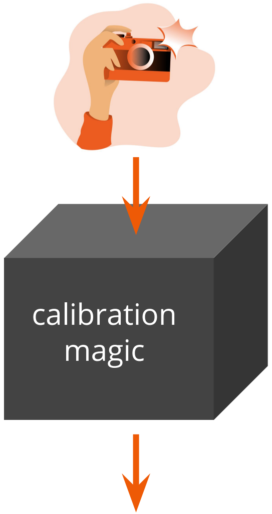
Geometry in Computer Vision

Yaroslava Lochman

Dec 2019

Geometry in CV. Outcome

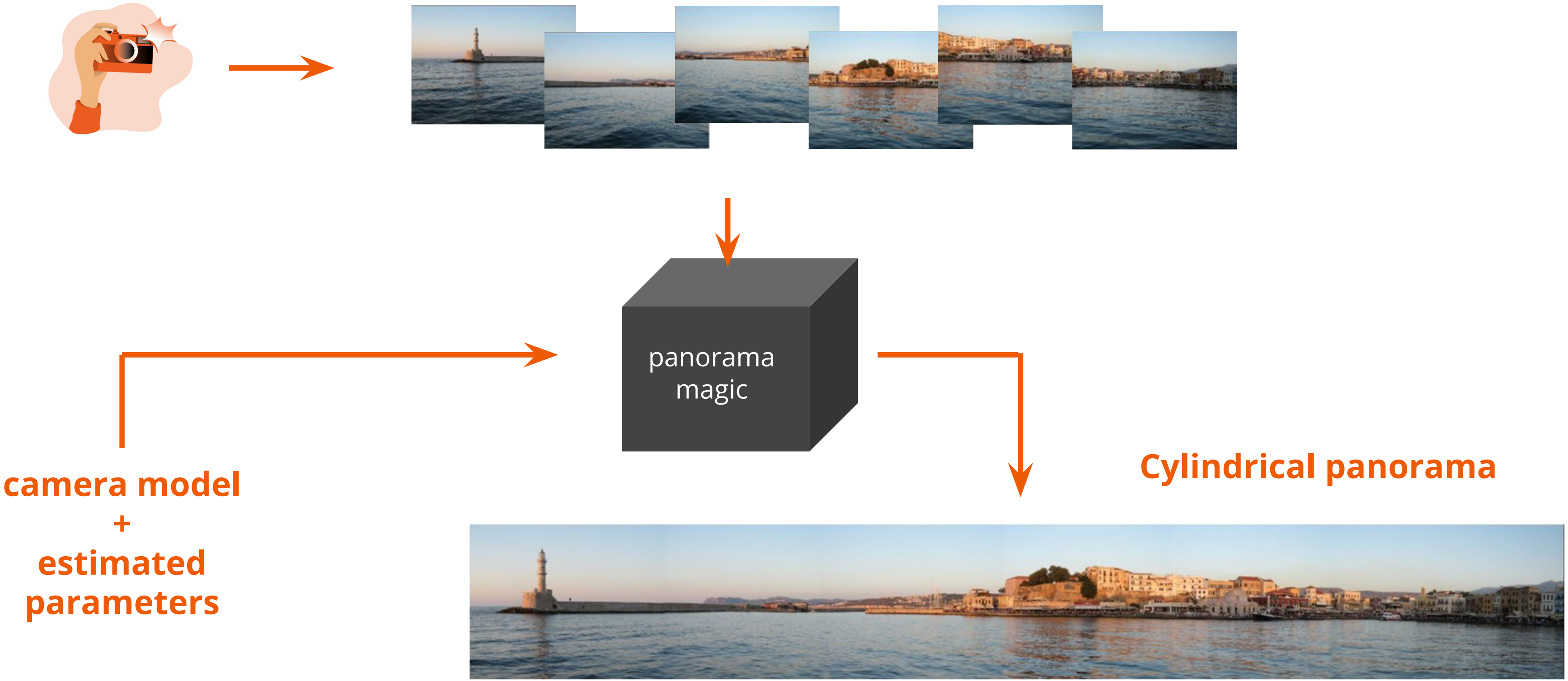


camera model

+

estimated parameters

Geometry in CV. Outcome



What else can you do with geometry?

Well…

What else can you do with geometry?

Other panoramas (projections) :)

Fisheye

Stereographic

What else can you do with geometry?

Other panoramas (projections) :)

Rectilinear

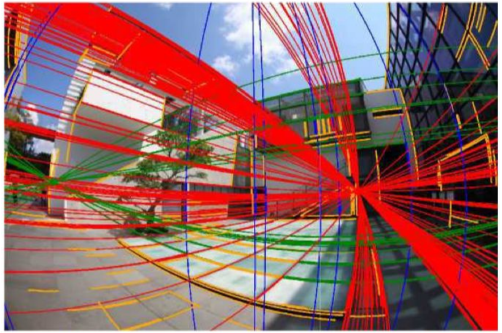
Cube

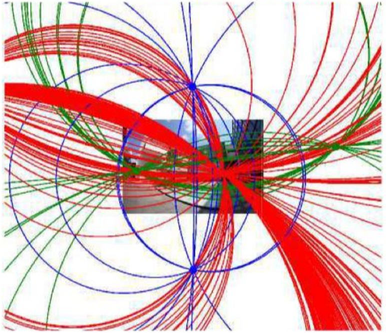
What else can you do with geometry?

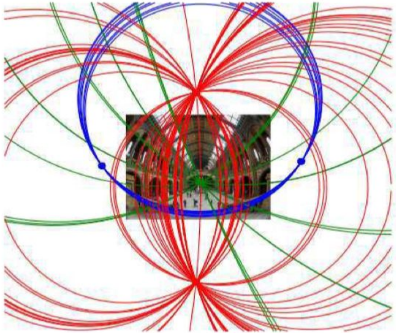
Scene Plane Rectification

What else can you do with geometry?

Scene Parsing







What else can you do with geometry?

Reconstruction of 3D objects (Structure from Motion)

What else can you do with geometry?

Augmented Reality

Credits

[1] Pajdla, Tomas. Elements of geometry for computer vision. FEE CTU, 2013

[2] Hartley, Richard, and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003

