

CS4277 / CS5477

3D Computer Vision

Lecture 4: Camera models and calibration

Asst. Prof. Lee Gim Hee

AY 2019/20

Semester 2

Course Schedule

Week	Date	Topic	Assignments
1	15 Jan	2D and 1D projective geometry	
2	22 Jan	Circular points and 3D projective geometry	
3	29 Jan	No Lecture	
4	05 Feb	Absolute conic and robust homography estimation	Assignment 1: Panoramic stitching (15%)
5	12 Feb	Camera models and calibration	
6	19 Feb	Single view metrology	Due: Assignment 1 Assignment 2: Camera calibration (15%)
-	26 Feb	Semester Break	No lecture
7	04 Mar	The fundamental and essential matrices	Due: Assignment 2
8	11 Mar	Absolute pose estimation from points and/or lines	Assignment 3: Relative and absolute pose estimation (20%)
9	18 Mar	Multiple-view geometry from points and/or lines	
10	25 Mar	Structure-from-Motion (SfM) and Visual Simultaneous Localization and Mapping (vSLAM)	Due: Assignment 3
11	01 Apr	Two-view and multi-view stereo	Assignment 4: Dense 3D model from multi-view stereo (20%)
12	08 Apr	Generalized cameras	
13	15 Apr	Factorization and non-rigid structure-from-motion	Due: Assignment 4

*Possible make-up lecture (to be confirmed): Auto-Calibration

Learning Outcomes

- Students should be able to:
 1. Describe the **pinhole model** for camera projection with the camera **intrinsic** and **extrinsic** parameters.
 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.

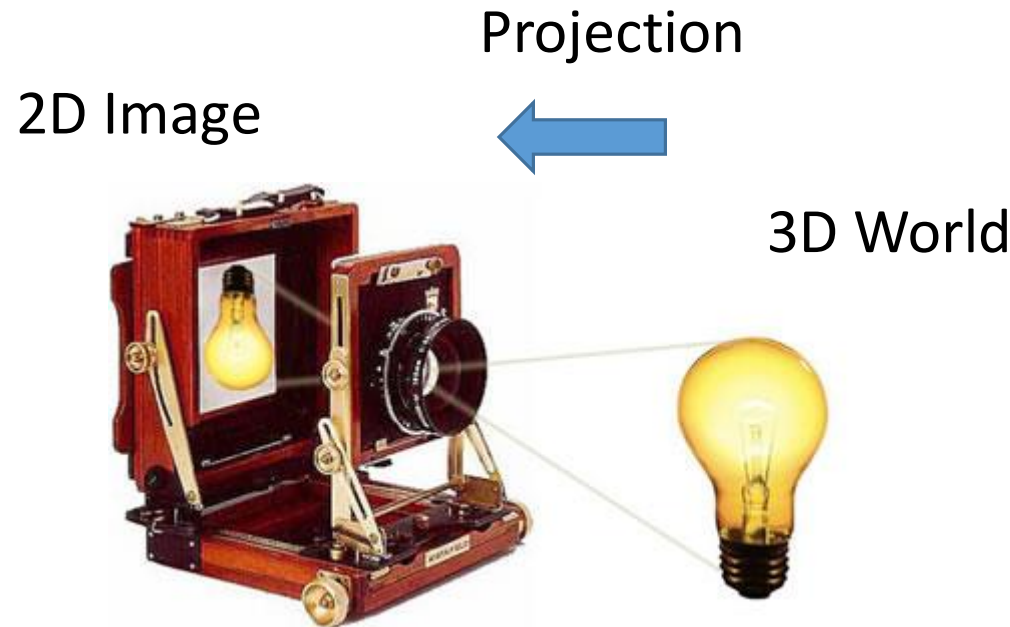


Image source: <http://www.shortcourses.com/guide/guide1-3.html>

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**.

