NUS National University of Singapore | School of Computing

CS4277 / CS5477
3D Computer Vision

Lecture 4:
Camera models and calibration

Assoc. Prof. Lee Gim Hee

AY 2022/23

Semester 2

Course Schedule

Week	Date	Topic	Assignments
1	11 Jan	2D and 1D projective geometry	**Assignment 0:** Getting started with Python (Ungraded)
2	18 Jan	3D projective geometry, Circular points and Absolute conic	
3	25 Jan	Rigid body motion and Robust homography estimation	
4	01 Feb	Camera models and calibration	**Assignment 1:** Metric rectification and robust homography (10%) **Due:** 2359hrs, 07 Feb
5	08 Feb	Single view metrology	**Assignment 2**: Affine 3D measurement from vanishing line and point (10%) **Due:** 2359hrs, 14 Feb
6	15 Feb	The Fundamental and Essential matrices	
-	22 Feb	Semester Break	No lecture
7	**01 Mar**	**Mid-term Quiz (20%)**	**In-person Quiz (LT 15, 1900hrs – 2000hrs)**
8	08 Mar	Absolute pose estimation from points or lines	
9	15 Mar	Three-view geometry from points and/or lines	
10	22 Mar	Structure-from-Motion (SfM) and bundle adjustment	**Assignment 3:** SfM and Bundle adjustment (10%) **Due:** 2359hrs, 28 Mar
11	29 Mar	Two-view and multi-view stereo	**Assignment 4:** Dense 3D model from multi-view stereo (10%) **Due:** 2359hrs, 04 Apr
12	05 Apr	3D Point Cloud Processing	
13	12 Apr	Neural Field Representations	

Final Exam: 03 MAY 2023

Learning Outcomes

- Students should be able to:

1. Describe camera projection with the pinhole model.
2. Identify the camera centre, principal planes, principal point, and principal axis from the projection matrix.
3. Use the projection matrix to get the forward and backward projection of a point.
4. Explain the properties of an affine camera.
5. Do calibration to find the intrinsic and extrinsic values of a projective camera.

Acknowledgements

- A lot of slides and content of this lecture are adopted from:

1. R. Hartley, and Andrew Zisserman: "Multiple view geometry in computer vision", Chapter 6.

2. Y. Ma, S. Soatto, J. Kosecka, S. S. Sastry, " An invitation to 3-D vision", Chapter 3.

3. Z. Y. Zhang, "A Flexible New Technique for Camera Calibration", TPAMI 2000.

What is a Camera?

- A camera is a mapping between the 3D world (object space) and a 2D image.

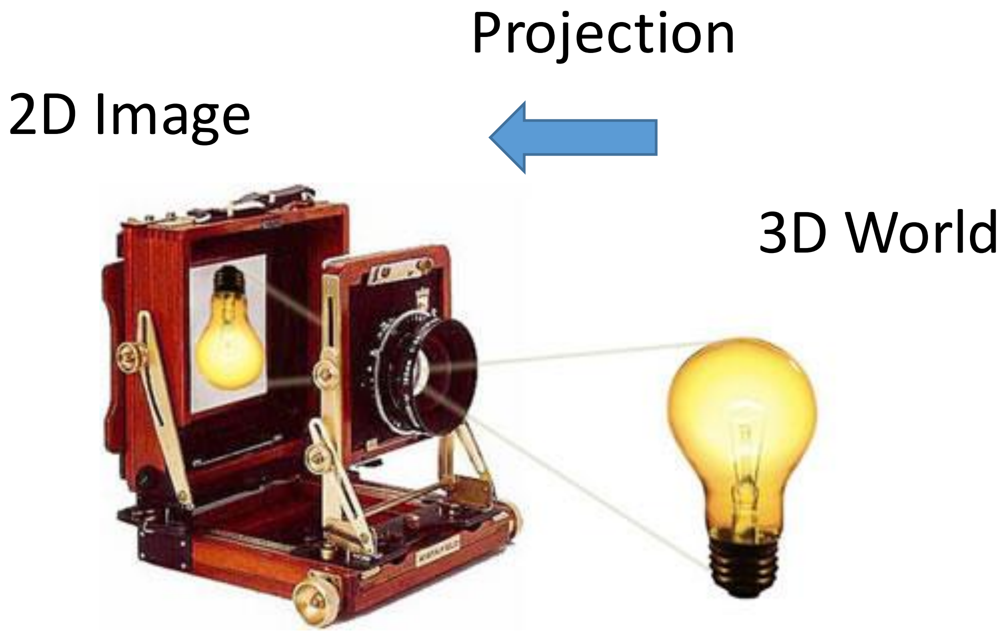


Image source: http://www.shortcourses.com/guide/guide1-3.html

Camera Models

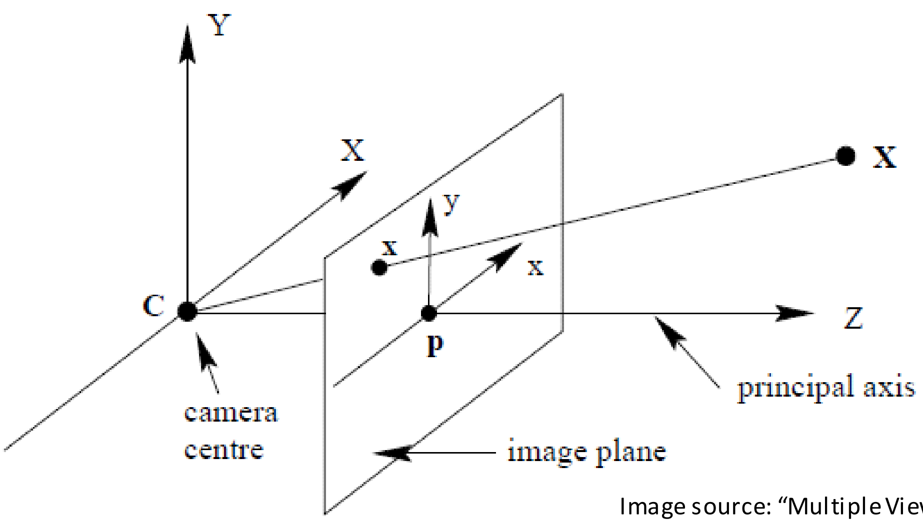
- In this lecture, we will look at camera models with central projection.
- Camera models with central projection fall into two major classes: those with a finite centre, and those with a centre "at infinity".
- We will see more details of the projective camera with a finite centre and affine camera with a centre "at infinity".

The Basic Pinhole Model

- The projective camera is based on the basic pinhole camera.

- Let the centre of projection be the origin of a Euclidean coordinate system.

- And consider the plane Z = f as the image plane or focal plane.



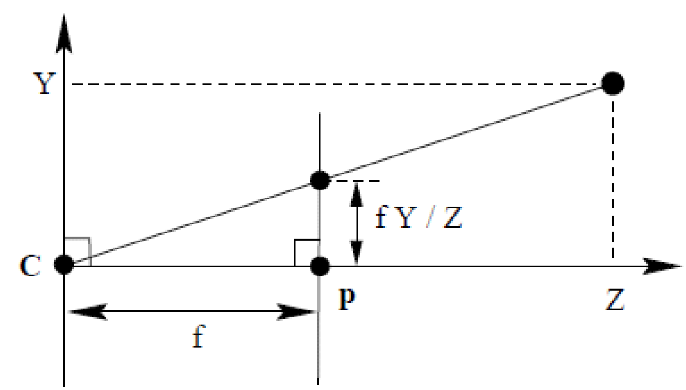


Image source: “Multiple View Geometry in Computer Vision”, Richard Hartley and Andrew Zisserman

The Basic Pinhole Model

- Using similar triangle, we can see that the point $(X, Y, Z)^\top$ is mapped to the point $(fX/Z, fY/Z, f)^\top$ on the image plane.

- Ignoring the final coordinate, we get the central projection mapping from world to image coordinates :

$$(X, Y, Z)^\top \mapsto (fX/Z, fY/Z)^\top, \text{ i.e. } \mathbb{R}^3 \mapsto \mathbb{R}^2$$