[3] Šára, Radim. TDV − 3D Computer Vision, Winter 2017

[4] Gkioulekas, Ioannis. Computational Photography, Fall 2019

[5] Kutulakos, Kyros. Computer Graphics, Fall 2010

Geometry in CV
part 1

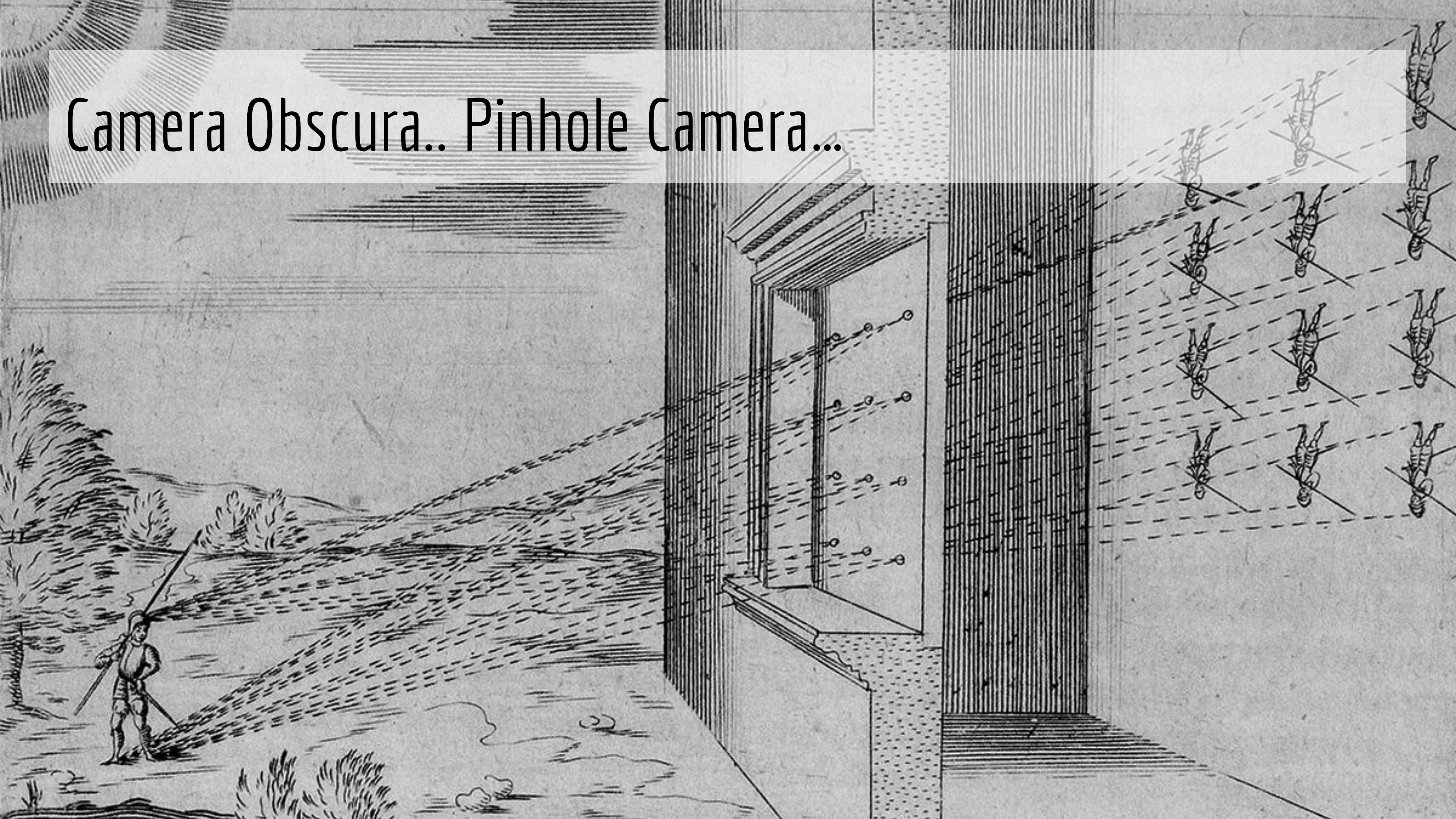
Camera Calibration

Yaroslava Lochman
Dec 10, 2019

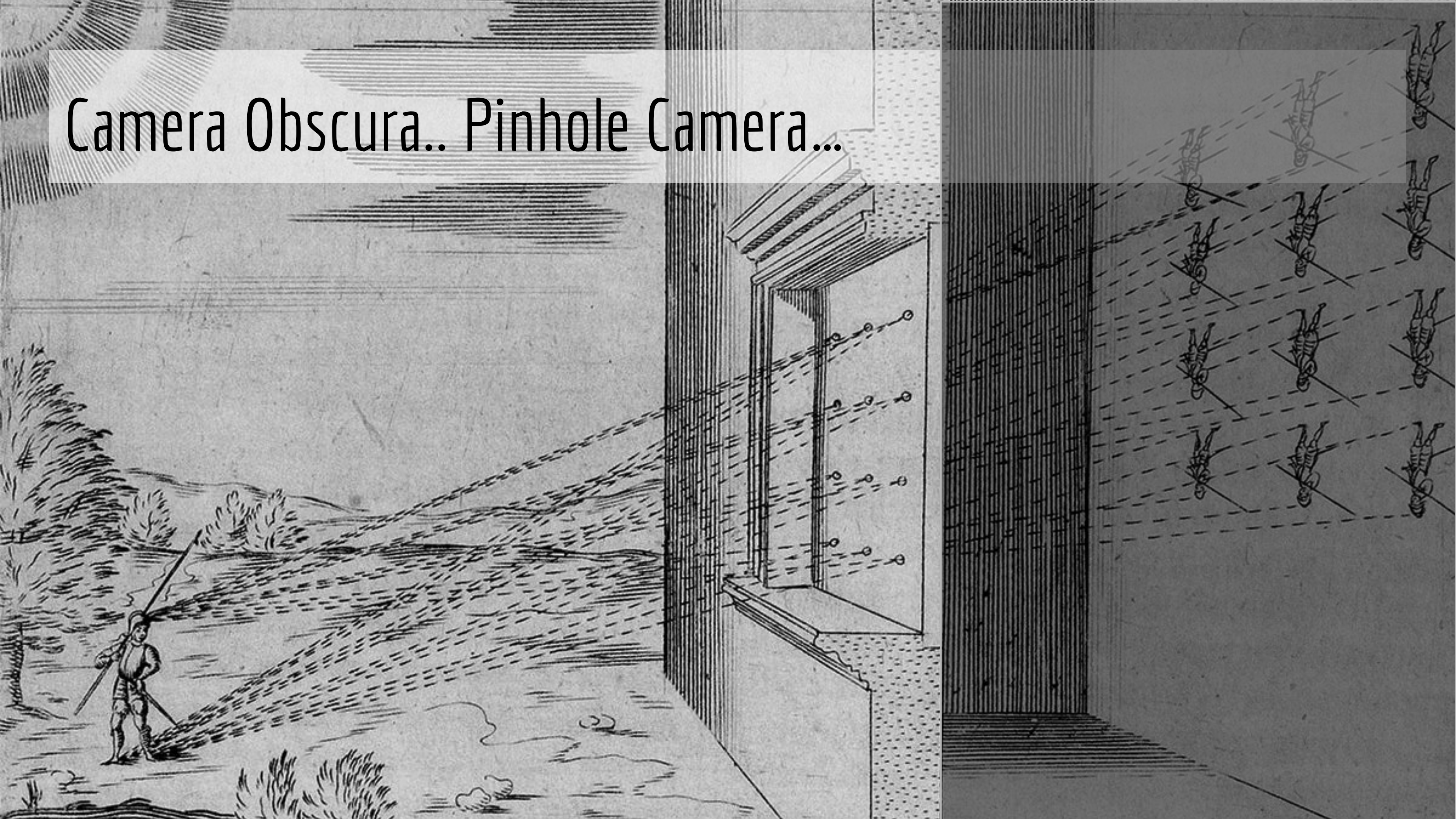
Outline

- Camera Intrinsics
  - Essential Internal Properties
- Camera Extrinsics
  - 3D Camera Pose
- Modeling Image Formation
  - Perspective Camera, Distortion Models
- Camera Calibration
  - Calibration from Vanishing Points
  - Calibration from Vanishing Points

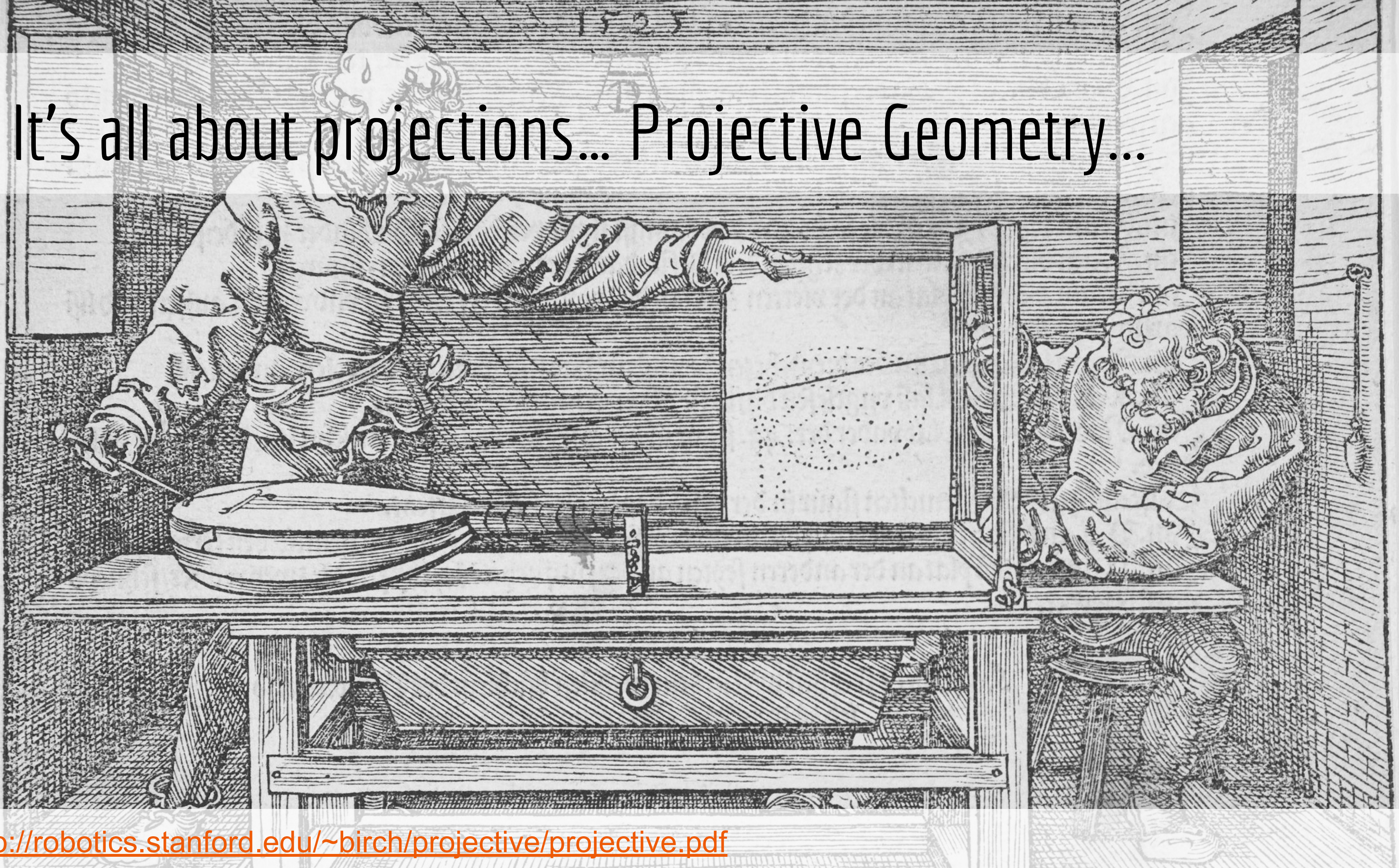
Camera Obscura.. Pinhole Camera...



Camera Obscura.. Pinhole Camera...



It's all about projections… Projective Geometry…



http://robotics.stanford.edu/~birch/projective/projective.pdf

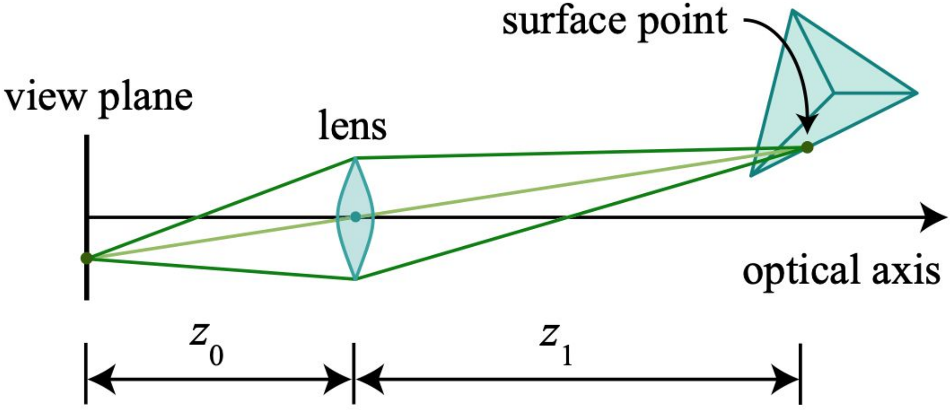
Camera Intrinsics

Camera Extrinsics

Modeling Image Formation

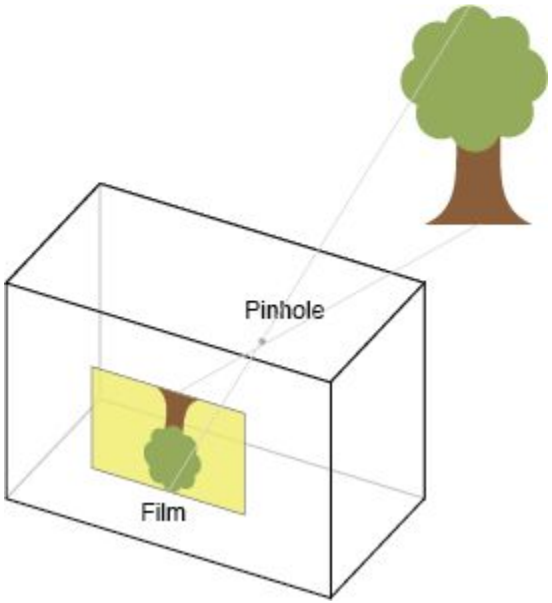
Camera Calibration

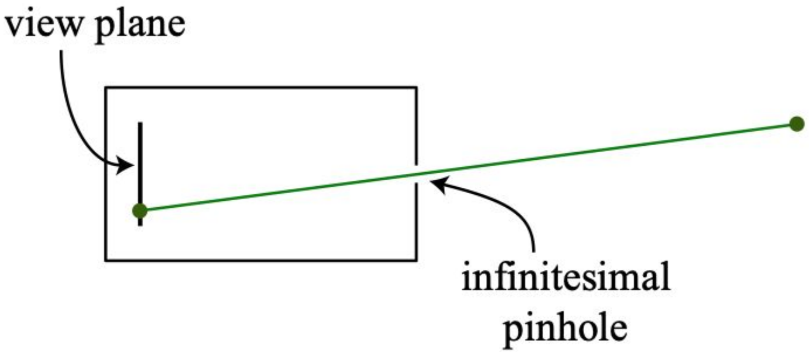
Thin Lens Model



$$\frac{1}{|f|} = \frac{1}{z_0} + \frac{1}{z_1}$$

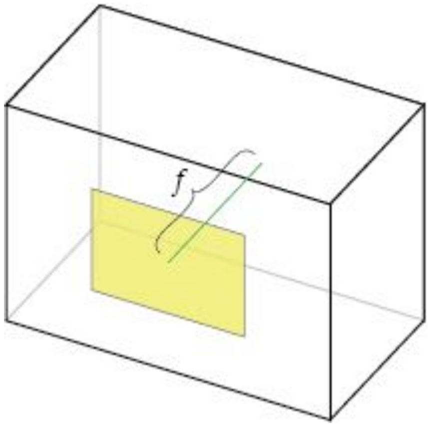
Pinhole Camera Model



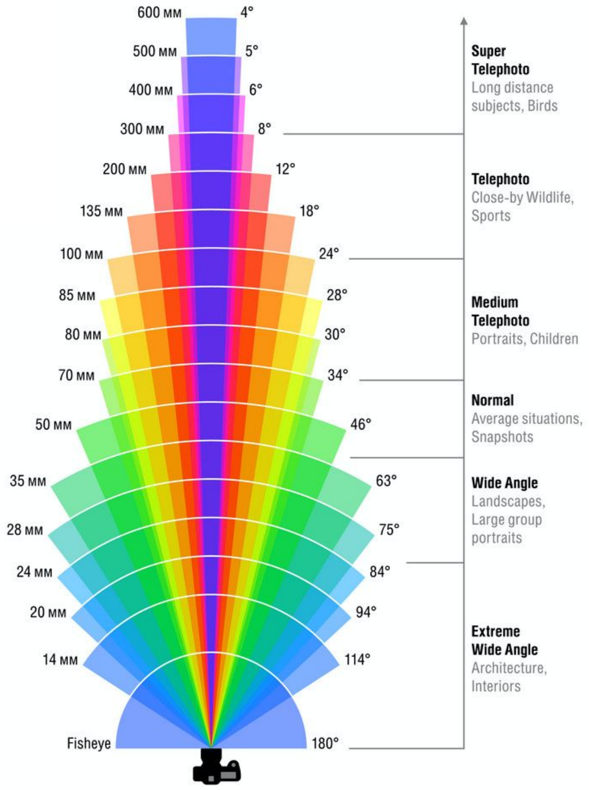


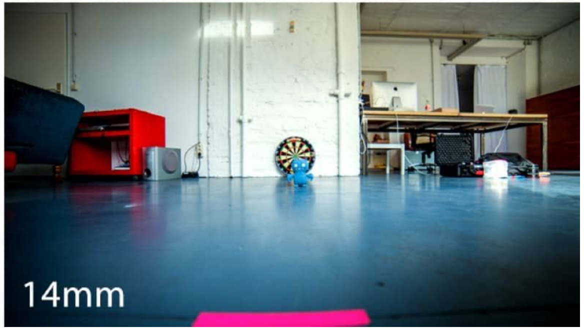
A pinhole camera is an idealization of the thin lens as aperture shrinks to zero.