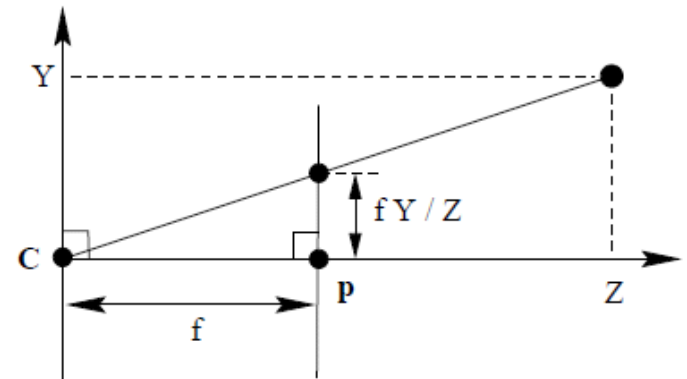
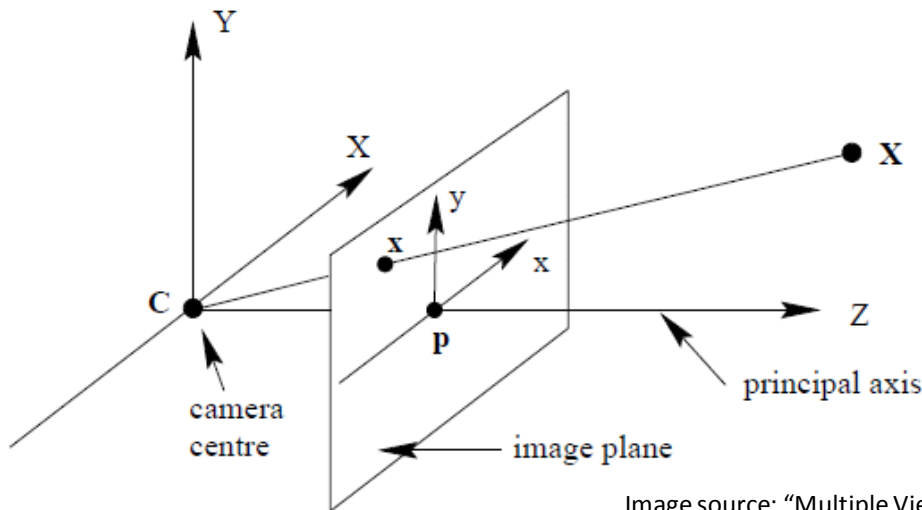


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

The Basic Pinhole Model

- Using **similar triangle**, we can see that the point $(X, Y, Z)^T$ is mapped to the point $(fX/Z, fY/Z, f)^T$ on the image plane.
- Ignoring the final coordinate, we get the **central projection mapping** from world to image coordinates :

