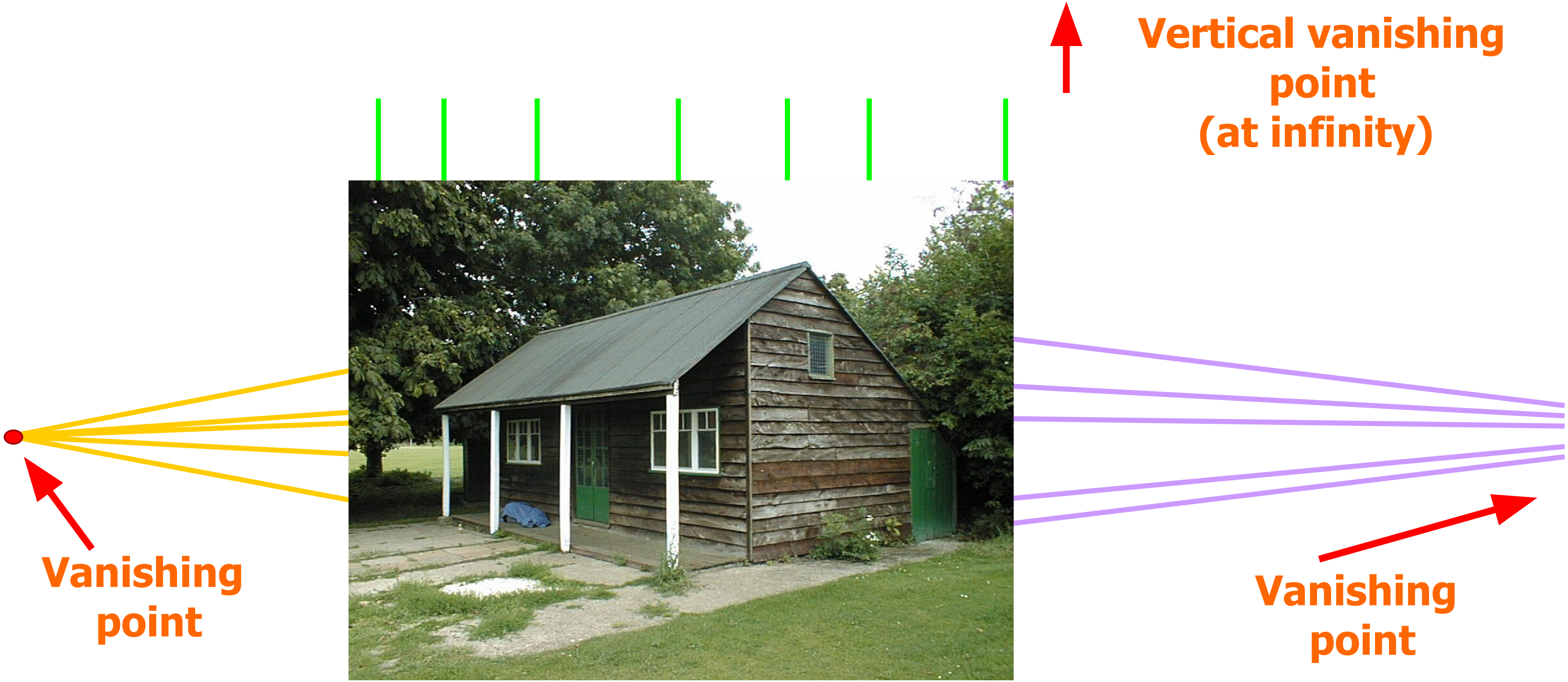
Parallel lines in the world intersect in the projected image at a "vanishing point".

Parallel lines on the same plane in the world converge to vanishing points on a "vanishing line".
E.G., the horizon.