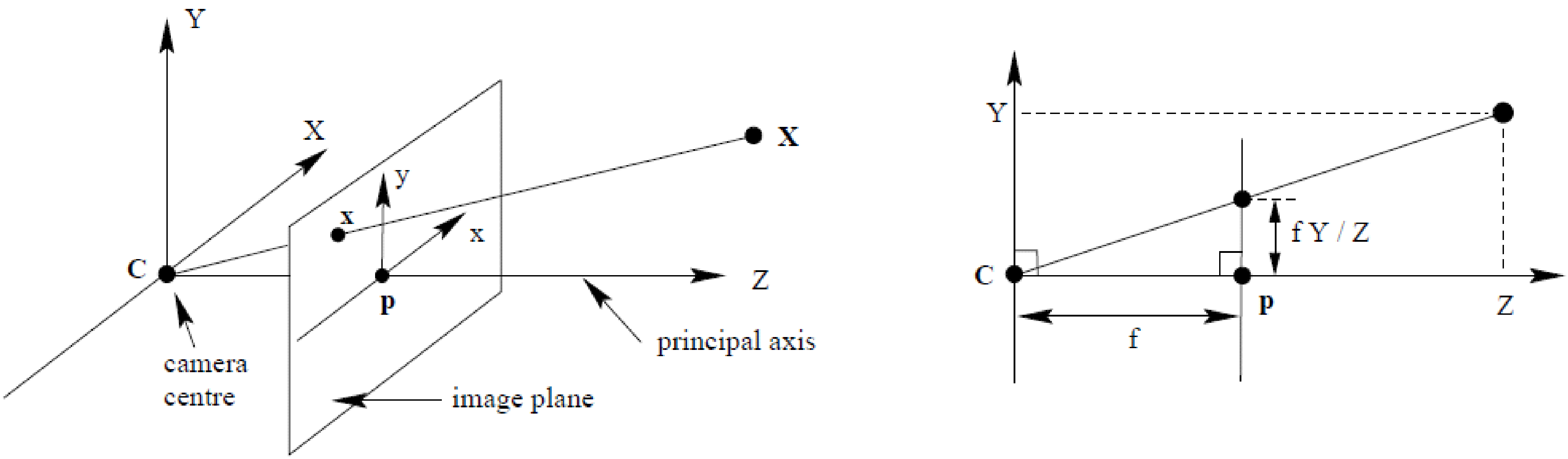
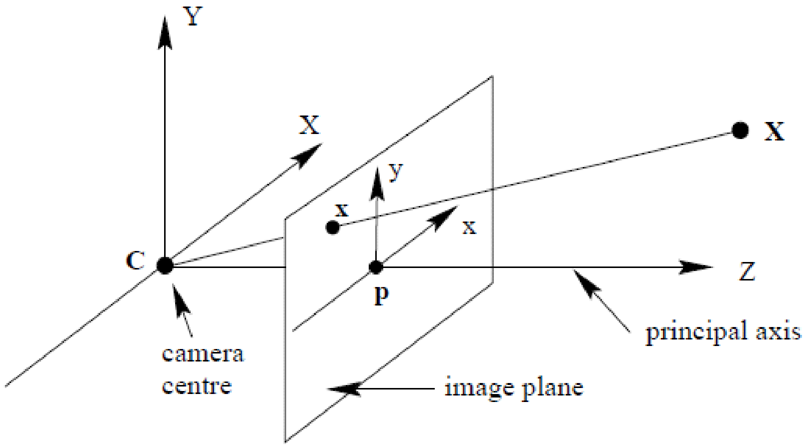


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

The Basic Pinhole Model

- Camera Centre or Optical Centre: Centre of projection.
- Principal Axis or Principal Ray: Line from camera centre perpendicular to image plane.
- Principal Point: Point where principal axis meets the image plane.
- Principal Plane: Plane through the camera centre parallel to the image plane.



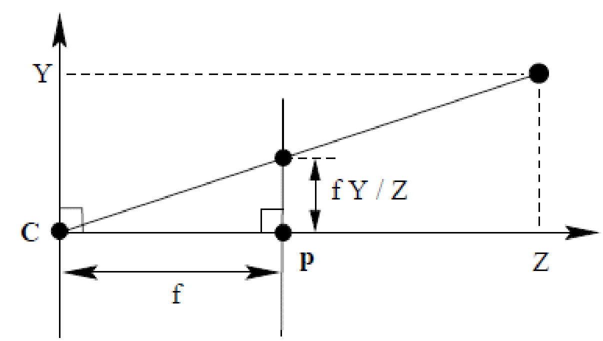


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Central Projection Using Homogeneous Coordinates

- The world and image points becomes a linear mapping in homogeneous coordinates:

$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX \\ fY \\ Z \end{pmatrix} = \underbrace{\begin{bmatrix} f & & & 0 \\ & f & & 0 \\ & & 1 & 0 \end{bmatrix}}_{\mathrm{diag}(f,f,1)[\mathrm{I} \mid \mathbf{0}]} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}.$$

- Letting $\mathrm{P} = \mathrm{diag}(f, f, 1)[\mathrm{I} \mid 0]$, $\mathbf{x} = (f\mathrm{X}, f\mathrm{Y}, \mathrm{Z})^\top$ and $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)^\top$, we get:

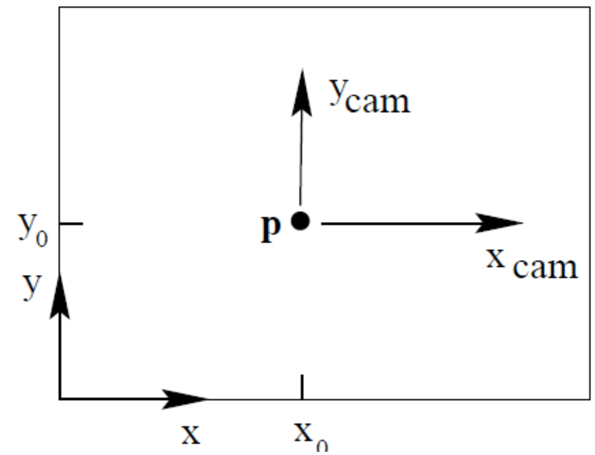
$$\mathbf{x} = \mathrm{P}\mathbf{X},$$

- P is the 3x4 homogeneous camera projection matrix.

Principal Point Offset

- In practice, the origin of coordinates in the image plane might not be at the principal point, i.e.

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



- $(p_x, p_y)^\mathsf{T}$ are the coordinates of the principal point.

- Expressing in homogeneous coordinates, we get:

$$\begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} f\mathrm{X} + \mathrm{Z}p_x \\ f\mathrm{Y} + \mathrm{Z}p_y \\ \mathrm{Z} \end{pmatrix} = \begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix}.$$

Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Camera Calibration Matrix

- Now, writing:

$$\mathtt{K} = \begin{bmatrix} f & & p_x \\ & f & p_y \\ & & 1 \end{bmatrix},$$

- We can rewrite

$$\begin{pmatrix} f\mathrm{X} + \mathrm{Z}p_x \\ f\mathrm{Y} + \mathrm{Z}p_y \\ \mathrm{Z} \end{pmatrix} = \begin{bmatrix} f & & p_x & 0 \\ & f & p_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} \mathrm{X} \\ \mathrm{Y} \\ \mathrm{Z} \\ 1 \end{pmatrix} \quad \text{as} \quad \mathbf{x} = \mathtt{K}[\mathtt{I} \mid \mathbf{0}]\mathbf{X}_{\mathrm{cam}}.$$

- The matrix K is called the camera calibration matrix.

Camera Rotation and Translation

- $\mathbf{X}_{\text{cam}} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)^{\mathsf{T}}$ is expressed in the camera coordinate frame, where the camera is at the origin and principal axis points in the z-axis.
- In general, 3D points are expressed in a different Euclidean coordinate frame, known as the world coordinate frame.
- The two frames are related via a rigid transformation (R, t).

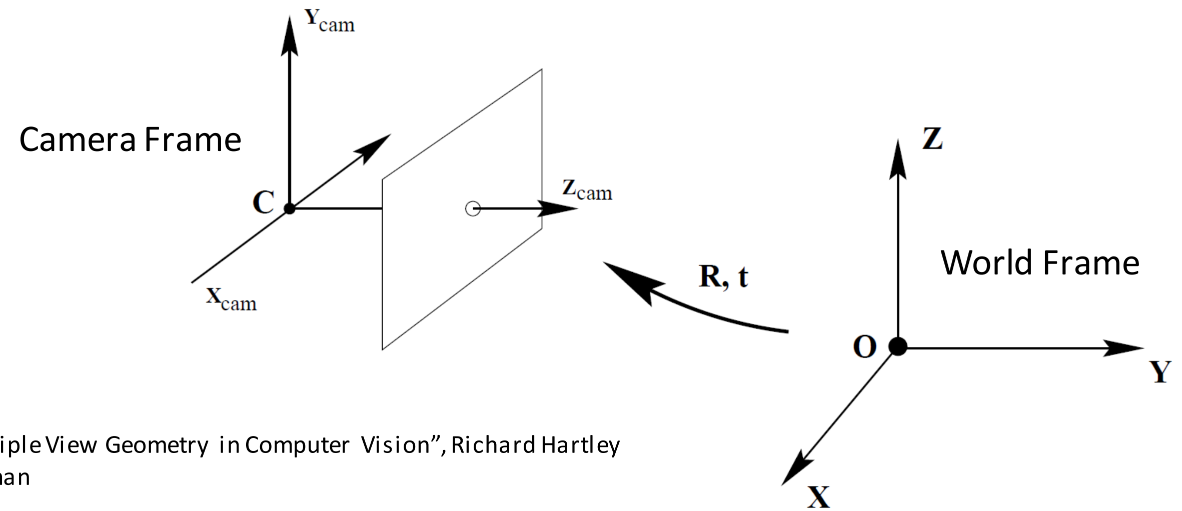
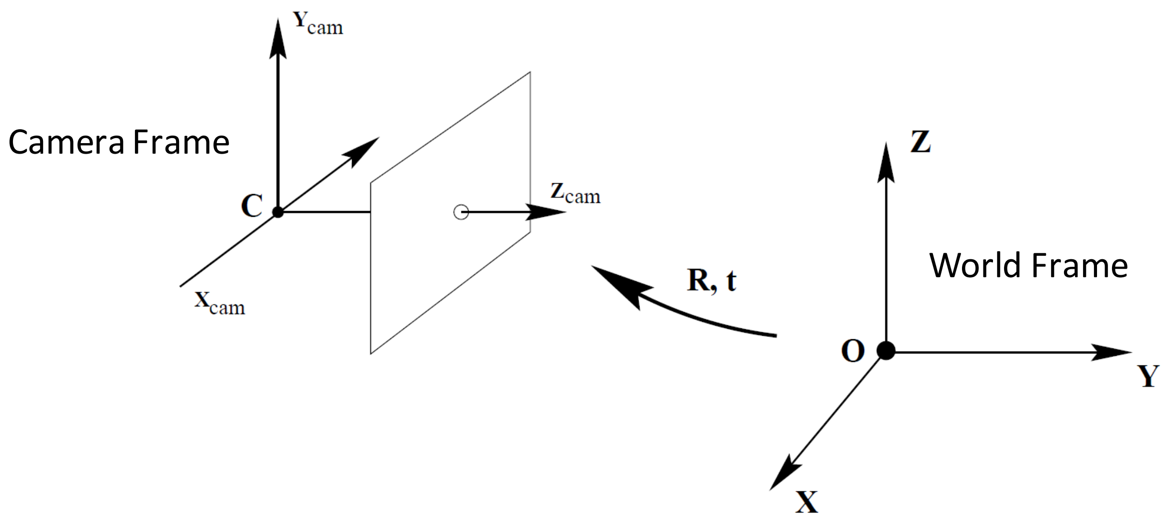