Focal Length in Pinhole Camera



Focal Length in Real Camera





Focal Length & Perspective

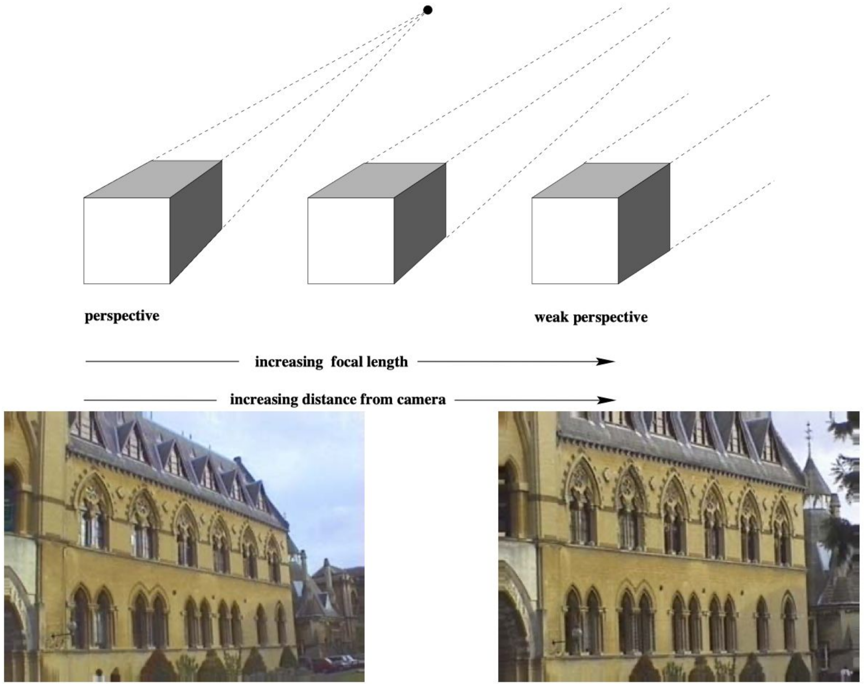
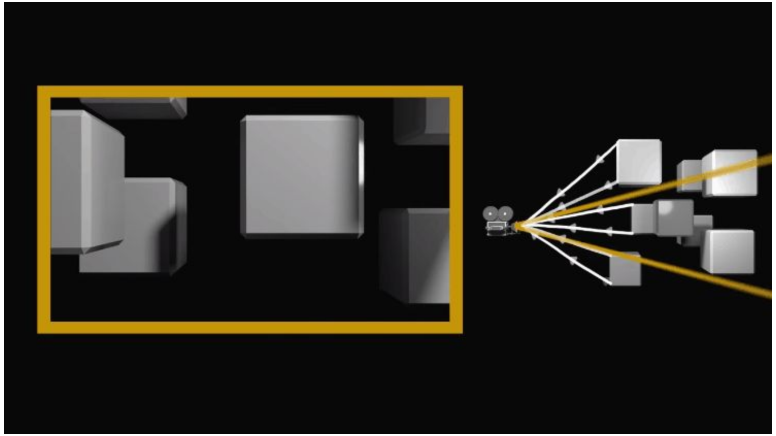
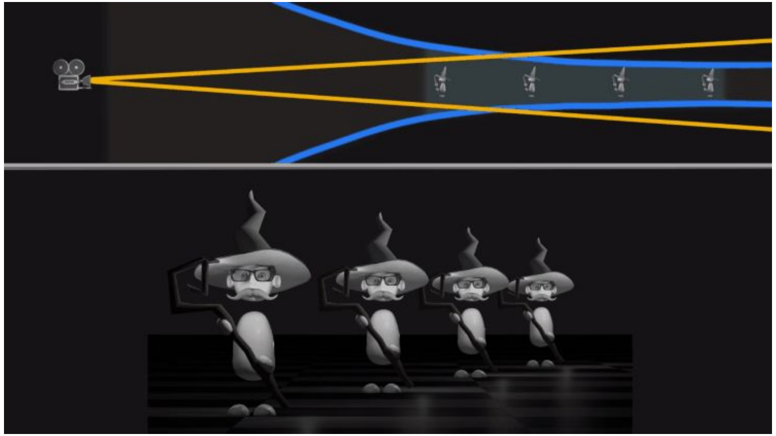
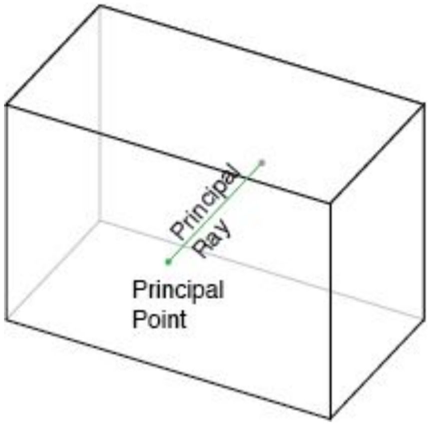


Fig. 6.7. *As the focal length increases and the distance between the camera and object also increases, the image remains the same size but perspective effects diminish.*

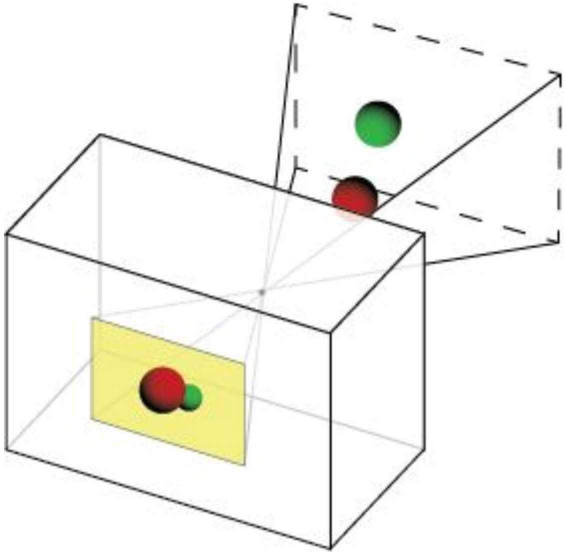




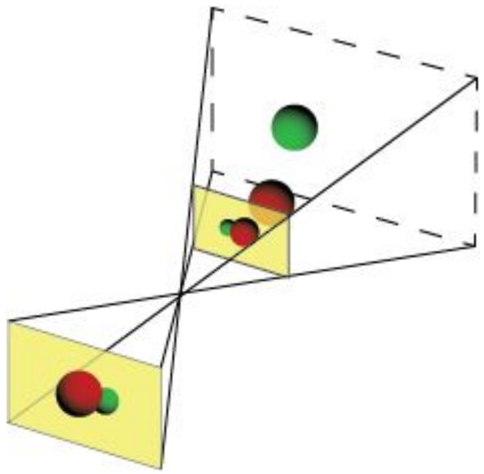
Principal Point (PP)



Viewing Frustrum



Viewing Frustrum



Viewing Frustrum

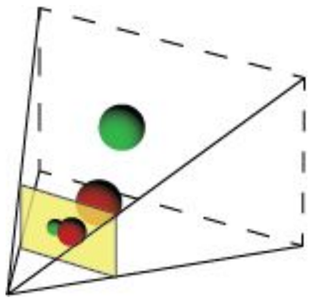
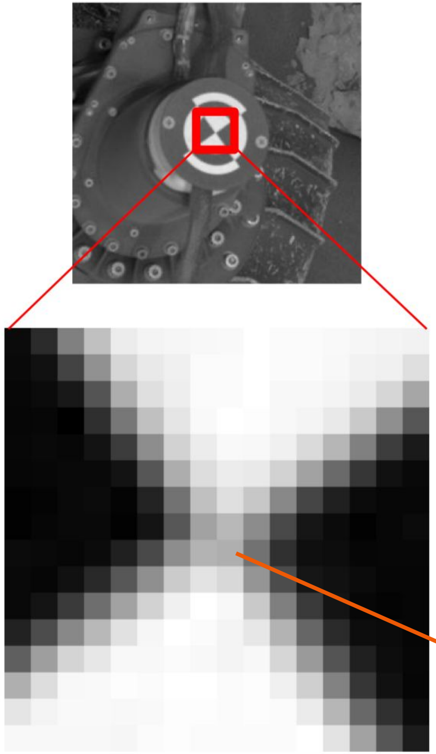


Image Units

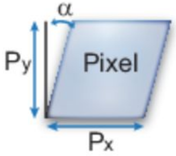


Film
(continuous, mm)

Image*
(discrete, pixels)

*from charge-coupled device
(CCD) image sensor

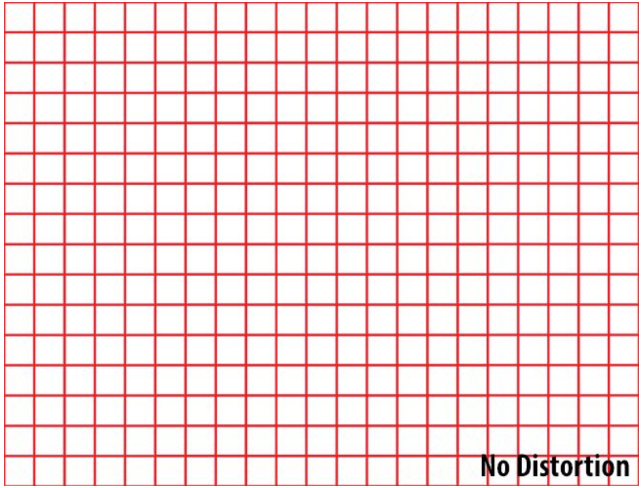
Orthogonal Raster, Unit Aspect (ORUA)
or **square pixels**



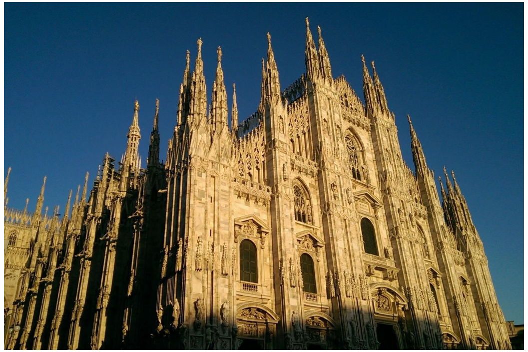
non-square pixels
with **skew** and **non-unit aspect**