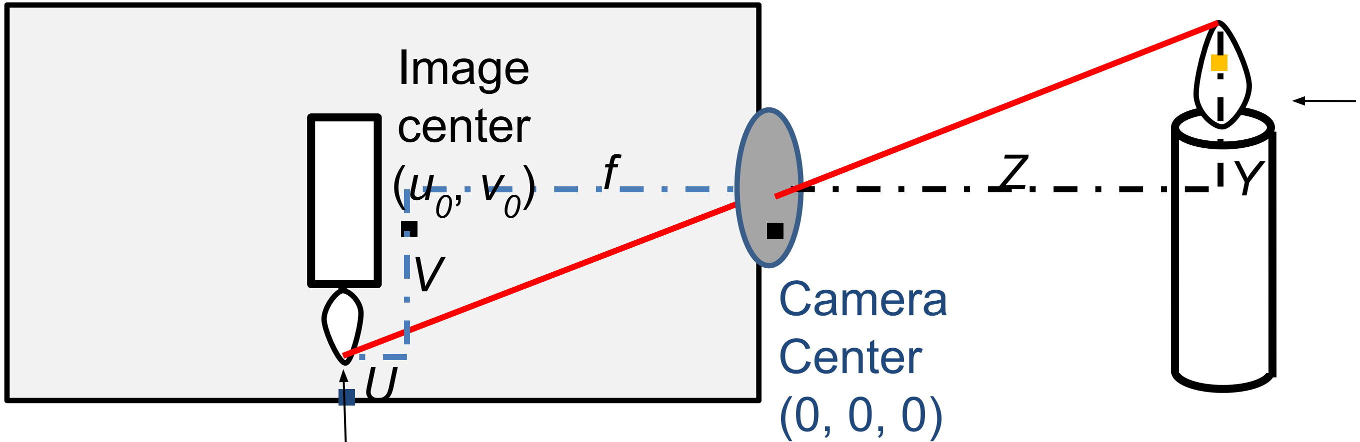




Vanishing points and lines



Projection: world coordinates - image coordinates



$$\mathbf{P} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

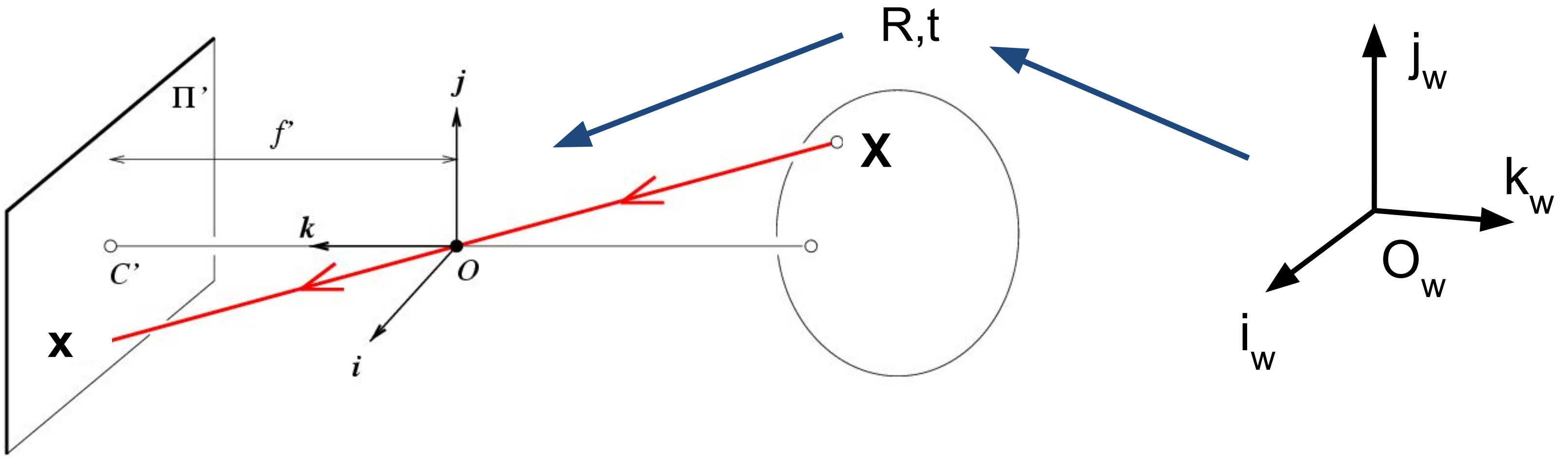
$$\mathbf{p} = \begin{bmatrix} U \\ V \end{bmatrix}$$

p = distance from image center

$$U = -X * \frac{f}{Z} \qquad V = -Y * \frac{f}{Z}$$

What is the effect if f and Z are equal?

Camera (projection) matrix



$$\mathbf{x} = \mathbf{K}\underbrace{\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}}_{\text{Extrinsic Matrix}}\mathbf{X}$$

x: Image Coordinates: (u,v,1)
K: Intrinsic Matrix (3x3)
R: Rotation (3x3)
t: Translation (3x1)
X: World Coordinates: (X,Y,Z,1)

Demo – Kyle Simek

"Dissecting the Camera Matrix"

Three-part blog series

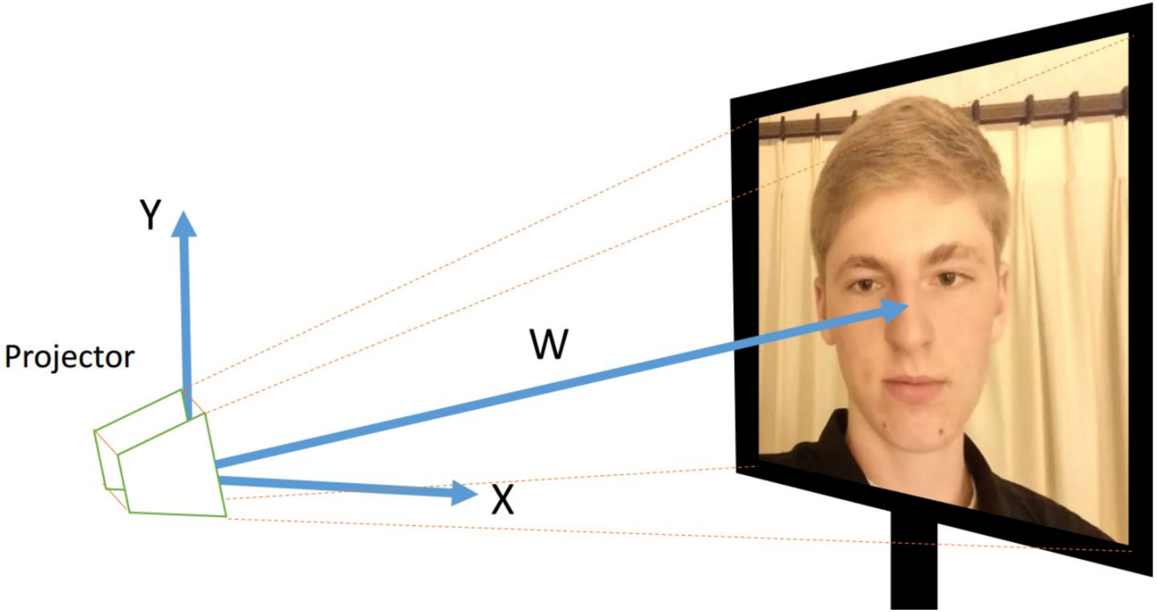
1. Dissecting the Camera Matrix, Part 1: Extrinsic/Intrinsic Decomposition ←
2. Dissecting the Camera Matrix, Part 2: The Extrinsic Matrix ←
3. Dissecting the Camera Matrix, Part 3: The Intrinsic Matrix ←

"Perspective toy"