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

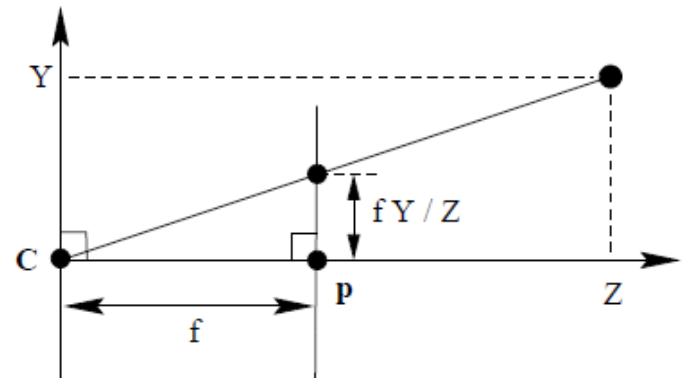
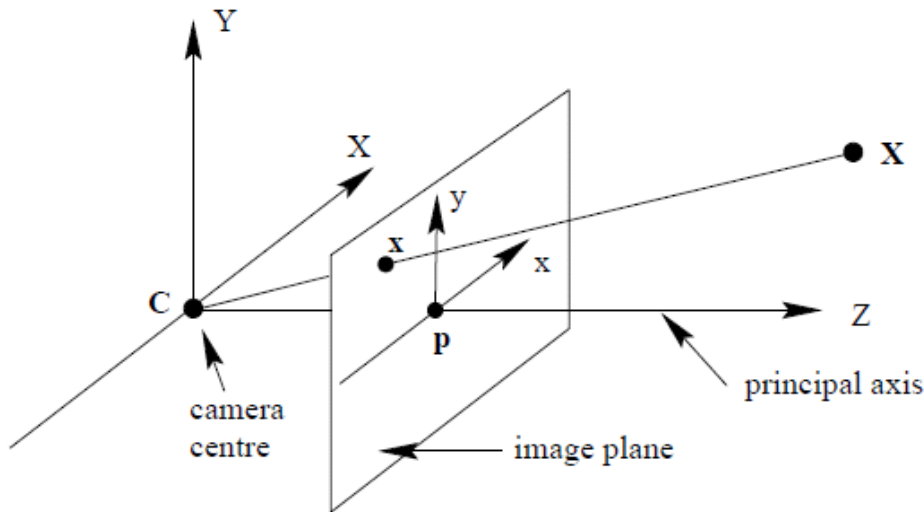
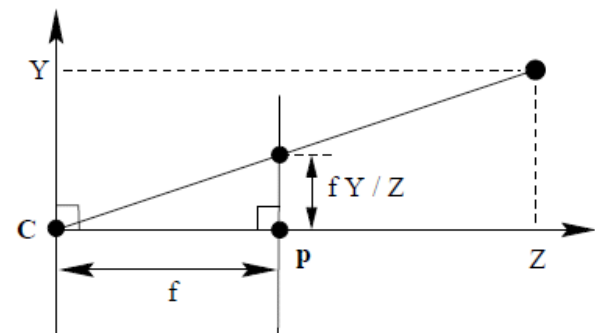
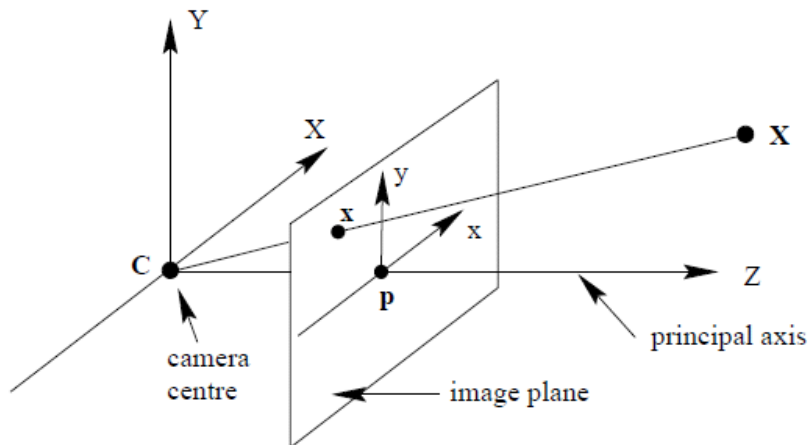


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

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.



Central Projection Using Homogeneous Coordinates

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

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

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

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

- 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.

$$(X, Y, Z)^T \mapsto (fX/Z + p_x, fY/Z + p_y)^T.$$

- $(p_x, p_y)^T$ are the coordinates of the **principal point**.
- Expressing in **homogeneous coordinates**, we get:

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

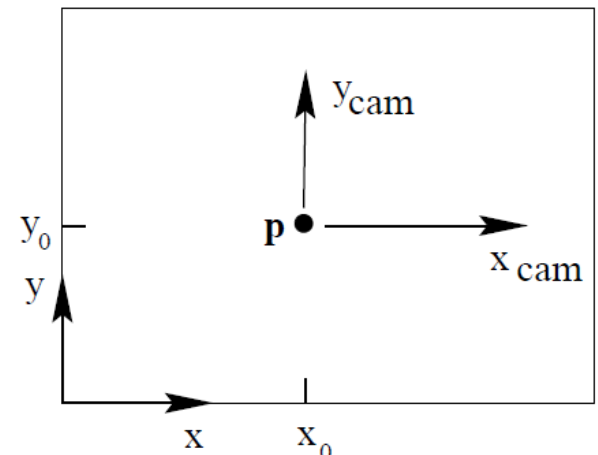


Image source: “Multiple View Geometry in Computer Vision”, Richard Hartley and Andrew Zisserman