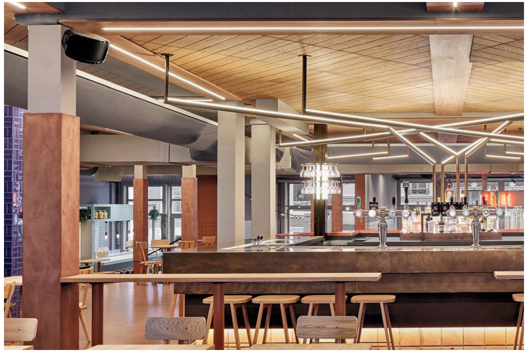
Anamorphic Format

Lens Distortion

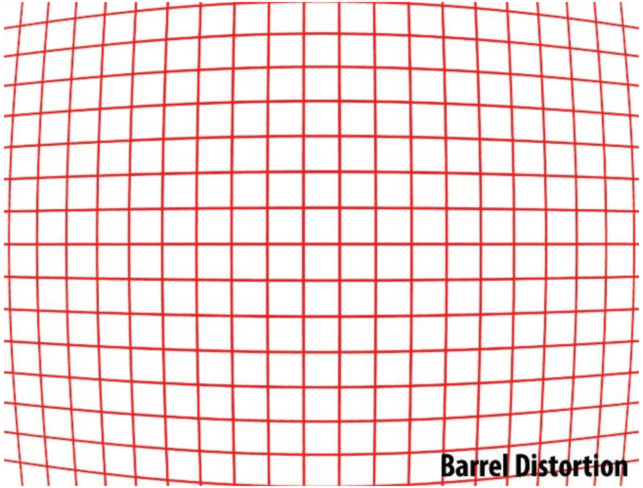


NO DISTORTION



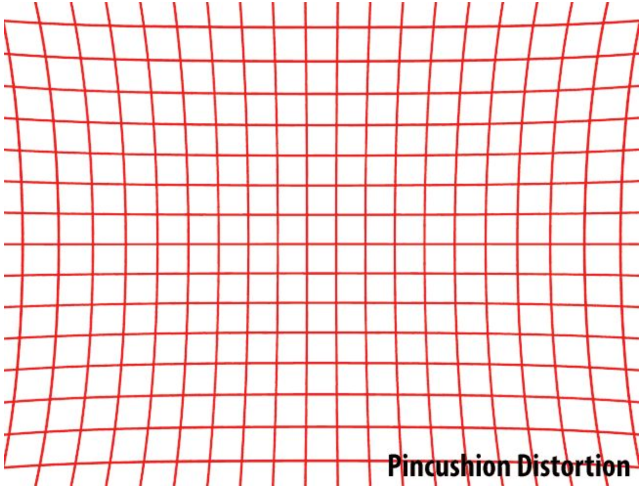


Lens Distortion



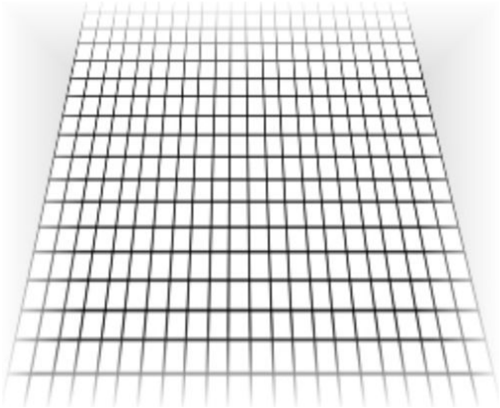
"BARREL" RADIAL DISTORTION

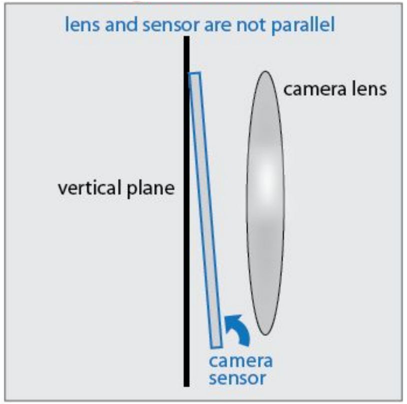
Lens Distortion

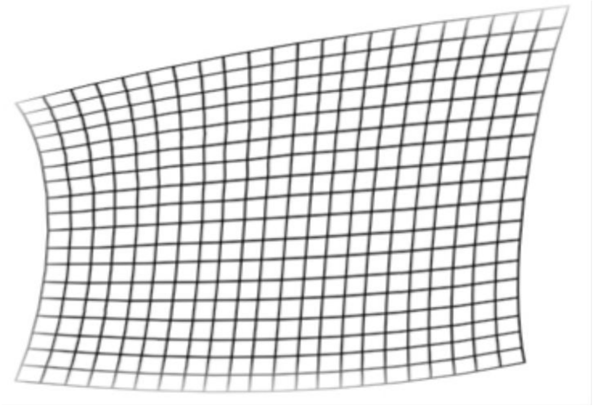


"PINCUSHION" RADIAL DISTORTION

Lens Distortion







TANGENTIAL DISTORTION
(PERSPECTIVE DISTORTION,
KEYSTONE DISTORTION)

COMPLEX

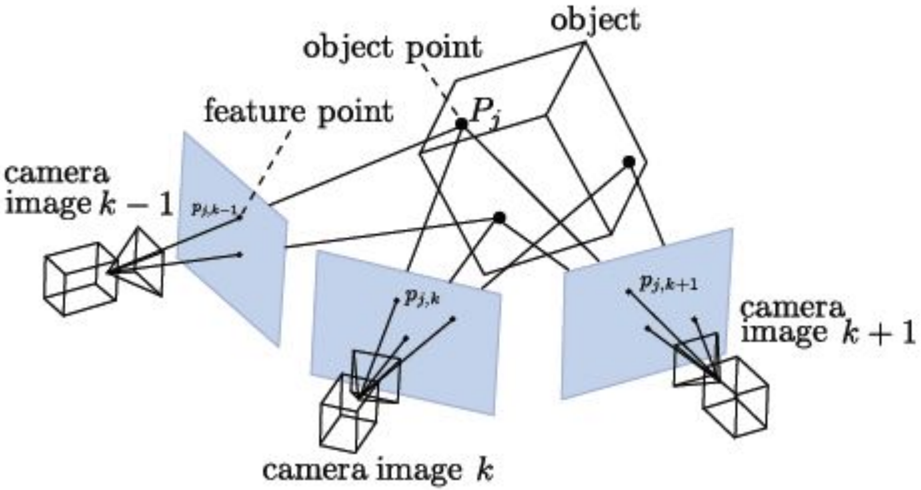
Camera Intrinsics

Camera Extrinsics

Modeling Image Formation

Camera Calibration

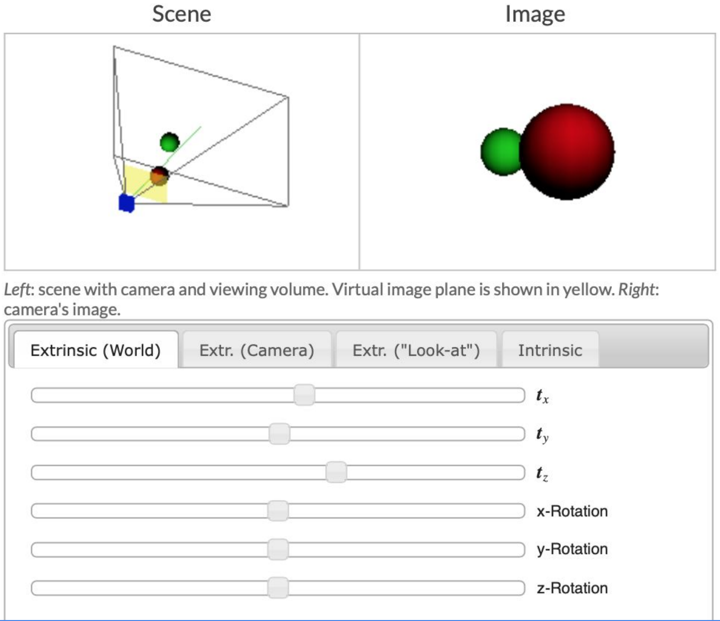
3D Location and Orientation



Demo: Synthetic Camera

Intrinsics: https://ksimek.github.io/2013/08/13/intrinsic

Extrinsics: https://ksimek.github.io/2012/08/22/extrinsic



Camera Model

Camera Pose

Modeling Image Formation*

Camera Calibration

*Geometrical Point of View

Camera Model

Camera

mapping from the 3D world space to a 2D image space given by a physical camera.

Camera model (mathematical model of a camera)

matrix with particular properties that represent the camera mapping

Homogeneous coordinates

a way of representing N-dimensional coordinates with N+1 numbers. A point $\mathbf{(x,y)^T}$ on the Euclidean plane lifted to R3, e.g., represented as point $\mathbf{(x,y,1)^T}$. Since all points along a ray project the same, the scale is unimportant so we consider all $\mathbf{(\alpha x,\alpha y,\alpha)^T}$ equivalent to the Euclidean point $\mathbf{(x,y,1)^T}$, where $\boldsymbol{\alpha}$ is nonzero. **Motivation:** translations and projections can be represented as linear operations.

Cameras Hierarchy

PINHOLE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

Cameras Hierarchy

CCD CAMERA (PERSPECTIVE CAMERA) MODEL

$$\mathrm{P} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathrm{R} & -\mathrm{R}\widetilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} = \mathrm{K}[\mathrm{R} \mid \mathbf{t}]$$

PINHOLE CAMERA MODEL

$$\mathrm{P} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathrm{R} & -\mathrm{R}\widetilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} = \mathrm{K}[\mathrm{R} \mid \mathbf{t}]$$

Cameras Hierarchy

FINITE PROJECTIVE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

CCD CAMERA (PERSPECTIVE CAMERA) MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

PINHOLE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

Cameras Hierarchy

GENERAL PROJECTIVE CAMERA MODEL

$\mathtt{P}$ — any arbitrary homogeneous 3x4 matrix of rank 3.

FINITE PROJECTIVE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ \mathbf{0} & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

CCD CAMERA (PERSPECTIVE CAMERA) MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ \mathbf{0} & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

PINHOLE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\widetilde{\mathbf{C}} \\ \mathbf{0} & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

Cameras Hierarchy

GENERAL PROJECTIVE CAMERA MODEL

$\mathtt{P}$ — any arbitrary homogeneous 3x4 matrix of rank 3.