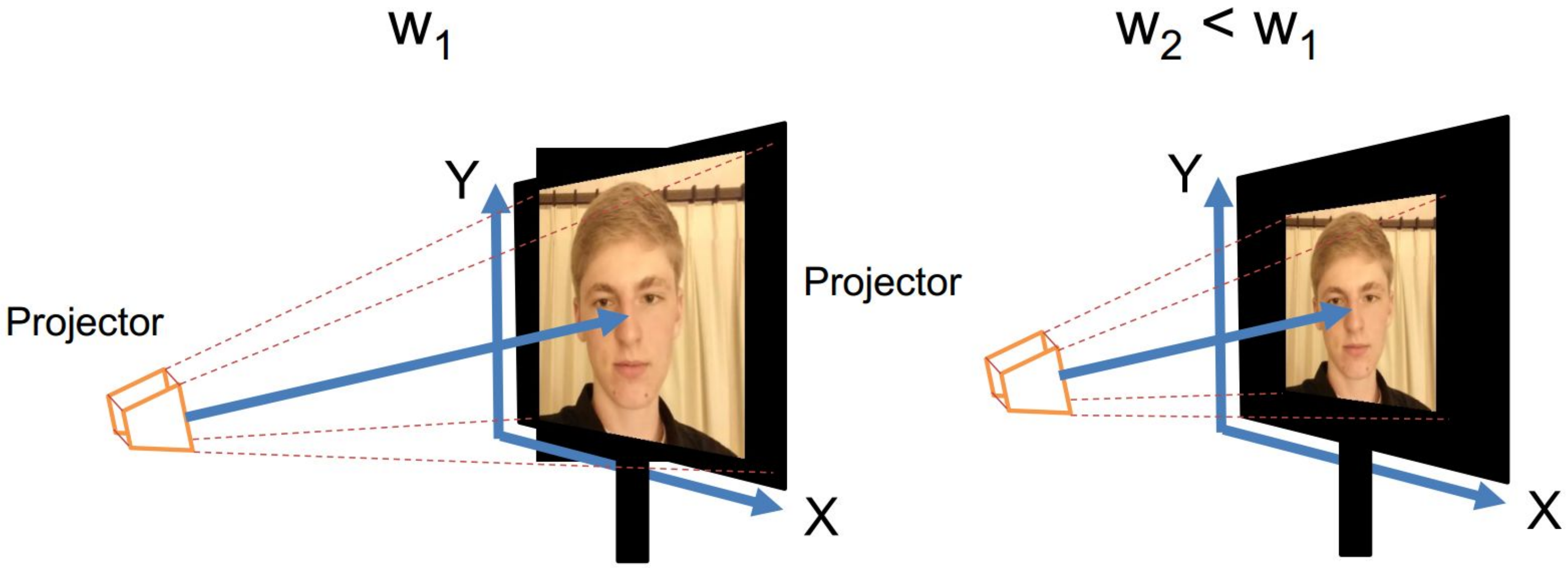
Perspective Camera Toy ←

Projective geometry

- 2D point in cartesian = (x,y) coordinates
- 2D point in projective = (x,y,w) coordinates



Varying w



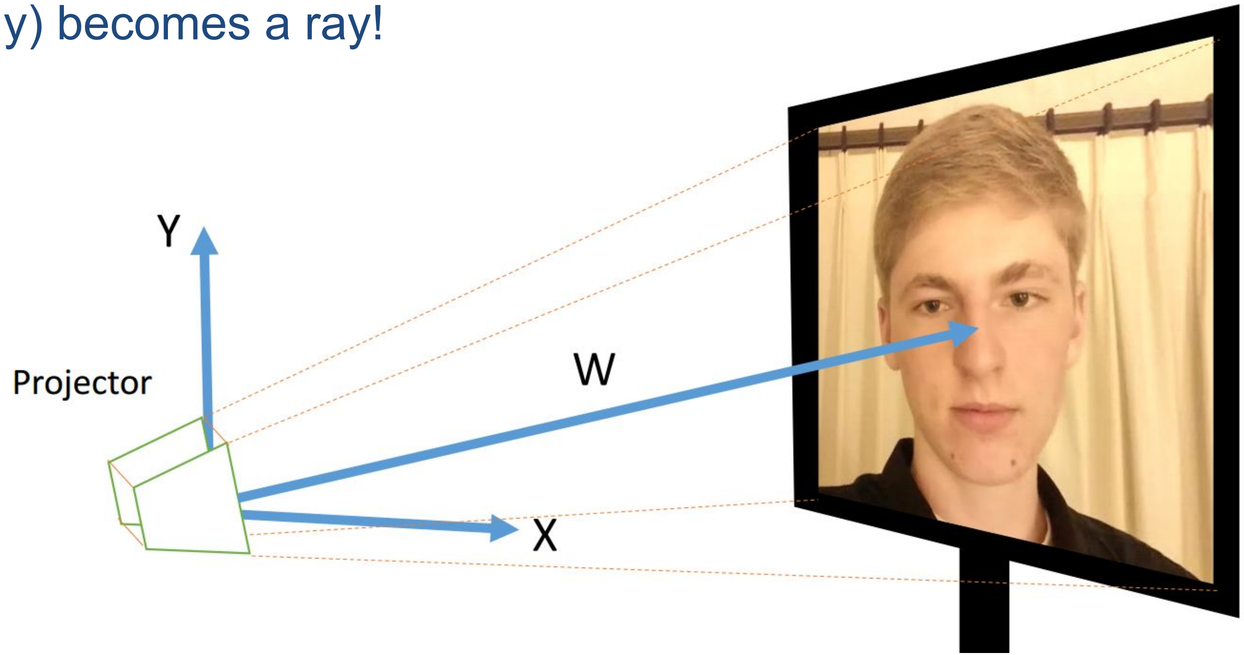
Projected image becomes smaller.

Projective geometry

2D point in projective = (x,y,w) coordinates

w defines the scale of the projected image.

Each point (x,y) becomes a ray!