Image source: “Multiple View Geometry in Computer Vision”, Richard Hartley and Andrew Zisserman

Camera Rotation and Translation

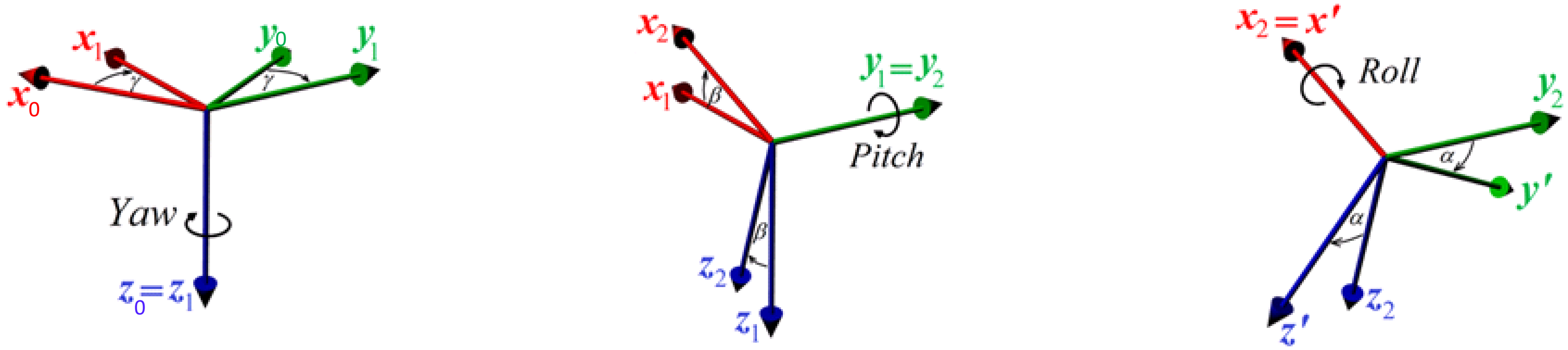


- Denoting the coordinates of the camera centre in the world frame as $\tilde{C}$, we write:

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

Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Euler Angles to Rotation Matrix



$$R_z(\gamma) = R_1^0 = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\Rightarrow X_0 = R_1^0 X_1$$

$$R_y(\beta) = R_2^1 = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

$$\Rightarrow X_1 = R_2^1 X_2$$

$$R_x(\alpha) = R_3^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$\Rightarrow X_3 = R_2^3 X_2$$

$$R_3^0 = R_1^0 R_2^1 R_3^2 = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$

$$\Rightarrow X_0 = R_3^0 X_3$$

Image Source: http://www.mdpi.com/1424-8220/15/3/7016/htm

Properties of Rotation Matrix

Rotation matrices are:

- Square matrices 2x2 (2 dimensional) or 3x3 (3 dimensional) with real entries.
- Orthonormal matrices with the following properties:

1. $\det(R) = \begin{cases} +1, & \text{Right-Hand coordinate frame} \\ -1, & \text{Left-Hand coordinate frame} \end{cases}$
2. $\mathrm{R}^{\top} = \mathrm{R}^{-1}$,
3. $r_i \times r_j = r_k$, (third column is the cross-product of the other two columns)
4. ${r_i}^{\top} r_j = 0$, where r_i is column i of the rotation matrix
5. $\|r_1\| = \|r_2\| = \|r_3\| = 1.$

The Basic Pinhole Model

- Putting $\mathbf{X}_{cam}$ back into $\mathbf{x} = \mathrm{K}[\mathrm{I} \mid \mathbf{0}]\mathbf{X}_{\mathrm{cam}}$, we get the general mapping of a pinhole camera:

$$\mathbf{x} = \mathrm{KR}[\mathrm{I} \mid -\widetilde{\mathbf{C}}]\mathbf{X}$$

where $\mathbf{X}$ is now in a world coordinate frame.

- We write the camera projection matrix as:

$$\mathrm{P} = \mathrm{KR}[\mathrm{I} \mid -\widetilde{\mathbf{C}}],$$

- P has 9 degrees of freedom: 3 for K (the elements f, p_x, p_y), 3 for R, and 3 for $\tilde{\mathbf{C}}$.

The Basic Pinhole Model

- The parameters contained in K are called the internal camera parameters, or the intrinsic of the camera.

- The parameters of R and $\tilde{\mathbf{C}}$ are called the external parameters or the extrinsic of the camera.

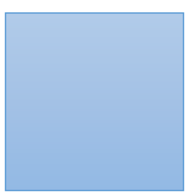
- It is often more convenient to represent the extrinsics in terms of (R, t):

$$\mathrm{P} = \mathrm{K}[\mathrm{R} \mid \mathbf{t}]$$

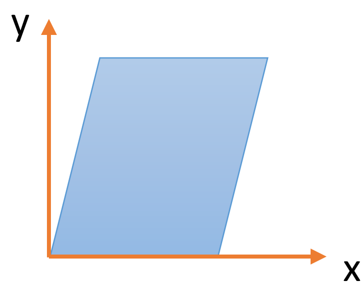
By rewriting $\mathbf{t} = -\mathrm{R}\widetilde{\mathrm{C}}$.

Non-Square and Skewed Pixels

- Same focal length f for both x and y directions in camera calibration matrix ⇒ Square pixel assumption.
- Pixels might be non-square and skewed in real cameras.
- More accurate to have:
    1. Different focal lengths for individual directions, i.e., f_x and f_y.
    2. Skew parameter, i.e., s in the x direction.



Square Pixel



Non-square pixel
with x-skew

Camera Intrinsic Matrix:

$$\mathrm{K} = \begin{bmatrix} f_x & s & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix}$$

Camera Intrinsic and Extrinsic

- In general, the camera projection matrix P has 11 degrees of freedom:

$$\mathrm{P} = \mathrm{K}[\mathrm{R} \quad \mathrm{t}]$$

Component	# DOF	Elements	Known As
K	5	f_x, f_y, s, p_x, p_y	Intrinsic Parameters
R	3	α, β, γ	Extrinsic Parameters
$\tilde{\mathrm{C}}$ or t	3	(C_x, C_y, C_z) or (t_x, t_y, t_z)	

Total: 11 DOF

Finite Projective Cameras

- The set of camera matrices of finite projective cameras

$$\mathtt{P} = \mathtt{KR}[\mathtt{I} \mid -\widetilde{\mathbf{C}}]$$

is identical with the set of homogeneous 3×4 matrices, i.e.

$$\mathtt{P} = \mathtt{M}[\mathtt{I} \mid \mathtt{M}^{-1}\mathbf{p}_4] = \mathtt{KR}[\mathtt{I} \mid -\widetilde{\mathbf{C}}]$$

for which the left-hand 3×3 submatrix M is non-singular.

- $\mathbf{p}_4$ is the last column of P.

Finite Projective Cameras: Camera Anatomy

Camera centre:

- The rank 3 matrix P has a 1-dimensional right null-space; and this 4-vector null-space is the camera centre $\mathbf{C}$, i.e.

$$\mathrm{P}\mathbf{C} = \mathbf{0},$$

which is an undefined image point $(0, 0, 0)^\top$.

Finite Projective Cameras: Camera Anatomy

Sketch of Proof:

- Consider the line containing $\mathbf{C}$ and any other point $\mathbf{A}$ in 3-space,

$$\mathbf{X}(\lambda) = \lambda\mathbf{A} + (1-\lambda)\mathbf{C} \ .$$

- Under the mapping $\mathbf{x} = \mathrm{P}\mathbf{X}$ points on this line are projected to

$$\mathbf{x} = \mathrm{P}\mathbf{X}(\lambda) = \lambda\mathrm{P}\mathbf{A} + (1-\lambda)\overbrace{\mathrm{P}\mathbf{C}}^{=0} = \lambda\mathrm{P}\mathbf{A}$$

- i.e. all points $\mathbf{X}(\lambda)$ are mapped to the same image point $\mathrm{P}\mathbf{A}$, hence, the line must be a ray through the camera centre.