LENS DISTORTED CAMERA

$$\boxed{\alpha \mathbf{x} = \mathtt{P}\mathbf{X}}$$

+

$$\boxed{\beta \tilde{\mathbf{x}} = f^d(\mathbf{x}, \lambda)}$$

FINITE PROJECTIVE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

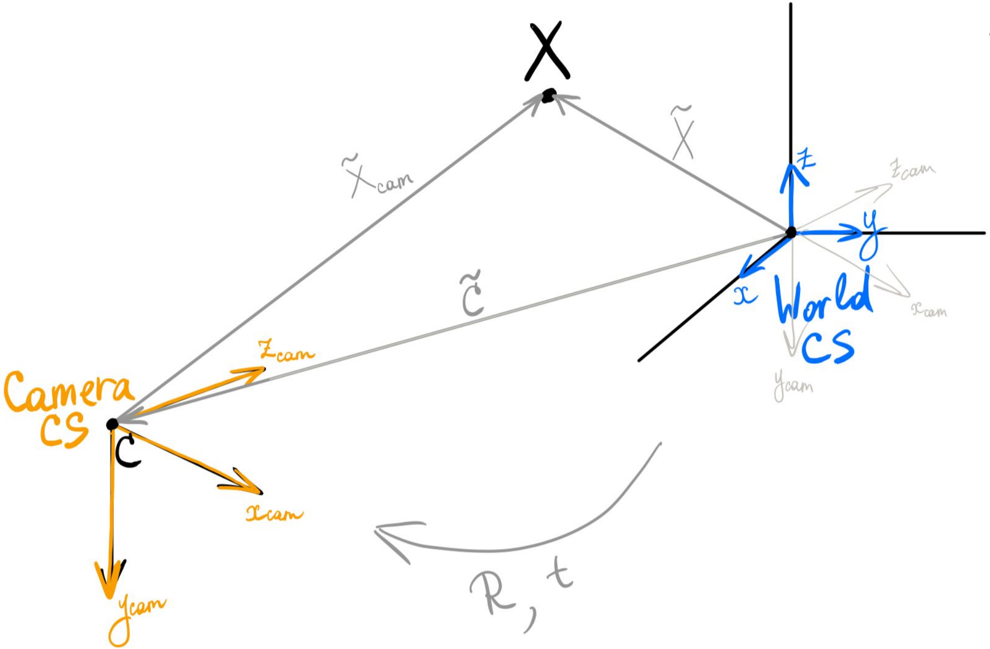
CCD CAMERA (PERSPECTIVE CAMERA) MODEL

$$\mathtt{P} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

PINHOLE CAMERA MODEL

$$\mathtt{P} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$$

Change of Basis: Camera Rotation and Translation



$$R = \begin{bmatrix} - x_{cam} - \\ - y_{cam} - \\ - z_{cam} - \end{bmatrix}$$

"look right"
"look down"
"look at"

$$t = -R\tilde{C}$$

Inhomogeneous coordinates:

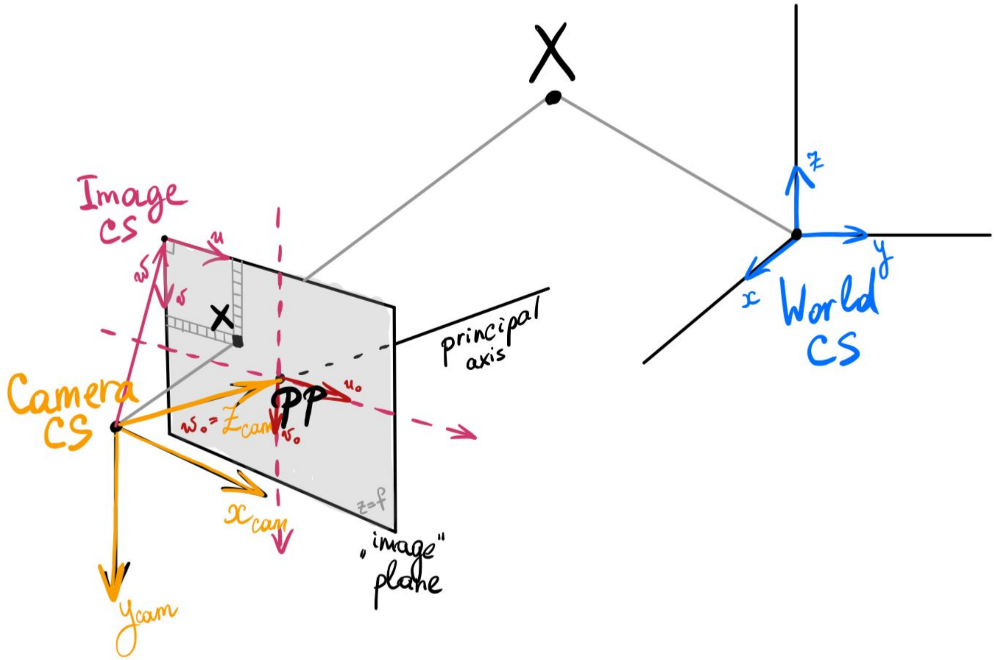
$$\tilde{\mathbf{X}}_{\mathrm{cam}} = \mathrm{R}(\tilde{\mathbf{X}} - \tilde{\mathbf{C}})$$

Homogeneous coordinates:

$$\mathbf{X}_{\mathrm{cam}} = \begin{bmatrix} \mathrm{R} & -\mathrm{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix} = \begin{bmatrix} \mathrm{R} & -\mathrm{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} \mathbf{X}$$

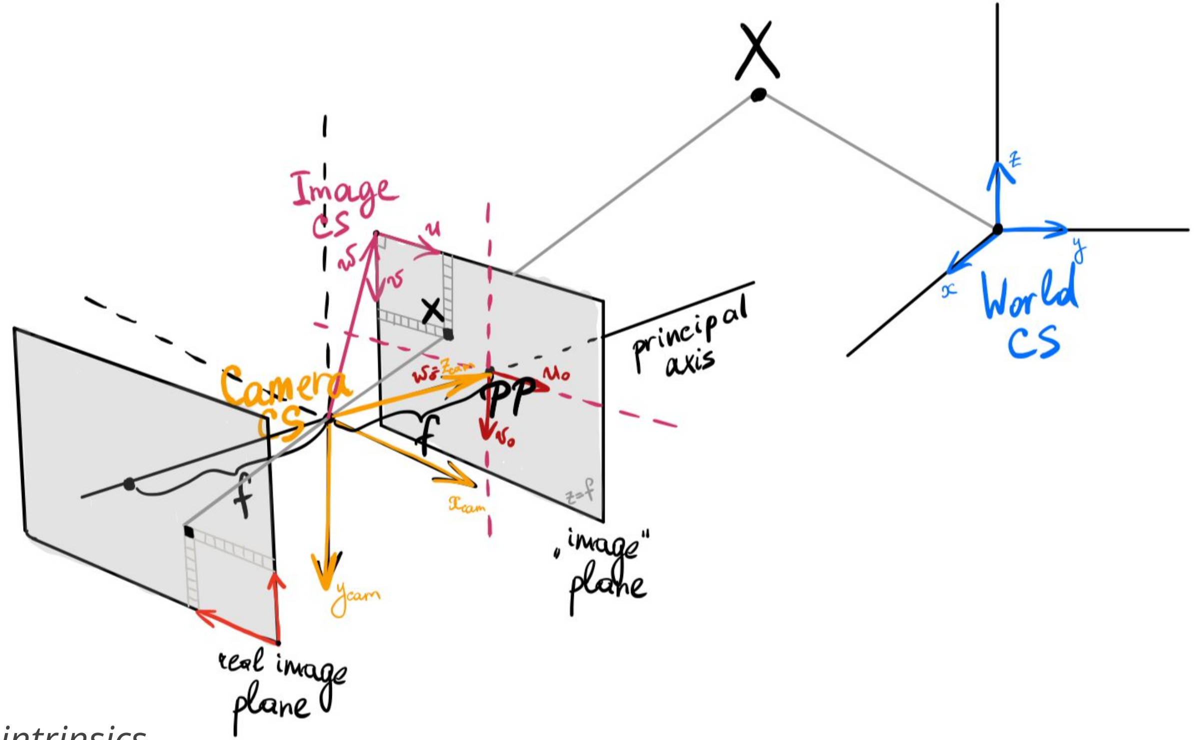
Recall camera extrinsics…

Change of Basis: Central Projection



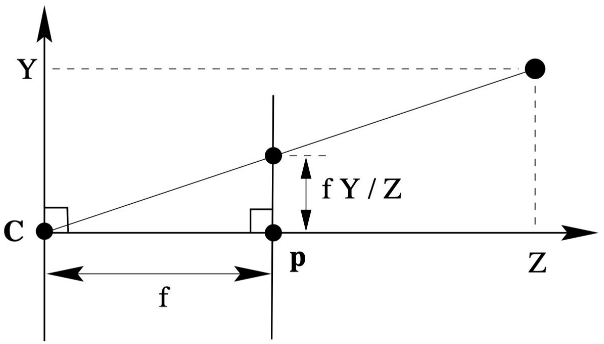
Recall camera intrinsics...

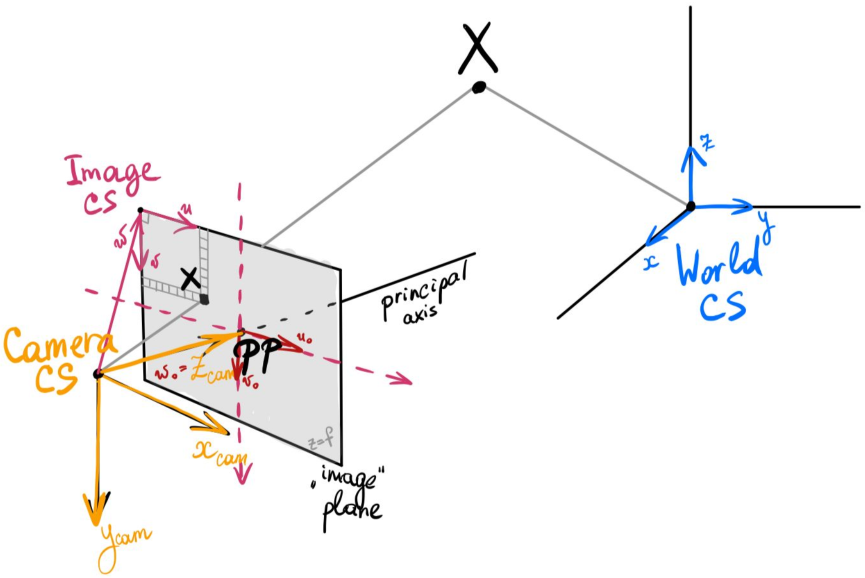
Change of Basis: Central Projection



Recall camera intrinsics...

Change of Basis: Central Projection





Inhomogeneous coordinates:

$$(X, Y, Z)^{\mathsf{T}} \mapsto (fX/Z, fY/Z)^{\mathsf{T}}$$

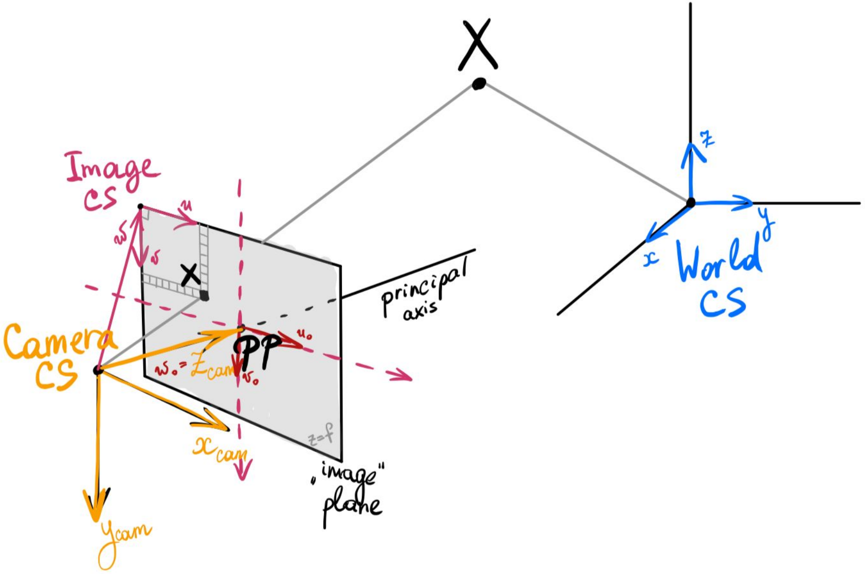
Homogeneous coordinates:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \left[\begin{array}{ccc|c} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{array}\right] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Camera Matrix **K**

Recall camera intrinsics…

Change of Basis: PP offset



Inhomogeneous coordinates:

$$(\mathrm{X}, \mathrm{Y}, \mathrm{Z})^\mathsf{T} \mapsto (f\mathrm{X}/\mathrm{Z} + p_x, f\mathrm{Y}/\mathrm{Z} + p_y)^\mathsf{T}$$