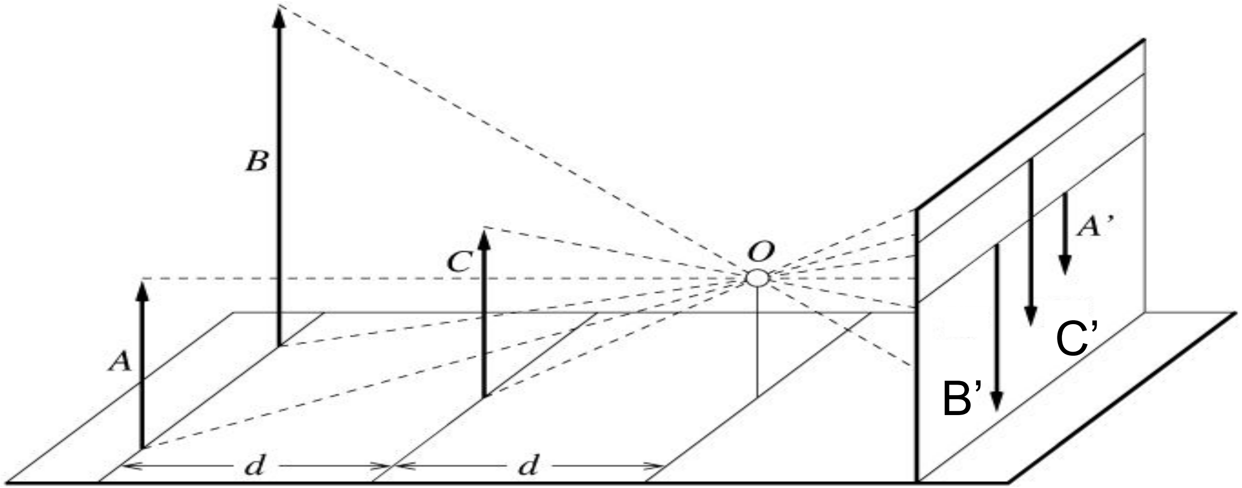


Projective geometry

In **3D**, point (x,y,z) becomes (x,y,z,w)

Perspective is w varying with z:

Objects far away are appear smaller



Homogeneous coordinates

Converting to homogeneous coordinates

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D (image) coordinates

$$(x, y, z) \Rightarrow \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3D (scene) coordinates

Converting from homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

2D (image) coordinates

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \Rightarrow (x/w, y/w, z/w)$$

3D (scene) coordinates

Homogeneous coordinates

Scale invariance in projection space

Homogeneous Coordinates

$$k\begin{bmatrix} x \\ y \\ w \end{bmatrix} = \begin{bmatrix} kx \\ ky \\ kw \end{bmatrix} \Rightarrow \begin{bmatrix} \frac{kx}{kw} \\ \frac{ky}{kw} \end{bmatrix} = \begin{bmatrix} \frac{x}{w} \\ \frac{y}{w} \end{bmatrix}$$

Cartesian Coordinates

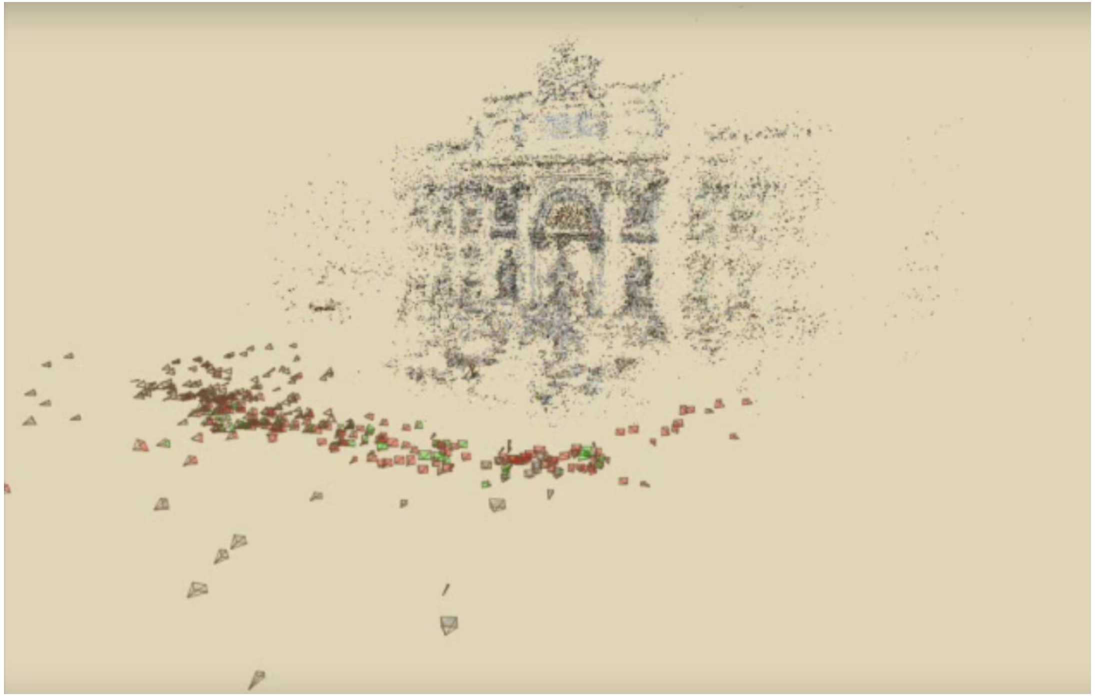
E.G., we can uniformly scale the projective space, and it will still produce the same image -> scale ambiguity

Pattern Recognition Lab

FRIEDRICH-ALEXANDER UNIVERSITÄT ERLANGEN-NÜRNBERG
FACULTY OF ENGINEERING

Homogeneous coordinates -- reference scale is important

Photo Tourism paper:



Basic geometry in homogeneous coordinates

- Line equation: ax + by + c = 0

$$line_i = \begin{bmatrix} a_i \\ b_i \\ c_i \end{bmatrix}$$

- Append 1 to pixel coordinate to get homogeneous coordinate

$$p_i = \begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix}$$

- Line given by cross product of two points

$$line_{ij} = p_i \times p_j$$

- Intersection of two lines given by cross product of the lines

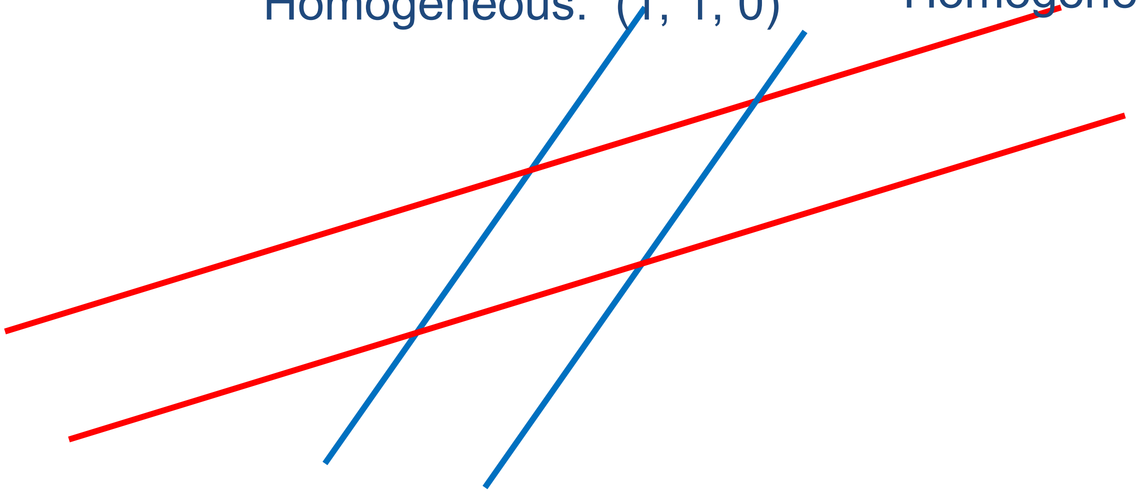
$$q_{ij} = line_i \times line_j$$

Another problem solved by homogeneous coordinates

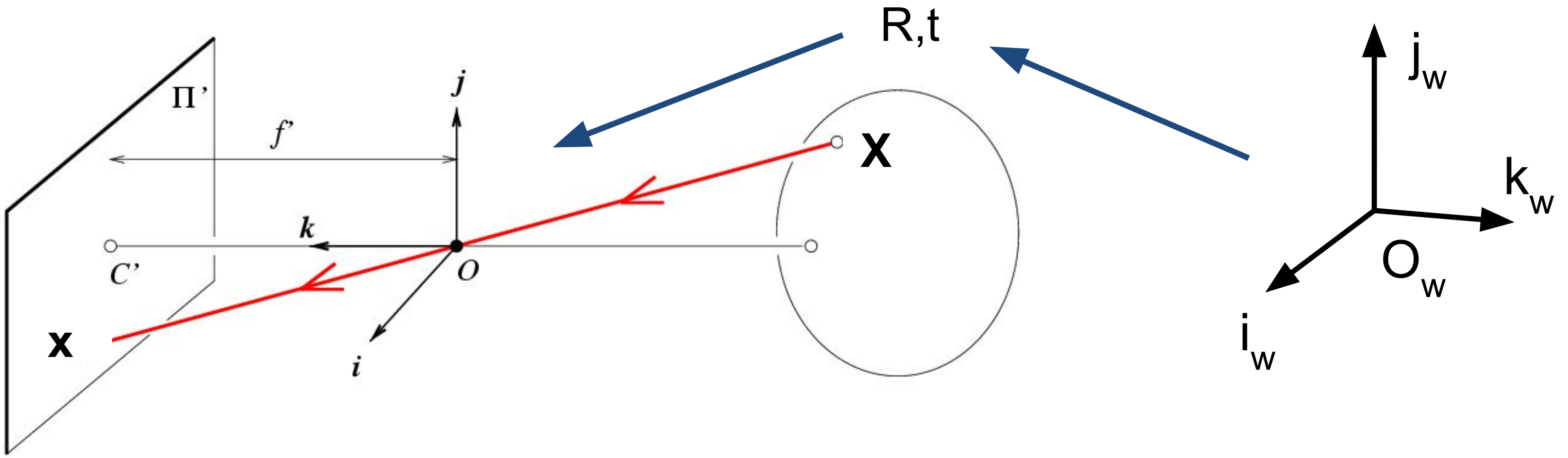
Intersection of parallel lines

Cartesian: (Inf, Inf)
Homogeneous: (1, 1, 0)

Cartesian: (Inf, Inf)
Homogeneous: (1, 2, 0)



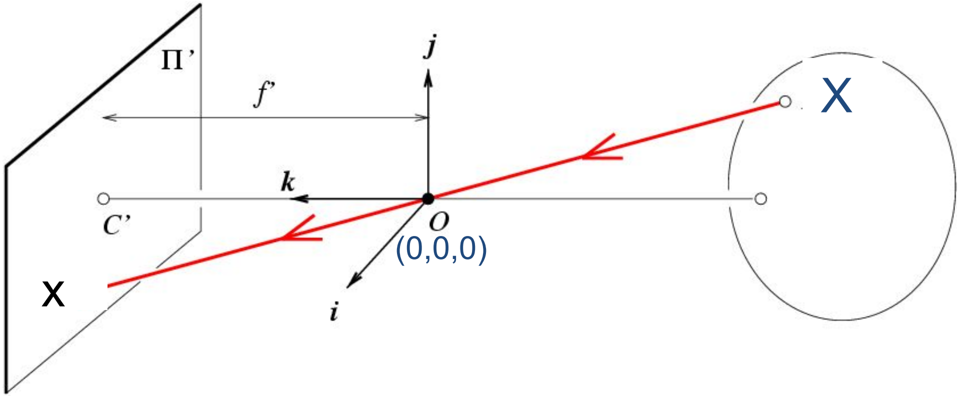
Camera (projection) matrix



$$\mathbf{x} = \mathbf{K}\underbrace{\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}}_{\text{Extrinsic Matrix}}\mathbf{X}$$

x: Image Coordinates: (u,v,1)
K: Intrinsic Matrix (3x3)
R: Rotation (3x3)
t: Translation (3x1)
X: World Coordinates: (X,Y,Z,1)