$\square$

Finite Projective Cameras: Camera Anatomy

Column vectors:

- With the notation that the columns of P are $\mathbf{p}_i$, $i = 1, \ldots, 4$, then $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3$ are the vanishing points of the world coordinate X, Y and Z axes, respectively.

- The column $\mathbf{p}_4$ is the image of the world origin.

Example:

The *x*-axis has direction $\mathbf{D} = (1, 0, 0, 0)^\top$, which is imaged at $\mathbf{p}_1 = \mathrm{P}\mathbf{D}$.

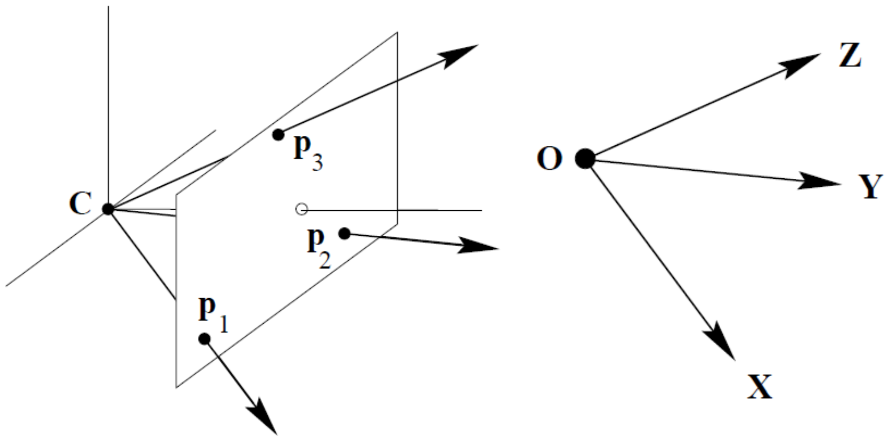


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Finite Projective Cameras: Camera Anatomy

Row vectors:

- The rows of the projective camera are 4-vectors which may be interpreted geometrically as particular world planes, i.e.

$$\mathrm{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} = \begin{bmatrix} \mathbf{P}^{1\mathsf{T}} \\ \mathbf{P}^{2\mathsf{T}} \\ \mathbf{P}^{3\mathsf{T}} \end{bmatrix}.$$

Finite Projective Cameras: Camera Anatomy

1. **Principal plane:**

- The principal plane is the plane through the camera centre parallel to the image plane.

Sketch of Proof:

- It consists of the set of points $\mathbf{X}$ imaged on the line at infinity of the image, i.e. $\mathrm{P}\mathbf{X} = (\mathrm{x}, \mathrm{y}, 0)^{\top}$.

- Thus, a point lies on the principal plane of the camera if and only if $\mathrm{P}^{3\top}\mathbf{X} = 0$, hence $\mathrm{P}^{3\top}$ is the principal plane. $\square$

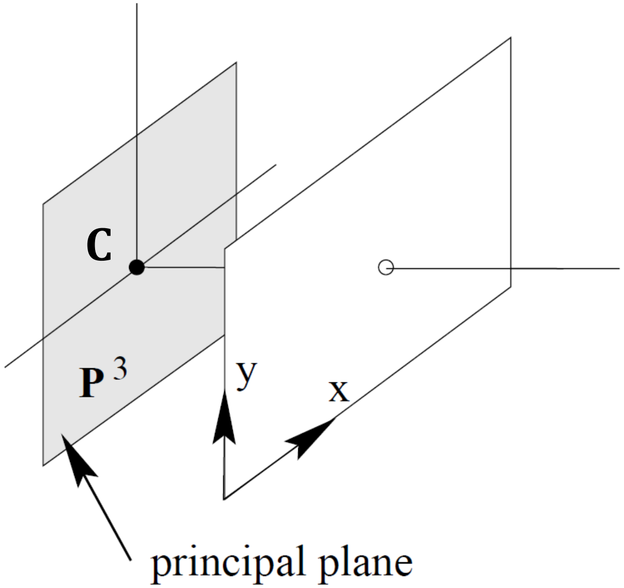


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Finite Projective Cameras: Camera Anatomy

2. **Axis plane:**

- $\mathbf{P}^1$ is defined by the camera centre $\mathbf{C}$ and the line $\mathrm{x} = 0$ in the image. Similarly, $\mathbf{P}^2$ is defined by the camera centre and the line $\mathrm{y} = 0$.

Sketch of Proof:

- A set of points $\mathbf{X}$ on the plane $\mathbf{P}^2$ satisfy $\mathbf{P}^{2\top}\mathbf{X} = 0$, hence $\mathrm{P}\mathbf{X} = (x, 0, w)^\top$ which are points on the line $\mathrm{y} = 0$.
- It follows from $\mathrm{P}\mathbf{C} = \mathbf{0}$ that $\mathbf{P}^{2\top}\mathbf{C} = 0$ and so $\mathbf{C}$ also lies on the plane $\mathbf{P}^2$.
- Similar result can be shown for $\mathbf{P}^1$. $\square$

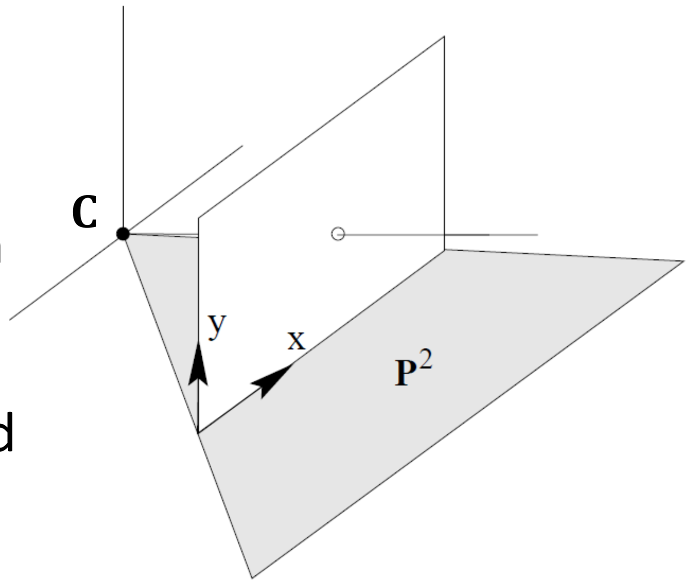


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Finite Projective Cameras: Camera Anatomy

The principal point:

- The principal axis is the line passing through the camera centre **C**, with direction perpendicular to the principal plane $\mathbf{P}^3$.
- The axis intersects the image plane at the principal point x_0.

Finite Projective Cameras: Camera Anatomy

Remarks:

- The point $\widehat{\mathbf{P}}^3 = (p_{31}, p_{32}, p_{33}, 0)^\top = (\mathbf{m}^3, 0)^\top$ denotes the direction of the normal vector (principal axis) of the principal plane.

- This point projects onto the image as the principal point, i.e., $\mathrm{x}_0 = \mathrm{P}\widehat{\mathbf{P}}^3$ which can be written as:

$\mathbf{x}_0 = \mathrm{M}\mathbf{m}^3$, where $\mathrm{P} = [\mathrm{M} \mid \mathbf{p}_4]$ and $\mathbf{m}^{3\top}$

is the third row of M.

Finite Projective Cameras: Camera Anatomy

The principal axis vector:

- Although any point $\mathbf{X}$ not on the principal plane may be mapped to an image point according to $\mathbf{x} = \mathrm{P}\mathbf{X}$.
- In reality, only half the points in space, those that lie in front of the camera, may be seen in an image.
- $\mathbf{v} = \det(\mathrm{M})\,\mathbf{m}^3$ is a vector in the direction of the principal axis, directed towards the front of the camera.

Finite Projective Cameras: Camera Anatomy

Remarks:

- We have seen earlier that $\mathbf{m}^3$ is the principal axis obtained from $\mathtt{P} = [\mathtt{M} \mid \mathbf{p}_4]$.
- However, P is only defined up to sign. This leaves an ambiguity on whether $\mathbf{m}^3$ or $-\mathbf{m}^3$ points in the +ve direction.
- The direction of the principal axis can be obtained from $\det(\mathrm{M})$, which is the signed area equivalent.

Summary of the properties of $\mathbf{P}$

Camera centre. The camera centre is the 1-dimensional right null-space $\mathbf{C}$ of $\mathtt{P}$, i.e. $\mathtt{P}\mathbf{C} = \mathbf{0}$.

- $\diamond$ **Finite camera** ($\mathtt{M}$ is not singular) $\mathbf{C} = \begin{pmatrix} -\mathtt{M}^{-1}\mathbf{p}_4 \\ 1 \end{pmatrix}$
- $\diamond$ **Camera at infinity** ($\mathtt{M}$ is singular) $\mathbf{C} = \begin{pmatrix} \mathbf{d} \\ 0 \end{pmatrix}$ where $\mathbf{d}$ is the null 3-vector of $\mathtt{M}$, i.e. $\mathtt{M}\mathbf{d} = \mathbf{0}$.

Column points. For $i = 1, \ldots, 3$, the column vectors $\mathbf{p}_i$ are vanishing points in the image corresponding to the X, Y and Z axes respectively. Column $\mathbf{p}_4$ is the image of the coordinate origin.

Principal plane. The principal plane of the camera is $\mathbf{P}^3$, the last row of $\mathtt{P}$.