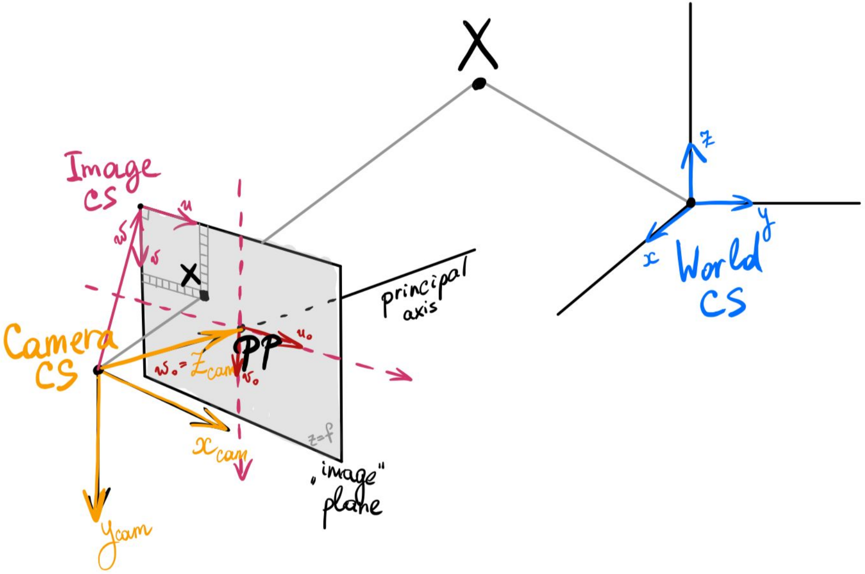
Homogeneous coordinates:

$$\begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} f\mathrm{X} + \mathrm{Z}p_x \\ f\mathrm{Y} + \mathrm{Z}p_y \\ \mathrm{Z} \end{pmatrix} = \left[\begin{array}{ccc|c} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{array} \right] \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix}$$

Camera Matrix **K**

Recall camera intrinsics...

Change of Basis: CCD to Image



Homogeneous coordinates:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \left[\begin{array}{ccc|c} \alpha_x & & x_0 & 0 \\ & \alpha_y & y_0 & 0 \\ & & 1 & 0 \end{array} \right] \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Camera Matrix **K**

Recall camera intrinsics...

Perspective Camera

A world point is *imaged* into an image point $\boxed{\mathbf{x} = \mathtt{KR}[\mathtt{I} \mid -\tilde{\mathrm{C}}]\mathbf{X}}$

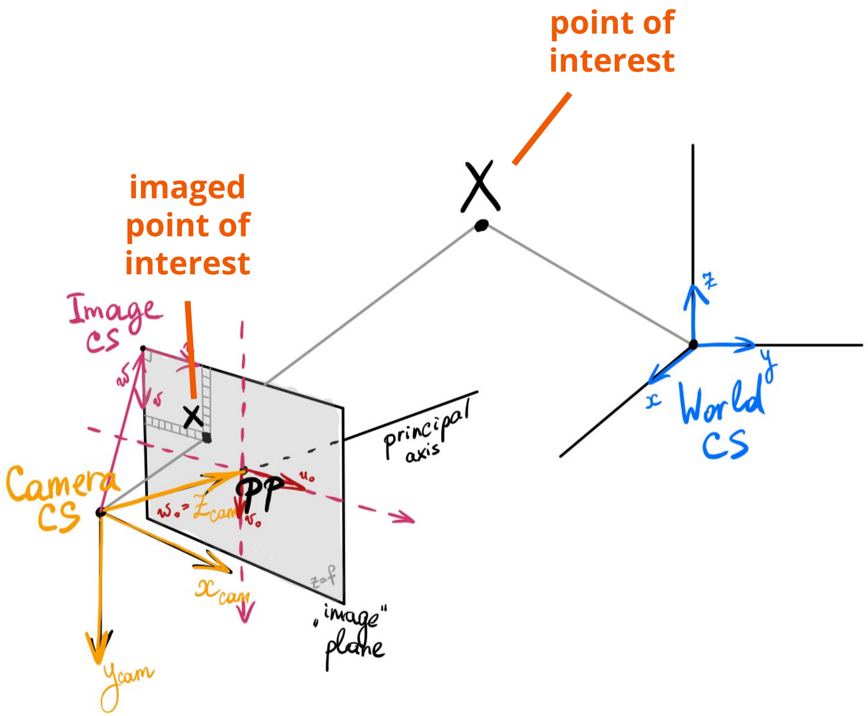
1. *expressed w.r.t. camera basis*

$$\mathbf{X}_{\text{cam}} = \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathrm{C}} \\ 0 & 1 \end{bmatrix} \mathbf{X}$$

2. *projected along ray from 3D to 2D*

$$\mathbf{x} = \mathtt{K}[\mathtt{I} \mid \mathbf{0}]\mathbf{X}_{\text{cam}}$$

$$\mathtt{K} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}$$



Lens Distorted Camera

A world point is *imaged* into an image point $\boxed{\mathbf{x} = \mathtt{K}\mathtt{R}[\mathtt{I} \mid -\tilde{\mathbf{C}}]\mathbf{X}}$

1. *expressed w.r.t. camera basis*

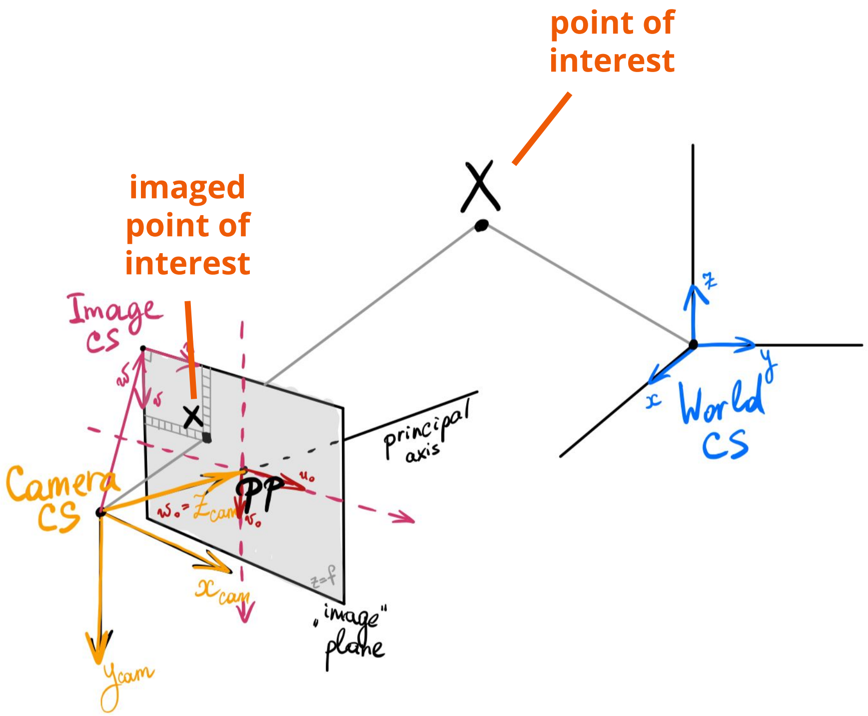
$$\mathbf{X}_{\text{cam}} = \begin{bmatrix} \mathtt{R} & -\mathtt{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} \mathbf{X}$$

2. *projected along ray from 3D to 2D*

$$\mathbf{x} = \mathtt{K}[\mathtt{I} \mid \mathbf{0}]\mathbf{X}_{\text{cam}}$$

$$\mathtt{K} = \begin{bmatrix} \alpha_x & & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix}$$

3. *distorted* $\boxed{\beta\tilde{\mathbf{x}} = f^d(\mathbf{x}, \lambda)}$

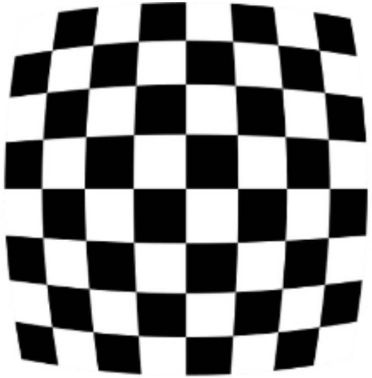


Radial Distortion Models: Brown

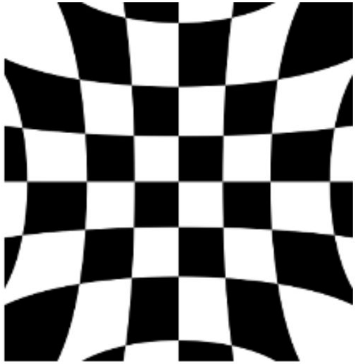
$$\boxed{\beta\tilde{\mathbf{x}} = f^d(\mathbf{x}, \lambda)}$$

$$\tilde{x} = x\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right)$$
$$\tilde{y} = y\left(1 + k_1 r^2 + k_2 r^4 + k_3 r^6\right)$$

where $r^2 = x^2 + y^2$



Barrel
typically $\boldsymbol{k_1 > 0}$



Pincushion
typically $\boldsymbol{k_1 < 0}$

* all points are distortion center subtracted

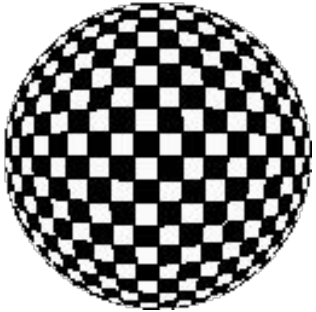
Radial Distortion Models: Fisheye

$$\boxed{\beta\tilde{\mathbf{x}} = f^d(\mathbf{x}, \lambda)}$$

$$\tilde{x} = (\theta_d / r)\, x$$
$$\tilde{y} = (\theta_d / r)\, y$$

where $r^2 = x^2 + y^2$

$$\theta = \operatorname{atan}(r)$$
$$\theta_d = \theta\left(1 + k_1\theta^2 + k_2\theta^4 + k_3\theta^6 + k_4\theta^8\right)$$



* all points are distortion center subtracted

Undistortion Model: Division

$$\boxed{\beta\tilde{\mathbf{x}} = f^d(\mathbf{x}, \lambda)}$$

One-parameter model

$$\gamma\mathbf{x} = f(\tilde{\mathbf{x}}, \lambda) = \left(\tilde{x}, \tilde{y}, 1 + \lambda(\tilde{x}^2 + \tilde{y}^2)\right)^\top$$

* all points are distortion center subtracted

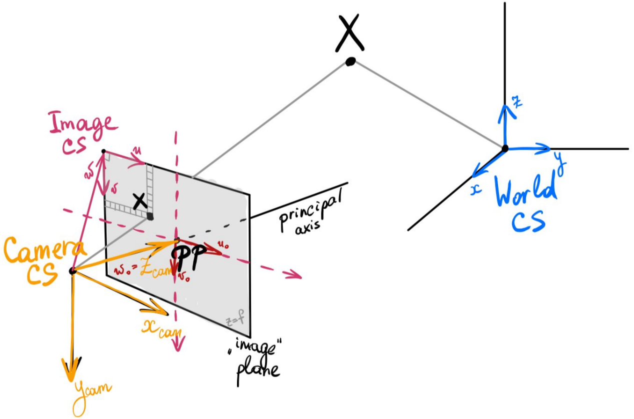
Camera Intrinsics

Camera Extrinsics

Modeling Image Formation

Camera Calibration

Camera Model from 6 imaged points



Given $\{\mathbf{x}_i \leftrightarrow \mathbf{X}_i\}_{i=1}^{6}$ **compute** $\mathtt{P}$

For each pair: $\alpha \mathbf{x}_i = \mathtt{Q}\mathbf{X}_i$

Q is a 3x4 matrix, determined up to a non-zero scale:

$$\mathtt{Q} = \varepsilon \mathtt{P}$$

q_i are 4x1 coordinate vectors

$$\mathtt{Q} = \begin{bmatrix} \mathrm{q}_1^\top \\ \mathrm{q}_2^\top \\ \mathrm{q}_3^\top \end{bmatrix} \qquad \begin{aligned} \alpha x &= \mathrm{q}_1^\top \mathbf{X} \\ \alpha y &= \mathrm{q}_2^\top \mathbf{X} \\ \alpha &= \mathrm{q}_3^\top \mathbf{X} \end{aligned} \longrightarrow \boxed{\begin{aligned} (\mathrm{q}_3^\top \mathbf{X})\, x &= \mathrm{q}_1^\top \mathbf{X} \\ (\mathrm{q}_3^\top \mathbf{X})\, y &= \mathrm{q}_2^\top \mathbf{X} \end{aligned}}$$

Camera Model from 6 imaged points

$$\boxed{\begin{aligned}(\mathrm{q}_3^\top \mathbf{X})\, x &= \mathrm{q}_1^\top \mathbf{X} \\ (\mathrm{q}_3^\top \mathbf{X})\, y &= \mathrm{q}_2^\top \mathbf{X}\end{aligned}}$$

Introduce vector of parameters (which are elements of **Q**)

$$\mathrm{q} = \begin{bmatrix} \mathrm{q}_1^\top & \mathrm{q}_2^\top & \mathrm{q}_3^\top \end{bmatrix}^\top$$

… and express the previous two equations in matrix form:

$$\begin{bmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -xX & -xY & -xZ & -x \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -yX & -yY & -yZ & -y \end{bmatrix} \mathrm{q} = 0$$

So each pair from $\{\mathbf{x}_i \leftrightarrow \mathbf{X}_i\}_{i=1}^{6}$ brings two rows into the matrix **M**.