Projection matrix



Intrinsic Assumptions

- Unit aspect ratio
- Optical center at (0,0)
- No skew

Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}[\mathbf{I} \quad \mathbf{0}]\mathbf{X} \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overbrace{\begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}^{\mathbf{K}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Slide Credit: Savarese

Remove assumption: known optical center

Intrinsic Assumptions
- Unit aspect ratio
- No skew

Extrinsic Assumptions
- No rotation
- Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

K

Remove assumption: equal aspect ratio

Intrinsic Assumptions

- No skew

Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}f_x & 0 & u_0 & 0\\0 & f_y & v_0 & 0\\0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

Remove assumption: non-skewed pixels

Intrinsic Assumptions

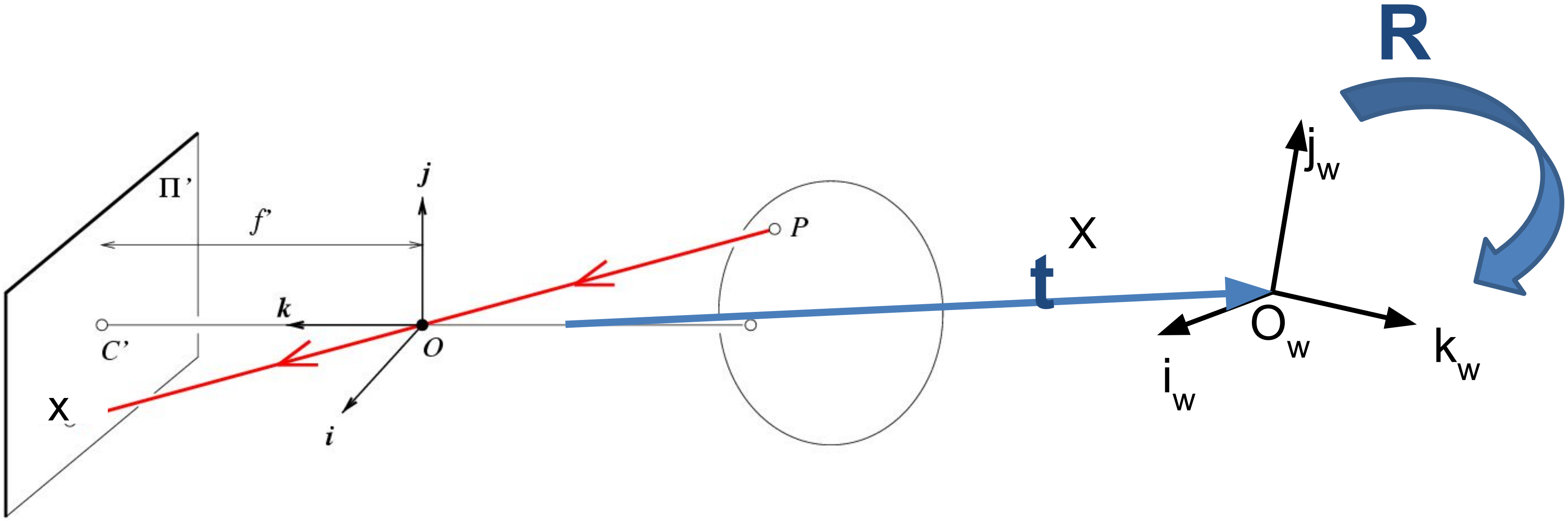
Extrinsic Assumptions

- No rotation
- Camera at (0,0,0)

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}f_x & s & u_0 & 0\\0 & f_y & v_0 & 0\\0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x\\y\\z\\1\end{bmatrix}$$

Note: different books use different notation for parameters

Oriented and Translated Camera



Allow camera translation

Intrinsic Assumptions

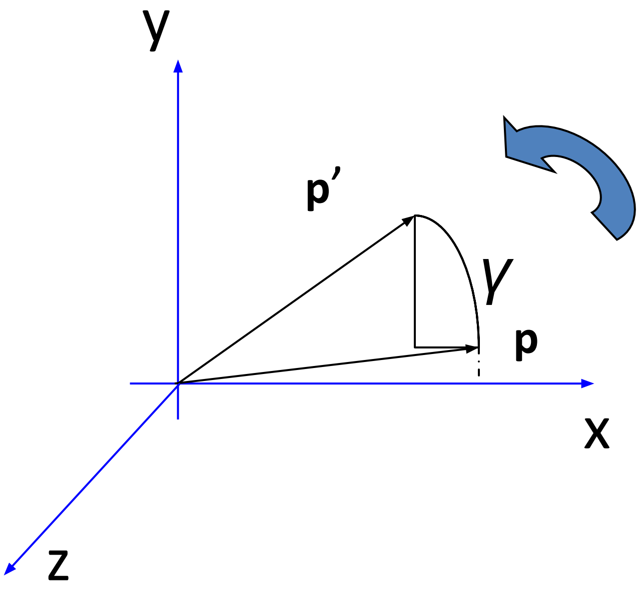
Extrinsic Assumptions

- No rotation

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{I} & \mathbf{t}\end{bmatrix}\mathbf{X} \Rightarrow w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