Axis planes. The planes $\mathbf{P}^1$ and $\mathbf{P}^2$ (the first and second rows of $\mathtt{P}$) represent planes in space through the camera centre, corresponding to points that map to the image lines $x = 0$ and $y = 0$ respectively.

Principal point. The image point $\mathbf{x}_0 = \mathtt{M}\mathbf{m}^3$ is the principal point of the camera, where $\mathbf{m}^{3\mathsf{T}}$ is the third row of $\mathtt{M}$.

Principal ray. The principal ray (axis) of the camera is the ray passing through the camera centre $\mathbf{C}$ with direction vector $\mathbf{m}^{3\mathsf{T}}$. The principal axis vector $\mathbf{v} = \det(\mathtt{M})\mathbf{m}^3$ is directed towards the front of the camera.

Table source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Action of a Projective Camera on Points

Forward projection:

- As seen, a general projective camera maps a point in space $\mathbf{X}$ to an image point according to the mapping $\mathbf{x} = \mathrm{P}\mathbf{X}$.
- Points $\mathbf{D} = (\mathbf{d}^\top, 0)^\top$ on the plane at infinity represent vanishing points; such points are mapped to:

$$\mathbf{x} = \mathrm{P}\mathbf{D} = [\mathrm{M} \mid \mathbf{p}_4]\mathbf{D} = \mathrm{M}\mathbf{d}$$

- Thus, are only affected by M, i.e., the first 3×3 submatrix of P.

Action of a Projective Camera on Points

Back-projection of points to rays:

- The ray is the line

$$\mathbf{X}(\lambda) = \mathrm{P}^{+}\mathbf{x} + \lambda\mathbf{C}$$

formed by the join of two points:

1. The camera center $\mathbf{C}$ (where $\mathrm{P}\mathbf{C} = \mathbf{0}$).
2. The point $\mathrm{P}^{+}\mathbf{x}$, where $\mathrm{P}^{+} = \mathrm{P}^{\top}(\mathrm{P}\mathrm{P}^{\top})^{-1}$ is the pseudo-inverse of P.

Action of a Projective Camera on Points

Back-projection of points to rays:

- For a finite camera where $\mathtt{M}^{-1}$ exists, we can write the line as:

$$\mathbf{X}(\mu) = \mu \begin{pmatrix} \mathtt{M}^{-1}\mathbf{x} \\ 0 \end{pmatrix} + \begin{pmatrix} -\mathtt{M}^{-1}\mathbf{p}_4 \\ 1 \end{pmatrix} = \begin{pmatrix} \mathtt{M}^{-1}(\mu\mathbf{x} - \mathbf{p}_4) \\ 1 \end{pmatrix},$$

where

- $\widetilde{\mathbf{C}} = -\mathtt{M}^{-1}\mathbf{p}_4$ is the inhomogenous camera center.
- $\mathtt{M}^{-1}\mathbf{x}$ is the ideal point $\mathbf{D} = ((\mathtt{M}^{-1}\mathbf{x})^\mathsf{T}, 0)^\mathsf{T}$ from the intersection of backprojected image point **x** and $\boldsymbol{\pi}_\infty$.

Action of a Projective Camera on Points

Depth of points:

- Let $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, \mathrm{T})^{\mathsf{T}}$ be a 3D point and $\mathrm{P} = [\mathrm{M} \mid \mathbf{p}_4]$ be a camera matrix for a finite camera. Suppose $\mathrm{P}(\mathrm{X}, \mathrm{Y}, \mathrm{Z}, \mathrm{T})^{\mathsf{T}} = w(x, y, 1)^{\mathsf{T}}$, then

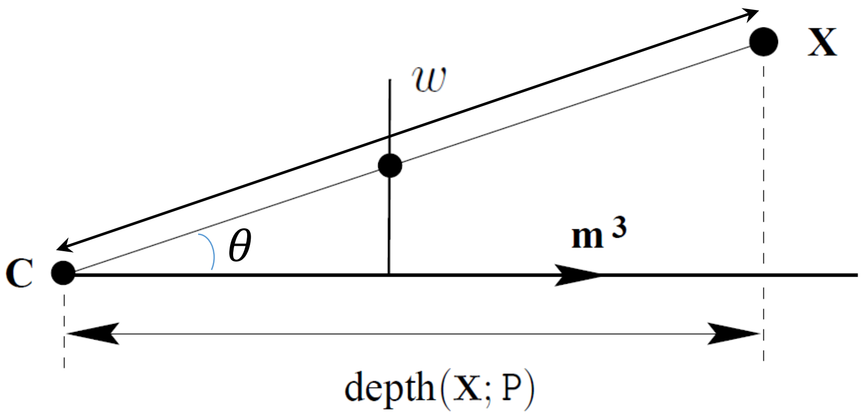
$$\mathrm{depth}(\mathbf{X}; \mathrm{P}) = \frac{\mathrm{sign}(\det \mathtt{M})w}{\mathrm{T}\|\mathbf{m}^3\|},$$

  is the depth of the point $\mathbf{X}$ in front of the principal plane of the camera.

- This formula is an effective way to determine if a point $\mathbf{X}$ is in front of the camera.

Action of a Projective Camera on Points

Proof:



3D Point: $\mathbf{X} = (\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 1)^\mathsf{T} = (\tilde{\mathrm{X}}^\mathsf{T}, 1)^\mathsf{T}$

Camera Centre: $\mathbf{C} = (\tilde{\mathrm{C}}, 1)^\mathsf{T}$

Image point: $\mathbf{x} = w(x, y, 1)^\mathsf{T} = \mathrm{P}\mathbf{X}$

$$w = \mathbf{P}^{3\mathsf{T}}\mathbf{X} = \mathbf{P}^{3\mathsf{T}}(\mathbf{X} - \mathbf{C}) = \overbrace{\mathbf{m}^{3\mathsf{T}}(\tilde{\mathrm{X}} - \tilde{\mathrm{C}})}^{\text{Dot product}}$$

The dot product can be written as: $\|\mathbf{m}^3\| \|(\tilde{\mathbf{X}} - \tilde{\mathbf{C}})\| \cos\theta = \mathrm{sign}(\det \mathrm{M})w$,

where the depth is given by:

$$\mathrm{depth}(\mathbf{X}; \mathrm{P}) = \|(\tilde{\mathbf{X}} - \tilde{\mathbf{C}})\| \cos\theta = \frac{\mathrm{sign}(\det \mathrm{M})w}{\|\mathbf{m}^3\|}. \qquad \square$$

Image source: “Multiple View Geometry in Computer Vision”, Richard Hartley and Andrew Zisserman

Decomposition of the Camera Matrix

- **Given:** The camera matrix P representing a general projective camera.

- **Find:** The camera centre, the orientation of the camera and the internal parameters of the camera.

Decomposition of the Camera Matrix

Finding the camera centre:

- The principal and two axis planes we have seen earlier intersect at the camera centre, i.e. the null-space of $\mathrm{P}\mathbf{C} = \mathbf{0}$, where

$$\mathrm{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} = \begin{bmatrix} \mathbf{P}^{1\mathsf{T}} \\ \mathbf{P}^{2\mathsf{T}} \\ \mathbf{P}^{3\mathsf{T}} \end{bmatrix}.$$

- The null-space is given by:

$$\begin{aligned} \mathrm{X} &= \det([\mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4]) & \mathrm{Y} &= -\det([\mathbf{p}_1, \mathbf{p}_3, \mathbf{p}_4]) \\ \mathrm{Z} &= \det([\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_4]) & \mathrm{T} &= -\det([\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3]). \end{aligned}$$

Decomposition of the Camera Matrix

Finding camera orientation and internal parameters:

- In the case of a finite camera:

$$\mathrm{P} = [\mathrm{M} \mid -\mathrm{M}\widetilde{\mathrm{C}}] = \mathrm{K}[\mathrm{R} \mid -\mathrm{R}\widetilde{\mathrm{C}}] ,$$

  KR can be found from the RQ decomposition of M.

- The ambiguity in the decomposition is removed by requiring that K have positive diagonal entries.

Decomposition of the Camera Matrix

Finding camera orientation and internal parameters:

- The matrix K has the form:

$$\mathtt{K} = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where

- α_x is the scale factor in the x-coordinate direction,
- α_y is the scale factor in the y-coordinate direction,
- s is the skew,
- $(x_0, y_0)^\mathsf{T}$ are the coordinates of the principal point.

The *aspect ratio* is α_y/α_x.

Euclidean vs Projective Spaces

- The development of the camera model has implicitly assumed that the world and image coordinate systems are Euclidean.

- However, the projective camera is a mapping from $\mathbb{P}^2 \rightarrow \mathbb{P}^3$, i.e. a composed effects of:

$$\mathrm{P} = \underbrace{[3 \times 3 \text{ homography}]}_{\text{projective transformation of the image}} \overbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}^{\text{projection from 3-space to an image}} \underbrace{[4 \times 4 \text{ homography}]}_{\text{projective transformation of 3-space}}$$

Cameras at Infinity: Affine camera

- The camera matrix of an affine camera has the form:

$$\mathrm{P}_{\mathrm{A}} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & t_1 \\ m_{21} & m_{22} & m_{23} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

- These are cameras with centre lying on the plane at infinity, i.e.