Camera Model from 6 imaged points

$$\begin{aligned}(q_3^\top \mathbf{X})\, x &= q_1^\top \mathbf{X} \\ (q_3^\top \mathbf{X})\, y &= q_2^\top \mathbf{X}\end{aligned}$$

Introduce vector of parameters (which are elements of **Q**)

$$q = \begin{bmatrix} q_1^\top & q_2^\top & q_3^\top \end{bmatrix}^\top$$

… and express the above two equations in matrix form:

$$\begin{bmatrix} X & Y & Z & 1 & 0 & 0 & 0 & 0 & -xX & -xY & -xZ & -x \\ 0 & 0 & 0 & 0 & X & Y & Z & 1 & -yX & -yY & -yZ & -y \end{bmatrix} q = 0$$

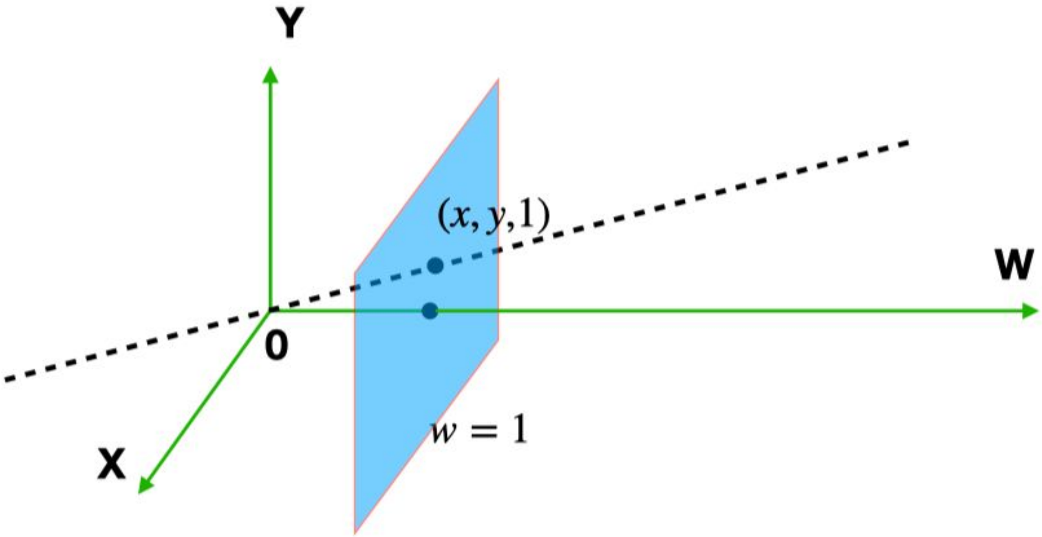
So each pair from $\{\mathbf{x}_i \leftrightarrow \mathbf{X}_i\}_{i=1}^{6}$ brings two rows into the matrix **M**.

6 pairs in general position → **11** linearly independent rows → a one-dimensional space of solutions.

If **Q** is a solution, then **τQ** is also a solution and both determine the same projection for any positive **τ**

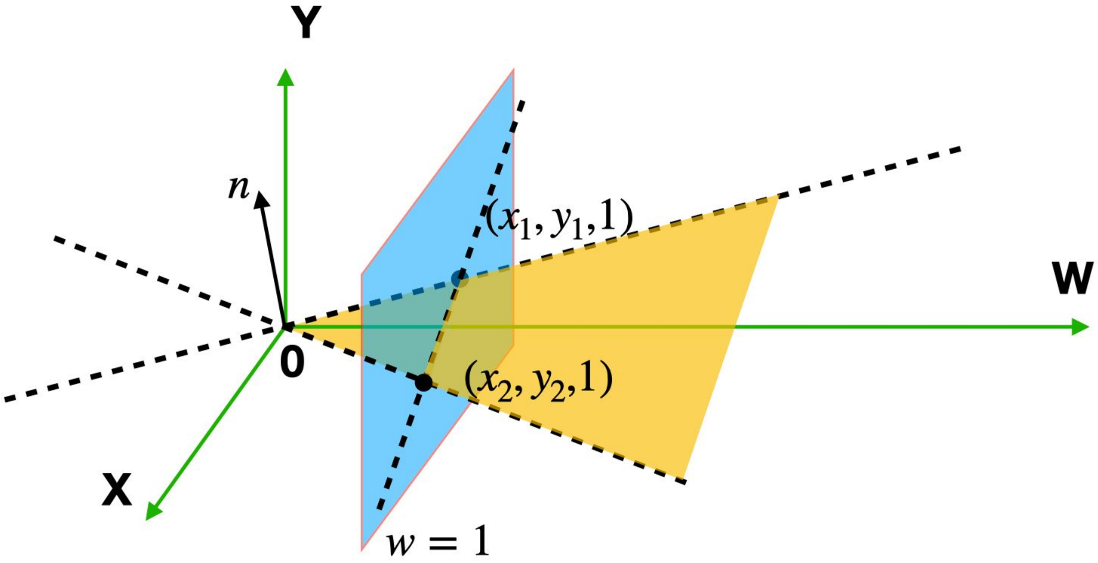
Homogeneous Coordinates for 2D Points & Lines

Every **point** in 2D **(x, y)** represents a ray in R^3 along which all points project the same, excluding **0**



Homogeneous Coordinates for 2D Points & Lines

Join of 2 points $y_1 = (x_1, y_1, 1)$ and $y_2 = (x_2, y_2, 1)$ plane in 3D — their vector product

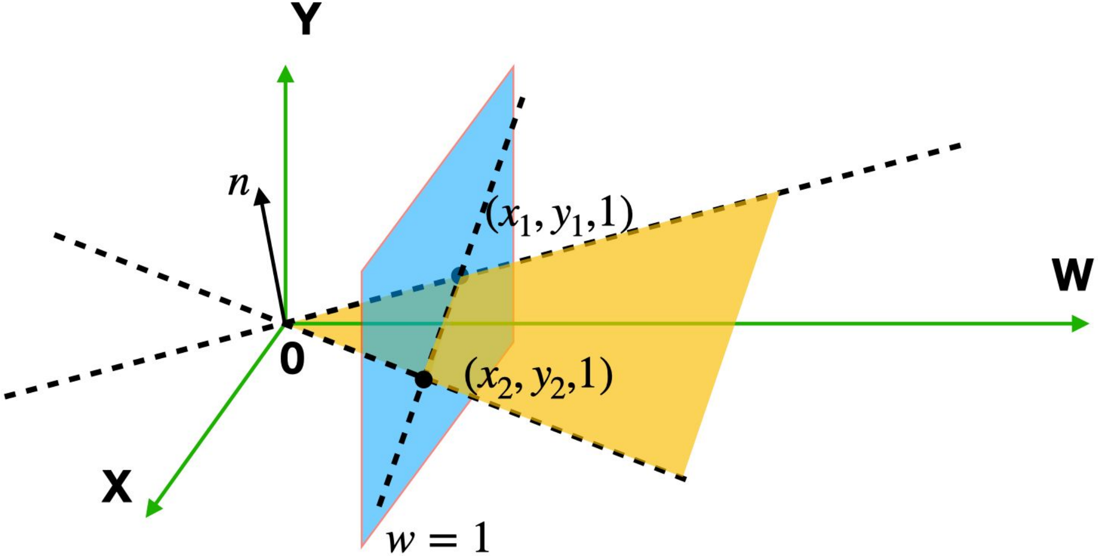


$$n \sim \begin{pmatrix} x_1 & y_1 & 1 \end{pmatrix} \times \begin{pmatrix} x_2 & y_2 & 1 \end{pmatrix}$$

*here and further, **x ~ y** / **x ≃ y** is for equivalence relation **αx = y**

Homogeneous Coordinates for 2D Points & Lines

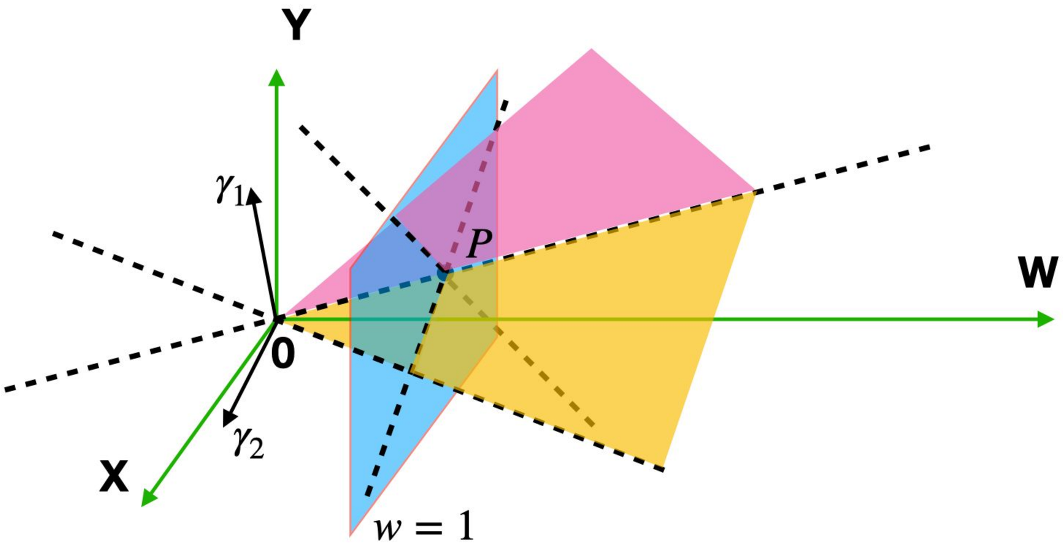
Every **line** in 2D **(x_1, y_1) – (x_2, y_2)** represents plane in 3D



$$n \sim \begin{pmatrix} x_1 & y_1 & 1 \end{pmatrix} \times \begin{pmatrix} x_2 & y_2 & 1 \end{pmatrix}$$