Camera Calibration Matrix

- Now, writing:

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

- We can rewrite

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

- The matrix K is called the **camera calibration matrix**.

Camera Rotation and Translation

- $\mathbf{X}_{\text{cam}} = (X, Y, Z, 1)^T$ 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) .

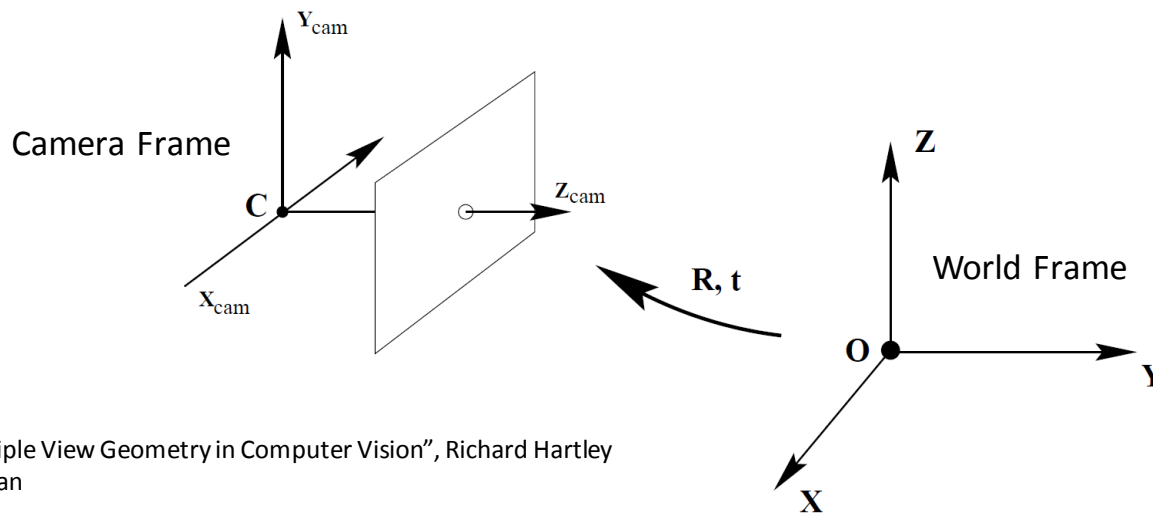
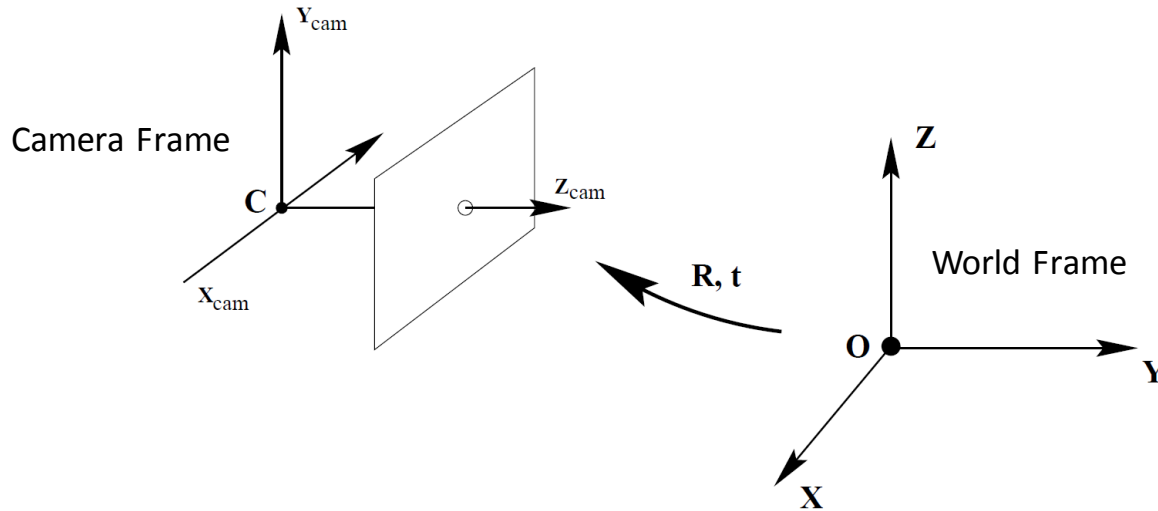