1. $\mathrm{C} = (\mathbf{d}, 0)^{\top}$ is an idea point, where d is the null-space of $\mathbf{M}_{2\times3}\mathbf{d} = \mathbf{0}$ since $\mathrm{P}\mathbf{C} = \mathbf{0}$.
2. $\mathbf{P}^{3\top} = (0,0,0,1)$ which is the principal plane must be the plane at infinity.

- The left hand 3×3 block of the camera matrix P_{A} is singular.

Cameras at Infinity: Affine camera

- The affine camera matrix can be decomposed into:

$$\mathrm{P}_{\mathrm{A}} = \underbrace{[3 \times 3 \text{ affine}]}_{\text{affine transformation of the image}} \overbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}^{\text{orthographic projection from 3-space to an image}} \underbrace{[4 \times 4 \text{ affine}]}_{\text{affine transformation of 3-space}}$$

Cameras at Infinity: Affine camera

- Alternatively:

$$\mathrm{P}_{\mathrm{A}} = \underbrace{\begin{bmatrix} \mathrm{K}_{2\times 2} & \tilde{\mathbf{x}}_0 \\ \hat{\mathbf{0}}^\top & 1 \end{bmatrix}}_{\text{calibration matrix}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{R} & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_x & s & p_x \\ & \alpha_y & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\top} & t_1 \\ \mathbf{r}^{2\top} & t_2 \\ \mathbf{0}^\top & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & s & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\top} & t_1 \\ \mathbf{r}^{2\top} & t_2 \\ \mathbf{0}^\top & 1 \end{bmatrix},$$

where

- $\tilde{\mathbf{x}}_0 = (p_x, p_y)$ is the principal point, which is conventionally set to 0;
- $\hat{\mathbf{0}}^\top = (0,0)$;
- (α_x, α_y) are the scale factors and s is the skew parameter.

Cameras at Infinity: Affine camera

- The affine camera matrix:

$$\mathrm{P}_{\mathrm{A}} = \begin{bmatrix} \alpha_x & s & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\mathsf{T}} & t_1 \\ \mathbf{r}^{2\mathsf{T}} & t_2 \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & t_1 \\ m_{21} & m_{22} & m_{23} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

- Has eight degrees of freedom corresponding to the eight non-zero and non-unit matrix elements.

- The sole restriction on the affine camera is that $\mathrm{M}_{2\times 3}$ has rank 2.

Affine Properties of Camera at Infinity

1. The plane at infinity in space is mapped to points at infinity in the image.

   Proof: This is easily seen by computing $\mathrm{P_A}(\mathrm{X}, \mathrm{Y}, \mathrm{Z}, 0)^\top = (\mathrm{X}, \mathrm{Y}, 0)^\top$. □

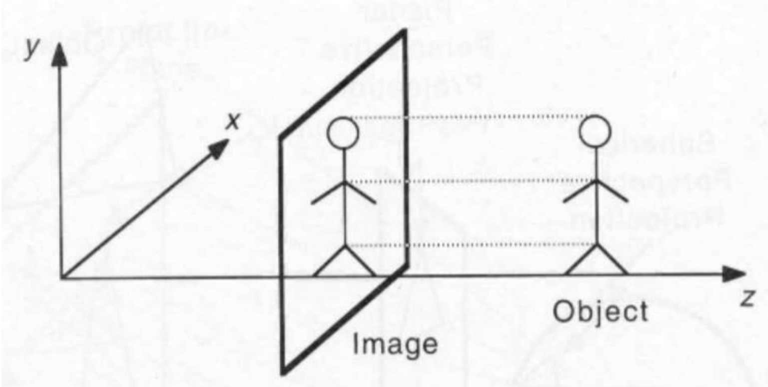
2. Parallel world lines are projected to parallel image lines.

   Sketch of Proof:
   Parallel world lines intersect at the plane at infinity, and this intersection point is mapped to a point at infinity in the image. Hence the image lines are parallel. □

A Hierarchy of Affine Cameras

1. **Orthographic projection:**

- No change in scale ⇒ camera calibration = identity.
- The optical center is located at infinity.
- The projection rays are parallel.
- The model ignores depth altogether.



Camera projection matrix:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}^{1\mathsf{T}} & t_1 \\ \mathbf{r}^{2\mathsf{T}} & t_2 \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$$

Slide adapted from: https://kth.instructure.com/files/1316041/download?download_frd=1

A Hierarchy of Affine Cameras

1. **Orthographic projection:**

Camera projection matrix:

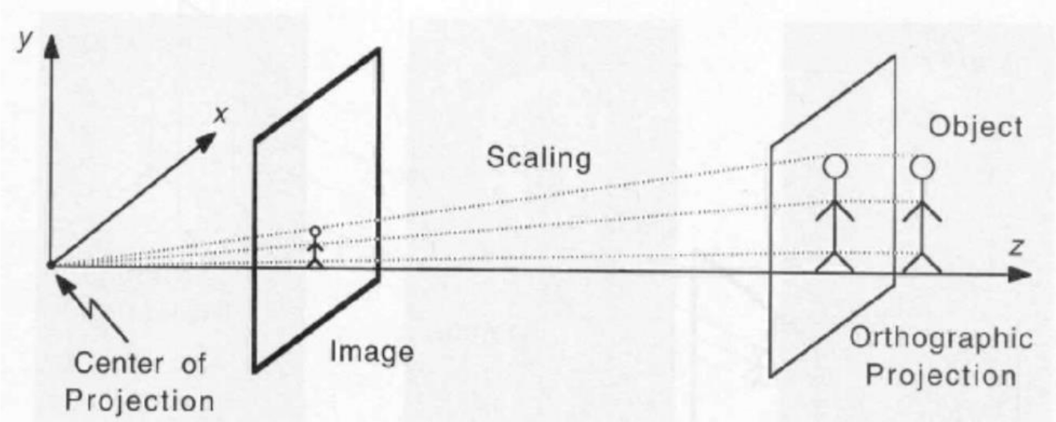
$$\mathrm{P} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathrm{R} & \mathbf{t} \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}^{1\mathsf{T}} & t_1 \\ \mathbf{r}^{2\mathsf{T}} & t_2 \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}$$

- An orthographic camera has five degrees of freedom: three parameters for rotation matrix R, plus two offset parameters t_1 and t_2.

- An orthographic projection matrix $\mathrm{P} = [\mathrm{M} \mid \mathbf{t}]$ is characterized by a matrix M with last row zero, first two rows orthogonal and of unit norm, and $t_3 = 1$.

A Hierarchy of Affine Cameras

2. **Scaled orthographic projection:**



A point in 3D space is:

i. projected to a reference plane using orthographic projection; and then

ii. projected to the image plane using a perspective projective.

Slide adapted from: https://kth.instructure.com/files/1316041/download?download_frd=1

A Hierarchy of Affine Cameras

2. **Scaled orthographic projection:**

Camera projection matrix:

$$\mathrm{P} = \begin{bmatrix} k & & \\ & k & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\top} & t_1 \\ \mathbf{r}^{2\top} & t_2 \\ \mathbf{0}^\top & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}^{1\top} & t_1 \\ \mathbf{r}^{2\top} & t_2 \\ \mathbf{0}^\top & 1/k \end{bmatrix}.$$

- It has six degrees of freedom; one additional for the equal scale factors.
- A scaled orthographic projection matrix $\mathrm{P} = [\mathrm{M} \mid \mathbf{t}]$ is characterized by a matrix M with last row zero, and the first two rows orthogonal and of equal norm.

Slide adapted from: https://kth.instructure.com/files/1316041/download?download_frd=1

A Hierarchy of Affine Cameras

3. **Weak perspective projection**

- Similar to scaled orthographic projection.
- Difference: allow two different scalings in the two different axial image directions.

Camera projection matrix:

$$\mathrm{P} = \begin{bmatrix} \alpha_x & & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\mathsf{T}} & t_1 \\ \mathbf{r}^{2\mathsf{T}} & t_2 \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix} .$$

A Hierarchy of Affine Cameras

3. **Weak perspective projection**

Camera projection matrix:

$$\mathrm{P} = \begin{bmatrix} \alpha_x & & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1\mathsf{T}} & t_1 \\ \mathbf{r}^{2\mathsf{T}} & t_2 \\ \mathbf{0}^{\mathsf{T}} & 1 \end{bmatrix}.$$

- It has seven degrees of freedom; one additional for the different scale factors.
- A weak perspective projection matrix $\mathrm{P} = [\mathrm{M} \mid \mathbf{t}]$ is characterized by a matrix M with last row zero, and first two rows orthogonal (no need for equal norm).