3D Rotation of Points

Rotation around the coordinate axes, counter-clockwise:



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

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

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

Slide Credit: Savarese

Allow camera rotation

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Degrees of freedom

$$\mathbf{x} = \mathbf{K}\begin{bmatrix}\mathbf{R} & \mathbf{t}\end{bmatrix}\mathbf{X}$$

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overset{5}{\begin{bmatrix} f_x & s & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{6}{\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Demo – Kyle Simek

"Dissecting the Camera Matrix"

Three-part blog series

1. Dissecting the Camera Matrix, Part 1: Extrinsic/Intrinsic Decomposition ←
2. Dissecting the Camera Matrix, Part 2: The Extrinsic Matrix ←
3. Dissecting the Camera Matrix, Part 3: The Intrinsic Matrix ←

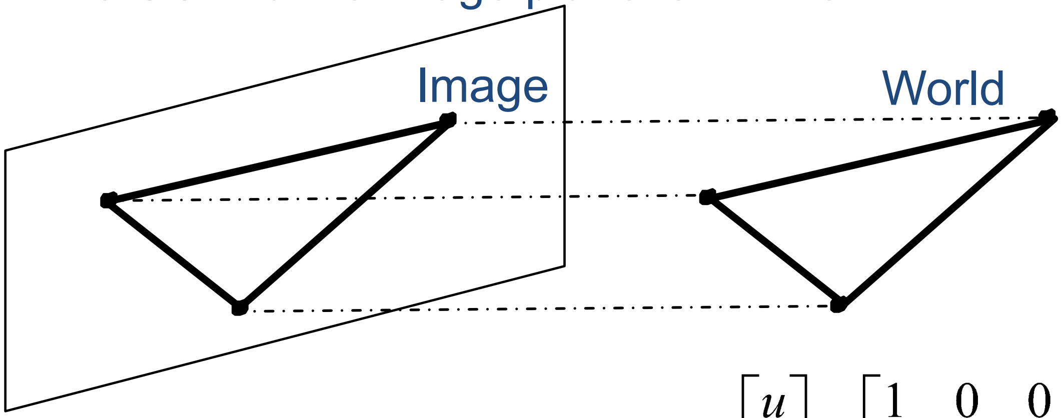
"Perspective toy"

Perspective Camera Toy ←

Orthographic Projection

Special case of perspective projection

- Distance from the COP to the image plane is infinite



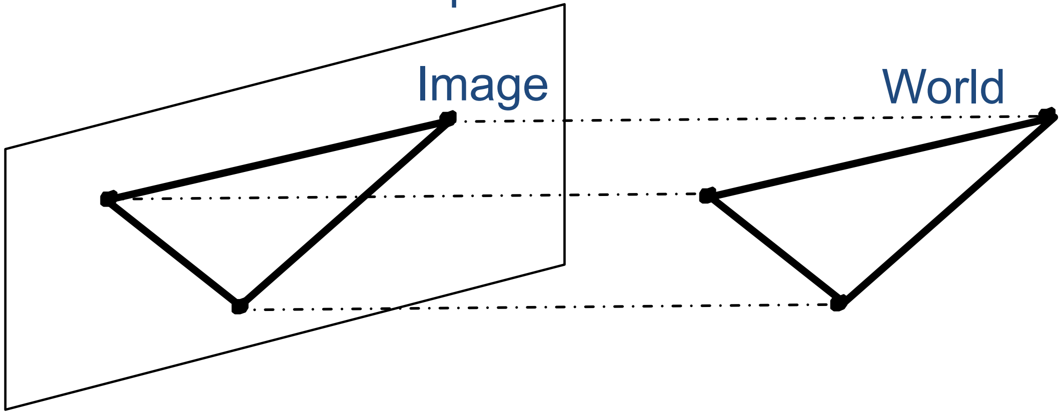
- Also called “parallel projection”
- What’s the projection matrix?

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Scaled Orthographic Projection

Special case of perspective projection

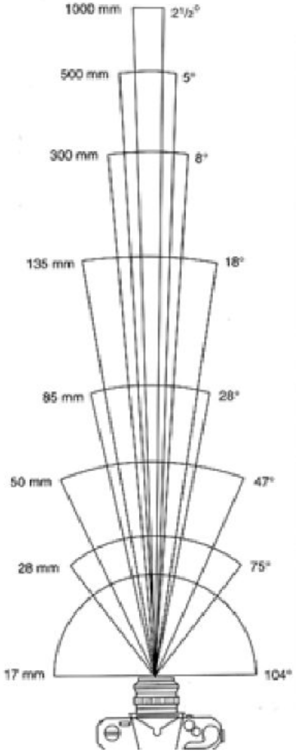
- Object dimensions are small compared to distance to camera



- Also called “weak perspective”
- What’s the projection matrix?

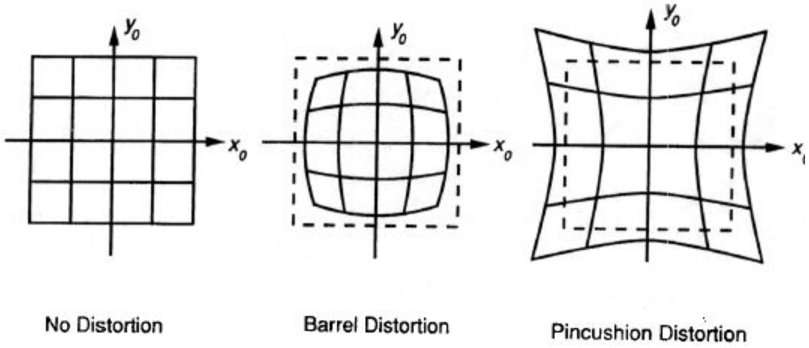
$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Field of View (Zoom, focal length)