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

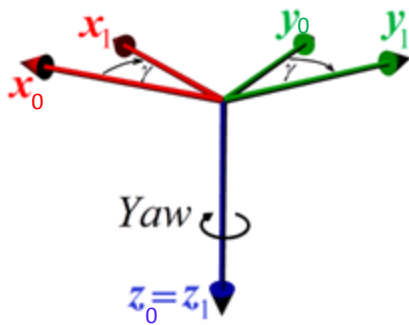
Camera Rotation and Translation



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

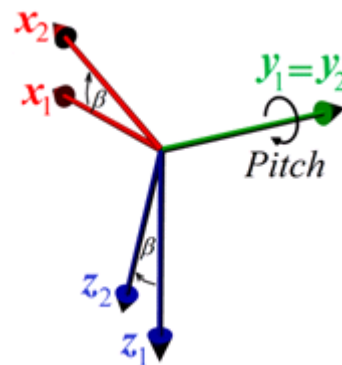
$$\mathbf{X}_{\text{cam}} = \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} \mathbf{R} & -\mathbf{R}\tilde{\mathbf{C}} \\ 0 & 1 \end{bmatrix} \mathbf{X}.$$

Euler Angles to Rotation Matrix



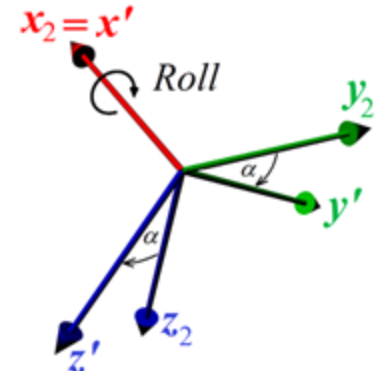
$$R_z(\gamma) = R_1^0 = \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

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



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

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



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

$$\Rightarrow X_3 = R_3^2 X_2$$

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

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

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. $R^T = R^{-1}$,
 3. $r_i \times r_j = r_k$, (third column is the cross-product of the other two columns)
 4. $r_i^T r_j = 0$, 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 $\mathbf{x} = K[\mathbf{I} \mid \mathbf{0}]\mathbf{X}_{cam}$, we get the **general mapping of a pinhole camera**:

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

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

- We write the **camera projection matrix** as:

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

- \mathbf{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{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) :

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

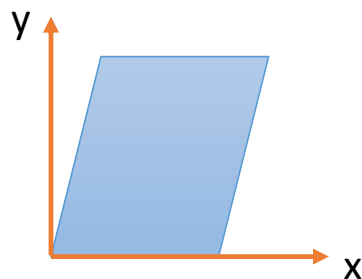
By rewriting $t = -R\tilde{C}$.

Non-Square and Skewed Pixels

- **Same focal length f** for both x and y directions in camera calibration matrix \Rightarrow 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:

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

Camera Intrinsic and Extrinsic

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

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

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

Finite Projective Cameras

- The set of camera matrices of **finite projective cameras**

$$P = KR[I \mid -\tilde{C}]$$

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

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

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.

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

which is an **undefined** image point $(0, 0, 0)^T$.