Homogeneous Coordinates for 2D Points & Lines

Intersection or Meet of 2 lines: $\gamma_1 = (a_1, b_1, c_1)$ & $\gamma_2 = (a_2, b_2, c_2)$ — their vector product



$$P \sim \gamma_1 \times \gamma_2 = (b_1c_2 - c_1b_2, c_1a_2 - a_1c_2, a_1b_2 - b_1a_2)$$

Homogeneous Coordinates

point	$\mathbf{p} = (X, Y, W)$
incidence	$\mathbf{p}^T\mathbf{u} = 0$
collinearity	$\lvert \mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3 \rvert = 0$
join of 2 points	$\mathbf{u} = \mathbf{p}_1 \times \mathbf{p}_2$

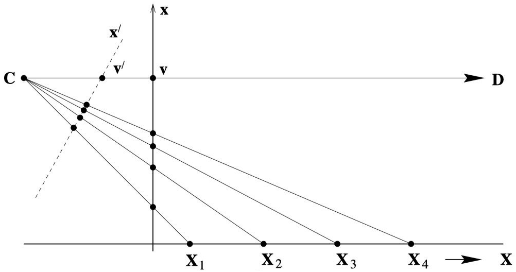
line	$\mathbf{u} = (a, b, c)$
incidence	$\mathbf{p}^T\mathbf{u} = 0$
concurrence	$\lvert \mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3 \rvert = 0$
intersection of 2 lines	$\mathbf{p} = \mathbf{u}_1 \times \mathbf{u}_2$

Points ⟵ Duality ⟶ **Lines**

Vanishing Points and Lines

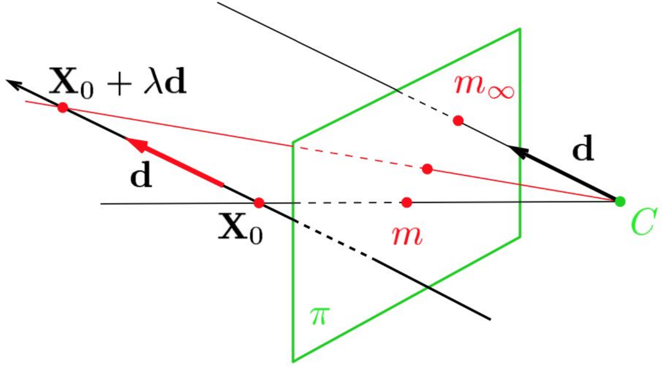
Point at Infinity or **Ideal Point**

a point with the last coordinate **zero**



Vanishing Point (VP) of the line

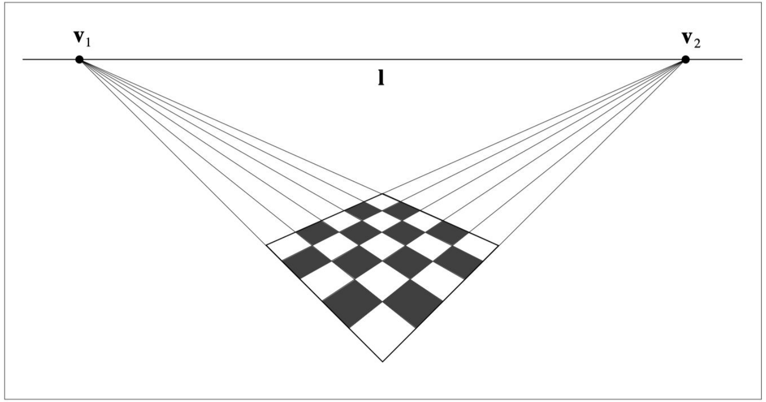
the limit of the projection of a point that moves along a line in space infinitely in one direction; it is an image of the **point at infinity.**



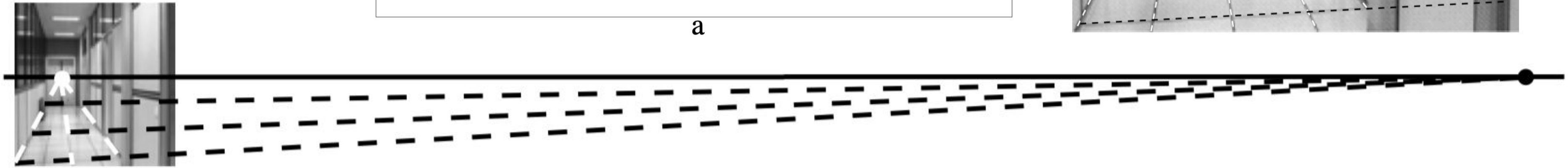
Vanishing Points and Lines

Vanishing Line (VL) of the plane

The line through the vanishing points of lines on the scene plane.



a

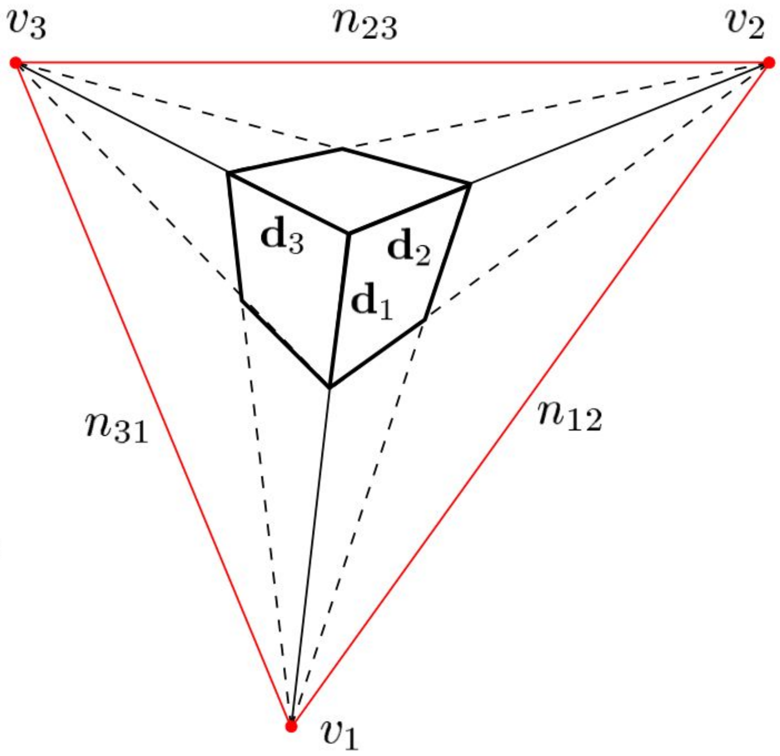


Camera Model from Vanishing Points

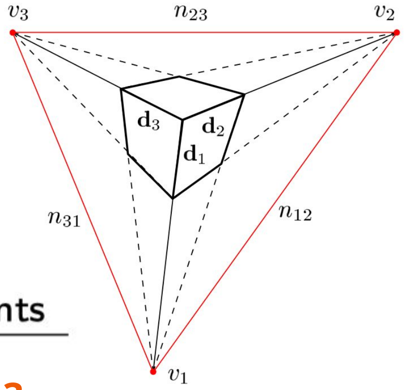
Given 3 finite VPs, compute **Camera Matrix K** and **principal point p**

$$\mathbf{d}_i \simeq \mathbf{Q}^{-1}\underline{\mathbf{v}}_i, \qquad\qquad i = 1, 2, 3$$

$$0 = \mathbf{d}_1^\top \mathbf{d}_2 = \underline{\mathbf{v}}_1^\top \mathbf{Q}^{-\top}\mathbf{Q}^{-1}\underline{\mathbf{v}}_2 = \underline{\mathbf{v}}_1^\top \underbrace{(\mathbf{K}\mathbf{K}^\top)^{-1}}_{\boldsymbol{\omega}\text{ (IAC)}}\underline{\mathbf{v}}_2$$



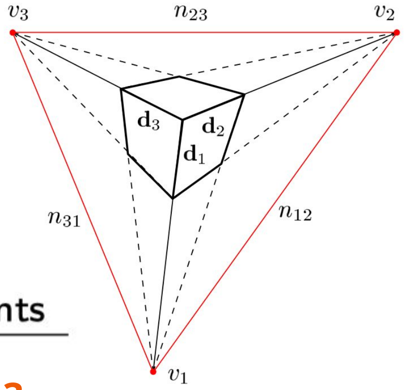
Camera Model from Vanishing Points



configuration	equation	# constraints
orthogonal v.p.	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1 ×3
orthogonal raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1
unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1

5 linear equations
for **5** parameters

Camera Model from Vanishing Points

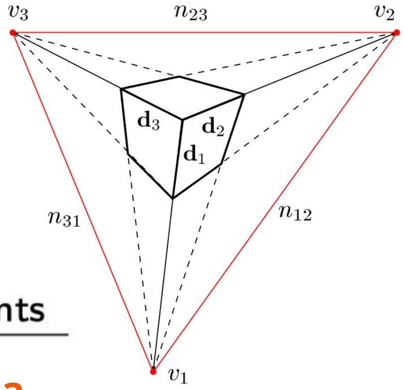


configuration	equation	# constraints
orthogonal v.p.	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1 ×3
orthogonal raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1
unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1

5 linear equations for **5** parameters

$$\boxed{\boldsymbol{\omega} \simeq \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ -u_0 & -v_0 & f^2 + u_0^2 + v_0^2 \end{bmatrix}}$$

Camera Model from Vanishing Points



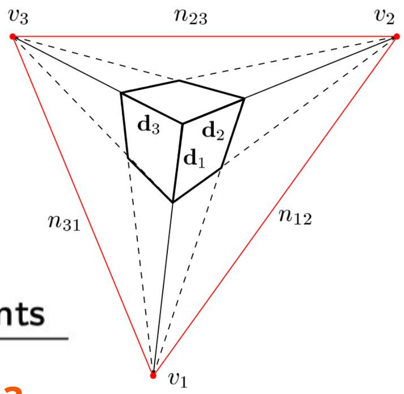
configuration	equation	# constraints
orthogonal v.p.	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1 ×3
orthogonal raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1
unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1

5 linear equations for **5** parameters

$$\boxed{\boldsymbol{\omega} \simeq \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ -u_0 & -v_0 & f^2 + u_0^2 + v_0^2 \end{bmatrix}}$$

We get **K** from $\boldsymbol{\omega}^{-1} = \mathbf{K}\mathbf{K}^\mathsf{T}$ by Choleski decomposition.

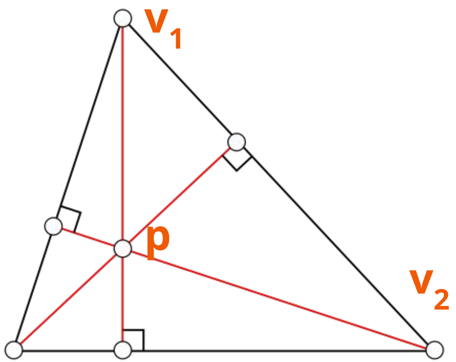
Camera Model from Vanishing Points



configuration	equation	# constraints	
orthogonal v.p.	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1	×3
orthogonal raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1	
unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1	

5 linear equations for **5** parameters

$$\boxed{\boldsymbol{\omega} \simeq \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ -u_0 & -v_0 & f^2 + u_0^2 + v_0^2 \end{bmatrix}}$$



We get **K** from **$\boldsymbol{\omega}^{-1} = \mathbf{K}\mathbf{K}^T$** by Choleski decomposition.
The principal point **p** is computed as the triangle orthocenter.

References, again

[1] Pajdla, Tomas. Elements of geometry for computer vision. FEE CTU, 2013

[2] Hartley, Richard, and Andrew Zisserman. Multiple view geometry in computer vision. Cambridge university press, 2003

[3] Šára, Radim. TDV – 3D Computer Vision, Winter 2017

[4] Gkioulekas, Ioannis. Computational Photography, Fall 2019

[5] Kutulakos, Kyros. Computer Graphics, Fall 2010