From London and Upton

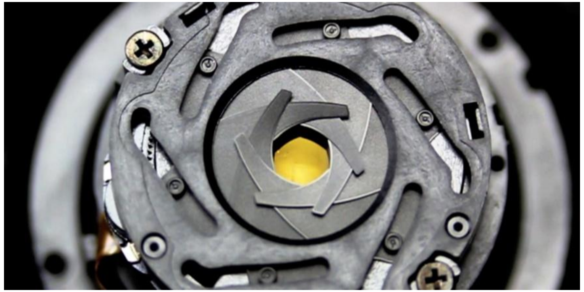
Beyond Pinholes: Radial Distortion



Corrected Barrel Distortion

Image from Martin Habbecke

Beyond Pinholes: Real apertures

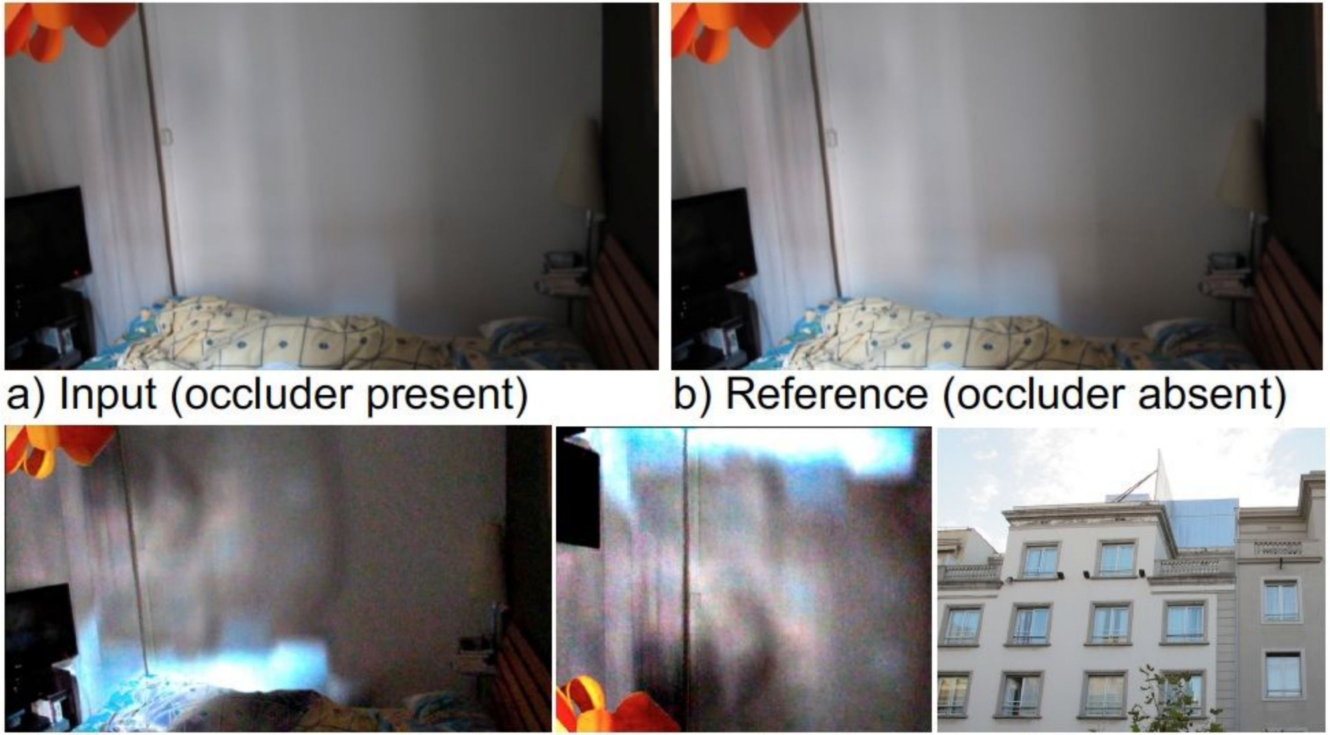


depth of focus

Accidental Cameras

Accidental Pinhole and Pinspeck Cameras
Revealing the scene outside the picture.
Antonio Torralba, William T. Freeman

Accidental Cameras

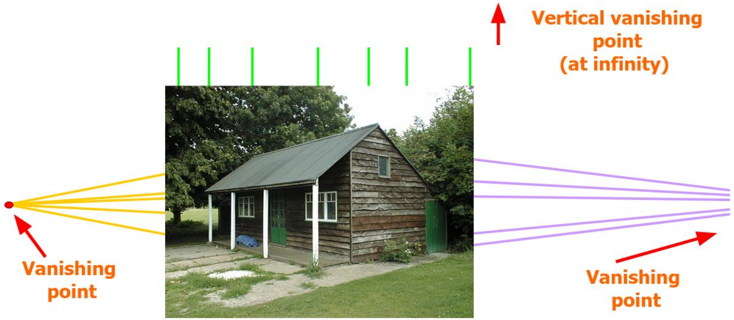


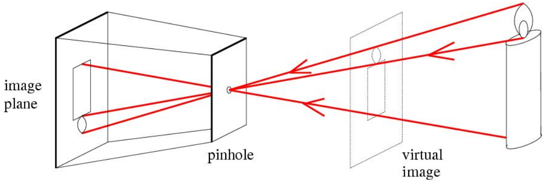
a) Input (occluder present) b) Reference (occluder absent)

c) Difference image (b-a) d) Crop upside down e) True view

Things to remember

- Vanishing points and vanishing lines
- Pinhole camera model and camera projection matrix
- Homogeneous coordinates





$$\mathbf{x} = \mathbf{K}[\mathbf{R} \quad \mathbf{t}]\mathbf{X}$$

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$