Finite Projective Cameras: Camera Anatomy

Sketch of Proof:

- Consider the **line** containing **C** and any other point **A** in 3-space,

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

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

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

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

□

Finite Projective Cameras: Camera Anatomy

Column vectors:

- With the notation that the columns of P are \mathbf{p}_i , $i = 1, \dots, 4$, 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 $\mathbf{D} = (1, 0, 0, 0)^T$, which is imaged at $\mathbf{p}_1 = P\mathbf{D}$.

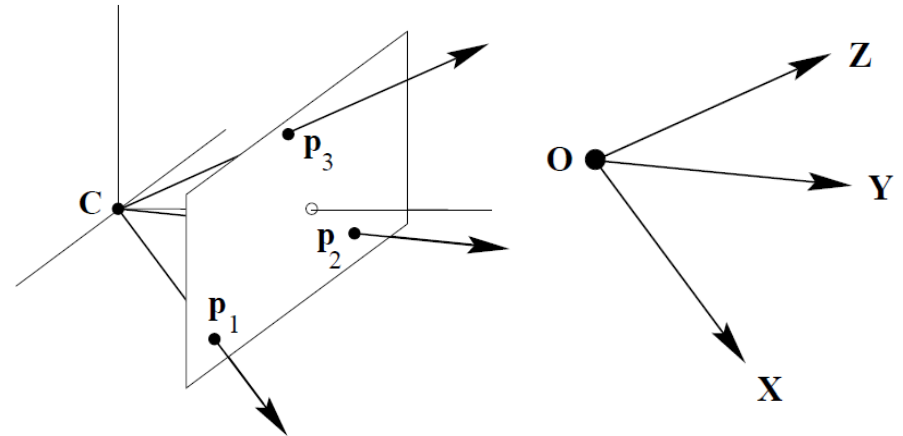


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

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.

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

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

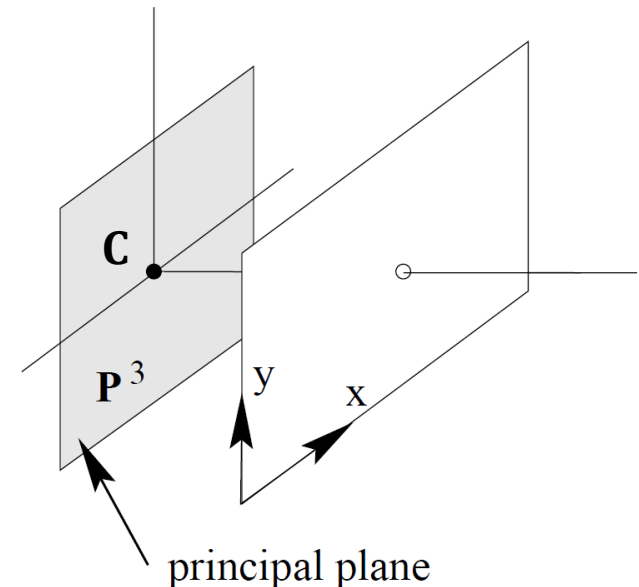


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Finite Projective Cameras: Camera Anatomy

2. Axis plane:

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

Sketch of Proof:

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

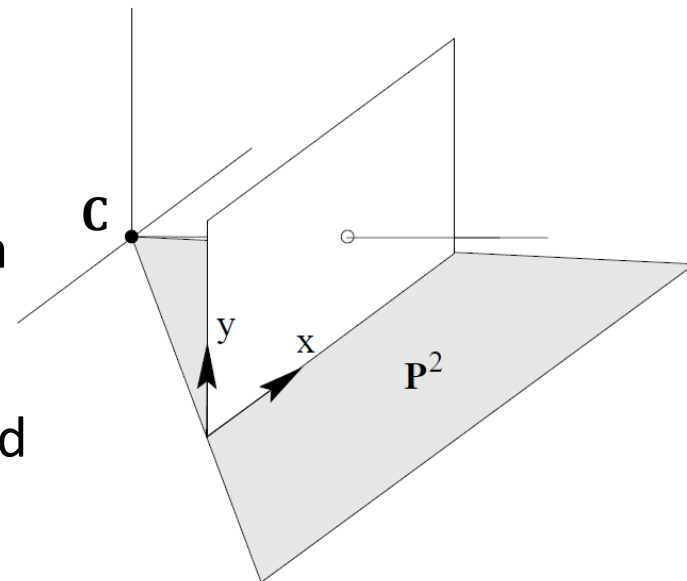


Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Finite Projective Cameras: Camera Anatomy

The principal point:

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

Finite Projective Cameras: Camera Anatomy

Remarks:

- The point $\hat{\mathbf{P}}^3 = (p_{31}, p_{32}, p_{33}, 0)^T$ denotes the **direction of the normal vector** of the principal plane.
- This point **projects onto the image** as the principal point, i.e. $\mathbf{x}_0 = P\hat{\mathbf{P}}^3$ which can be written as:

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

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 $\mathbf{x} = P\mathbf{X}$.
- In reality, **only half the points** in space, those that lie in front of the camera, may be seen in an image.
- $\mathbf{v} = \det(M) \mathbf{m}^3$ 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 $P = [M \mid \mathbf{p}_4]$.
- 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 $\det(M)$, which is the **signed area equivalent**.

Summary of the properties of P

Camera centre. The camera centre is the 1-dimensional right null-space C of P , i.e. $PC = 0$.

- ◇ **Finite camera** (M is not singular) $C = \begin{pmatrix} -M^{-1}p_4 \\ 1 \end{pmatrix}$
- ◇ **Camera at infinity** (M is singular) $C = \begin{pmatrix} d \\ 0 \end{pmatrix}$ where d is the null 3-vector of M , i.e. $Md = 0$.

Column points. For $i = 1, \dots, 3$, the column vectors p_i are vanishing points in the image corresponding to the X , Y and Z axes respectively. Column p_4 is the image of the coordinate origin.

Principal plane. The principal plane of the camera is P^3 , the last row of P .

Axis planes. The planes P^1 and P^2 (the first and second rows of 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 $x_0 = Mm^3$ is the principal point of the camera, where m^{3T} is the third row of M .

Principal ray. The principal ray (axis) of the camera is the ray passing through the camera centre C with direction vector m^{3T} . The principal axis vector $v = \det(M)m^3$ 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 $\mathbf{x} = \mathbf{P}\mathbf{X}$.
- Points $\mathbf{D} = (\mathbf{d}^T, 0)^T$ on the plane at infinity represent **vanishing points**; such points are mapped to:

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

- Thus are **only affected by \mathbf{M}** , i.e. the first 3×3 submatrix of \mathbf{P} .

Action of a Projective Camera on Points

Back-projection of points to rays:

- The ray is **the line**

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

formed by the join of two points:

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

Action of a Projective Camera on Points

Back-projection of points to rays:

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

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

where

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

Action of a Projective Camera on Points

Depth of points:

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

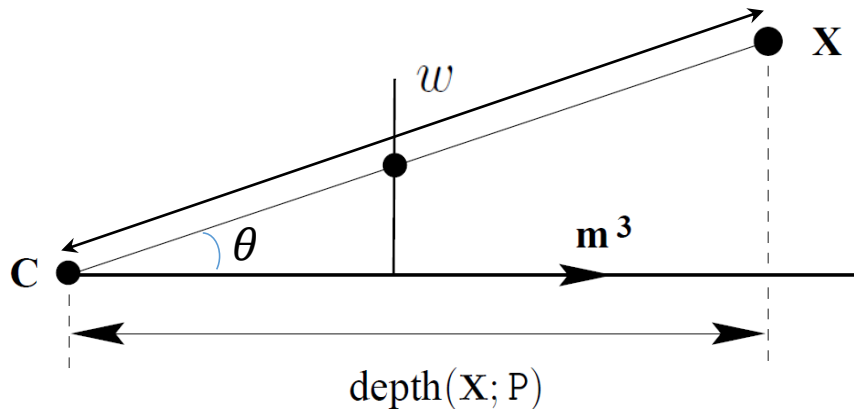
$$\text{depth}(\mathbf{X}; P) = \frac{\text{sign}(\det M)w}{T\|\mathbf{m}^3\|}$$