Calibration of Projective Camera

- We have seen that the camera projection matrix P has 11 degrees of freedom:

$$P = K[R \quad t]$$

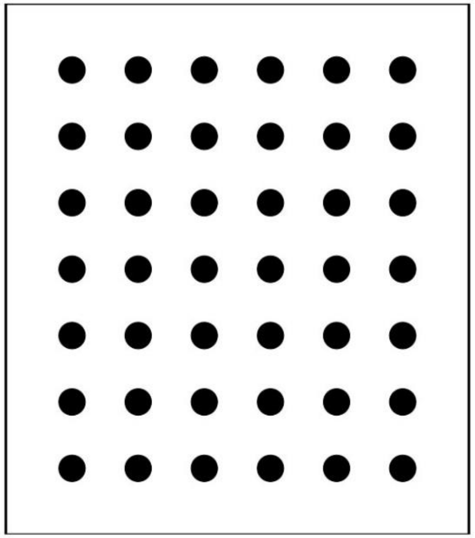
Component	# DOF	Elements	Known As
K	5	f_x, f_y, s, p_x, p_y	Intrinsic Parameters
R	3	α, β, γ	Extrinsic Parameters
$\tilde{C}$ or t	3	(C_x, C_y, C_z) or (t_x, t_y, t_z)	

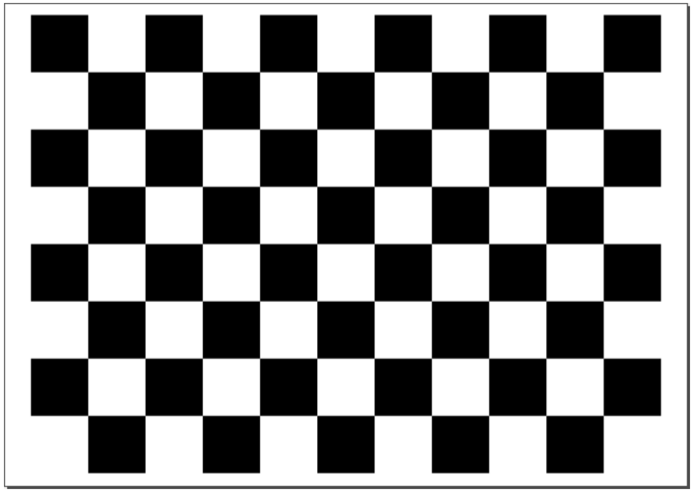
Total: 11 DOF

How do we find all the 11 parameters?

Calibration of Projective Camera

- Estimation of the camera intrinsic and extrinsic parameters is known as resectioning.
- Most used approach: Use a 2D calibration pattern (e.g. a checkerboard).



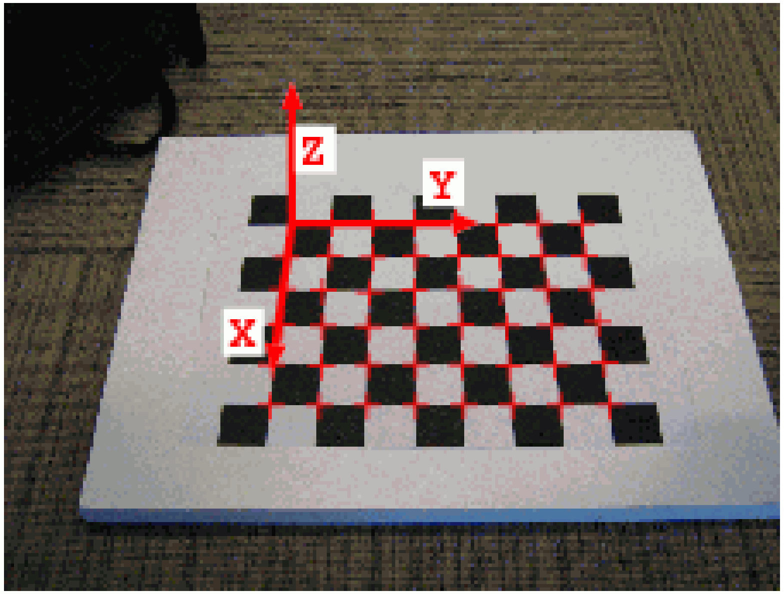


Camera Calibration: Open Source

- Z. Y. Zhang, "A Flexible New Technique for Camera Calibration", TPAMI 2000.
- Bouguet Calibration Toolbox: http://www.vision.caltech.edu/bouguetj/calib_doc/
- OpenCV Calibration: http://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html
- Matlab Image Processing Toolbox: http://www.mathworks.com/help/vision/single-camera-calibration.html

Calibration of Projective Camera

- Set the world coordinate system to the corner of the checkerboard.
- Now all 3D points on the checkerboard lie on a single plane, i.e. Z=0.



Calibration of Projective Camera

- Let us denote the i^{th} column of the rotation matrix R by r_i, we have:

Scale factor

$$s\begin{bmatrix}x\\y\\1\end{bmatrix} = \mathrm{K}[r_1 \quad r_2 \quad r_3 \quad t]\begin{bmatrix}X\\Y\\0\\1\end{bmatrix}$$

3D points lie on a plane, i.e. Z=0

- 2D-3D correspondence $(x, y) \leftrightarrow (X, Y)$ respectively lies on planes, hence related by a homography:

$$s\begin{bmatrix}x\\y\\1\end{bmatrix} = \underbrace{\mathrm{K}[r_1 \quad r_2 \quad t]}\begin{bmatrix}X\\Y\\1\end{bmatrix} \implies s[h_1 \quad h_2 \quad h_3] = \mathrm{K}[r_1 \quad r_2 \quad t]$$

where h_i is the i^{th} column H

Homography: $\mathrm{H} = [h_1 \quad h_2 \quad h_3]$

Calibration of Projective Camera

- Recall $r_1 \times r_2 = r_3 \Rightarrow$ we get two independent constraints:

$$s[h_1 \quad h_2 \quad h_3] = \mathrm{K}[r_1 \quad r_2 \quad t]$$
$$\Rightarrow sK^{-1}h_1 = r_1, \qquad sK^{-1}h_2 = r_2$$

- Using the orthonormal constraints of a rotation matrix, we get:

$$r_1^\top r_2 = 0 \Longrightarrow \qquad h_1^\top K^{-\top}K^{-1}h_2 = 0 \tag{1}$$

$$\|r_1\| = \|r_2\| \Longrightarrow \qquad h_1^\top K^{-\top}K^{-1}h_1 = h_2^\top K^{-\top}K^{-1}h_2 \tag{2}$$

- Equations (1) and (2) are now independent of the camera extrinsics.

Calibration of Projective Camera

- Let us denote:

$$\mathrm{K}^{-\top}\mathrm{K}^{-1} = B = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix}$$

- *B* is symmetric and positive definite.
- Since *B* is symmetric, it can be represented as a 6-vector:

$$\mathbf{b}=[B_{11}, B_{12}, B_{13}, B_{22}, B_{23}, B_{33}]^\top$$

Calibration of Projective Camera

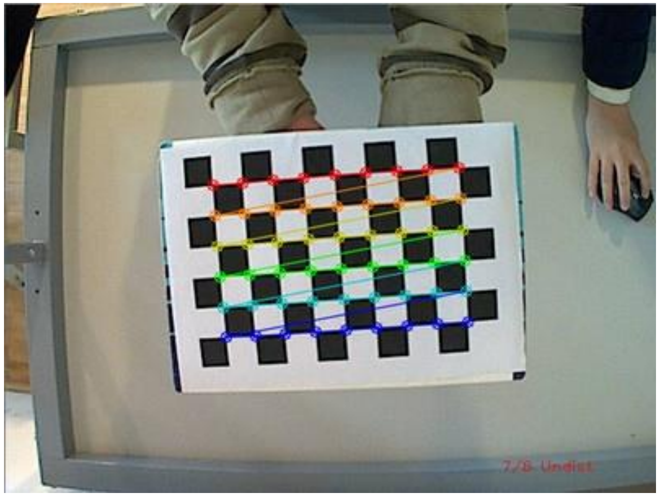
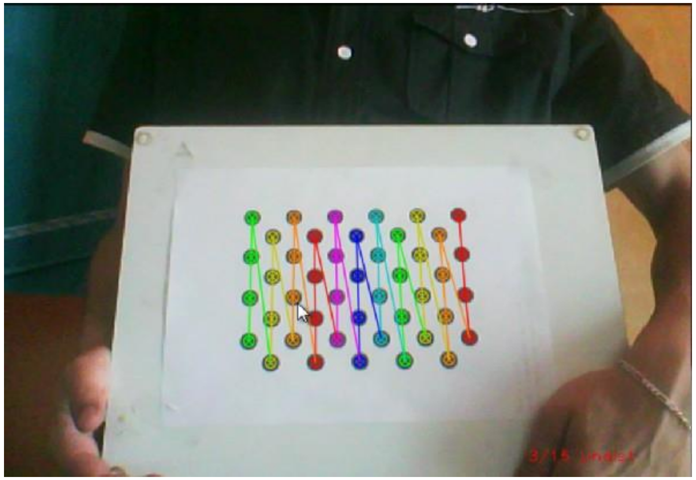
- Putting the 6-vector **b** into Equations (1) and (2).
- Re-arranging the homography terms, we get:

$$a\mathbf{b}=0$$

- a is a 2x6 matrix made up of the homography terms h_1 and h_2.

Calibration of Projective Camera

- Each view of the checkerboard gives us two constraints.
- A minimum of three different views to solve for the 6 unknowns in **b**.
- At least four 2D-3D correspondences per plane for homography.

OpenCV detects the point correspondences automatically

Image source: http://docs.opencv.org/2.4/doc/tutorials/calib3d/camera_calibration/camera_calibration.html

Calibration of Projective Camera

- For *n* ≥ *3* different views, we get: $A\mathbf{b}=0$
- A is a 2*n* x 6 matrix obtained from stacking 2*n* constraints together.
- A least-squares solution of **b** can be obtained by taking the 6-vector right null-space of A (using SVD).

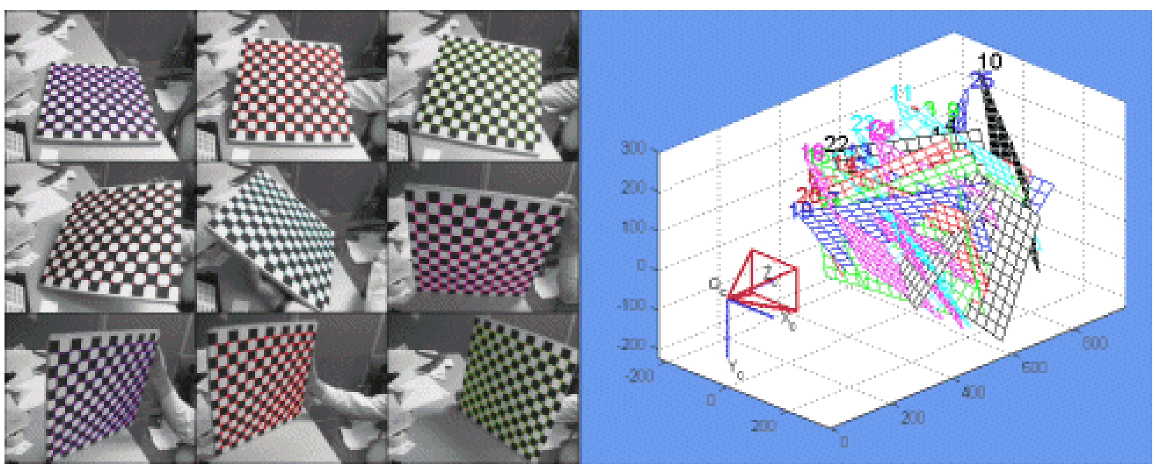


Image Source: http://www.vision.caltech.edu/bouguetj/calib_doc/

Calibration of Projective Camera

- K can be recovered from B by doing Cholesky decomposition $\Rightarrow f_x, f_y, s, p_x, p_y$ can be recovered.

- Once K is known, the extrinsic parameters of all views can be solved:

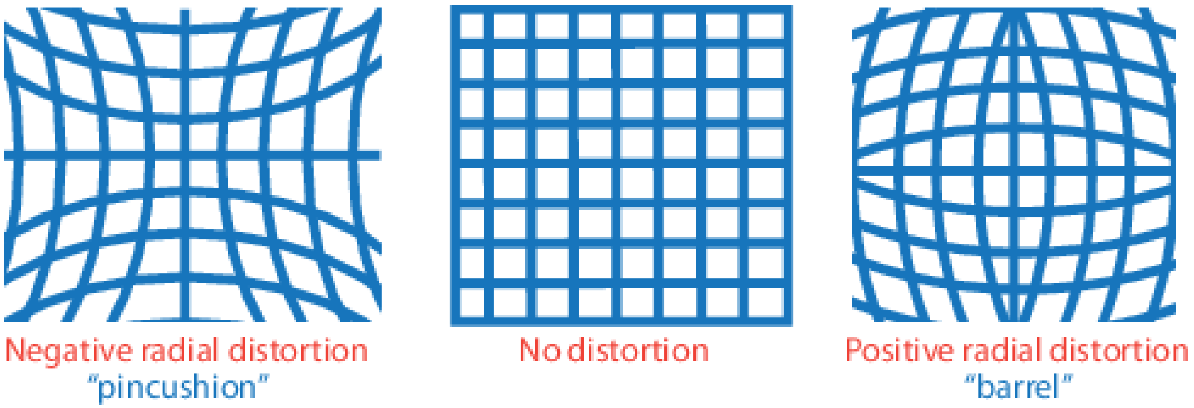
$$r_1 = s\mathrm{K}^{-1}h_1, \quad r_2 = s\mathrm{K}^{-1}h_2, \quad r_3 = r_1 \times r_2, \quad t = s\mathrm{K}^{-1}h_3,$$

where

$$s = \frac{1}{\|\mathrm{K}^{-1}h_1\|} = \frac{1}{\|\mathrm{K}^{-1}h_2\|}$$

Lens Distortion

1. Radial distortion (More common)



2. Tangential distortion (Less common)

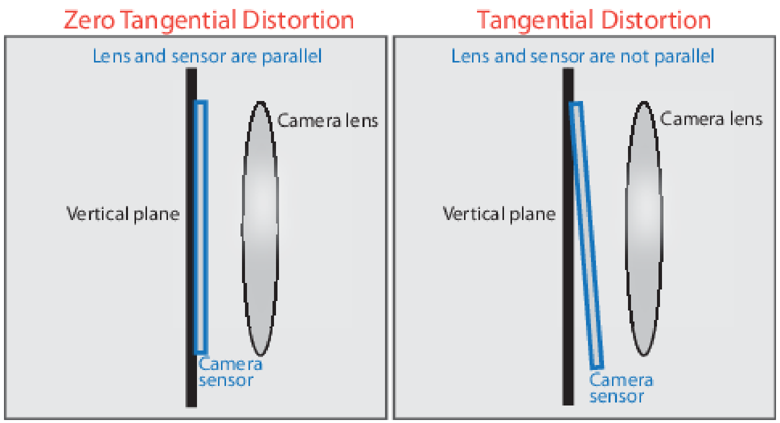


Image source: http://www.mathworks.com/help/vision/ug/camera-calibration.html

Lens Distortion: Radial Distortion

- Let x = (x, y) be the image projection of a 3D point without distortion.
- The image point after radial distortion is given by:

$$\mathrm{x}_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = (1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_5 r^6) \begin{bmatrix} x \\ y \end{bmatrix}$$

where

- $r^2 = x^2 + y^2$
- $\kappa_1, \kappa_2, \kappa_5$: 3 Radial distortion parameters

Reference: "Close-Range Camera Calibration" **-** D.C. Brown, Photogrammetric Engineering, pages 855-866, Vol. 37, No. 8, 1971.

Lens Distortion: Tangential Distortion

- The image point after tangential distortion is given by:

$$\mathrm{dx} = \begin{bmatrix} 2\kappa_3 xy + \kappa_4(r^2 + 2x^2) \\ \kappa_3(r^2 + 2y^2) + 2\kappa_4 xy \end{bmatrix}$$

where

- $r^2 = x^2 + y^2$
- κ_3, κ_4 : 2 Tangential distortion parameters

Reference: "Close-Range Camera Calibration" - D.C. Brown, Photogrammetric Engineering, pages 855-866, Vol. 37, No. 8, 1971.

Lens Distortion

- Combining radial and tangential distortions:

$$\mathrm{x_d} = \mathrm{x}_r + \mathrm{dx}$$
$$= (1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_5 r^6)\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2\kappa_3 xy + \kappa_4(r^2 + 2x^2) \\ \kappa_3(r^2 + 2y^2) + 2\kappa_4 xy \end{bmatrix}$$

where

- $r^2 = x^2 + y^2$
- $\kappa_1, \kappa_2, \kappa_5$: 3 Radial distortion parameters
- κ_3, κ_4 : 2 Tangential distortion parameters

Lens Distortion: Maximum Likelihood Estimation

Steps:

1. Estimate intrinsic parameters in K, i.e. f_x, f_y, s, p_x, p_y, and extrinsic parameters, i.e. R_i and t_i for all views without taking lens distortions into account.

2. Initialize all lens distortion parameters to 0, i.e. $\kappa_1 = \kappa_2 = \kappa_3 = \kappa_4 = \kappa_5 = 0$.

3. Minimize the total reprojection error over all parameters:

$$\underset{\mathrm{K},R,t,\kappa}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| \mathrm{x}_{ij} - \pi\left(\mathrm{K}, R_i, t_i, \kappa, \mathrm{X_j}\right) \right\|^2$$

Use Levenberg-Marquardt to minimize this!

Lens Distortion: Maximum Likelihood Estimation

views # 3D points

$$\underset{\mathrm{K},R,t,\kappa}{\operatorname{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| \mathrm{x}_{ij} - \pi(\mathrm{K}, R_i, t_i, \kappa, \mathrm{X_j}) \right\|^2$$

- X_j : j^{th} 3D point
- x_{ij} : 2D image point from the i^{th} view corresponding to the X_j
- K : camera intrinsic
- (R_i, t_i) : extrinsic of the i^{th} view
- $\kappa = (\kappa_1, \kappa_2, \kappa_3, \kappa_4, \kappa_5)$: lens distortion parameters
- $\pi(.)$: projection function + lens distortion

$$\breve{\mathrm{x}}_{ij} = \mathrm{K}[\mathrm{R}_i \quad \mathrm{t}_i]\mathrm{X_j} \quad \Longrightarrow \quad \pi(.) = \mathrm{x_{d_{ij}}} = \mathrm{x}_r(\breve{\mathrm{x}}_{ij}) + \mathrm{dx}(\breve{\mathrm{x}}_{ij})$$

Lens Distortion Correction

Before and after lens distortion correction

Summary

- We have looked at how to:

1. Describe camera projection with the pinhole model.
2. Identify the camera centre, principal planes, principal point, and principal axis from the projection matrix.
3. Use the projection matrix to get the forward and backward projection of a point.
4. Explain the properties of an affine camera.
5. Do calibration to find the intrinsic and extrinsic values of a projective camera.