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: $\mathbf{X} = (X, Y, Z, 1)^T = (\tilde{\mathbf{X}}^T, 1)^T$

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

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

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

The dot product can be written as: $\|\mathbf{m}^3\| \|(\tilde{\mathbf{X}} - \tilde{\mathbf{C}})\| \cos \theta = \text{sign}(\det \mathbf{M}) w$,
where the depth is given by:

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

□

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 seen earlier intersect at the camera centre, i.e. the null-space of $PC = \mathbf{0}$, where

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

- The null-space is given by:

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

Decomposition of the Camera Matrix

Finding camera orientation and internal parameters:

- In the case of a finite camera:

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

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:

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

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,
- $(x_0, y_0)^T$ 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:

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

Cameras at Infinity: Affine camera

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

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

- These are cameras with **centre lying on the plane at infinity**, i.e.
 1. $C = (\mathbf{d}, 0)^T$ is an **idea point**, where \mathbf{d} is the null-space of $\mathbf{M}_{2 \times 3} \mathbf{d} = \mathbf{0}$ since $\mathbf{P}\mathbf{C} = \mathbf{0}$.
 2. $\mathbf{P}^{3T} = (0,0,0,1)$ 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:

orthographic projection
from 3-space to an image

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

Cameras at Infinity: Affine camera

- Alternatively:

$$\begin{aligned} P_A &= \overbrace{\begin{bmatrix} K_{2 \times 2} & \tilde{\mathbf{x}}_0 \\ \hat{\mathbf{0}}^T & 1 \end{bmatrix}}^{\text{calibration matrix}} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \\ &= \begin{bmatrix} \alpha_x & s & p_x \\ & \alpha_y & p_y \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1T} & t_1 \\ \mathbf{r}^{2T} & t_2 \\ \mathbf{0}^T & 1 \end{bmatrix} = \begin{bmatrix} \alpha_x & s & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}^{1T} & t_1 \\ \mathbf{r}^{2T} & t_2 \\ \mathbf{0}^T & 1 \end{bmatrix}, \end{aligned}$$

where

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

Cameras at Infinity: Affine camera

- The affine camera matrix:

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

- 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 $\mathbf{M}_{2 \times 3}$ 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 $P_A(X, Y, Z, 0)^T = (X, Y, 0)^T$. \square

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

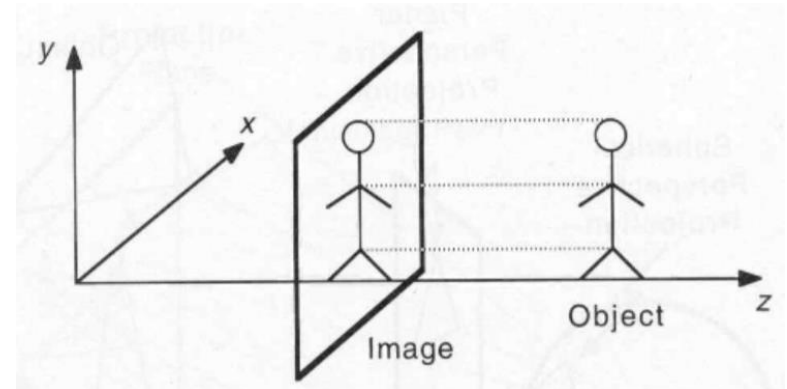
Proof:

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. \square

A Hierarchy of Affine Cameras

1. Orthographic projection:

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



Camera projection matrix:

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

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

A Hierarchy of Affine Cameras

1. Orthographic projection:

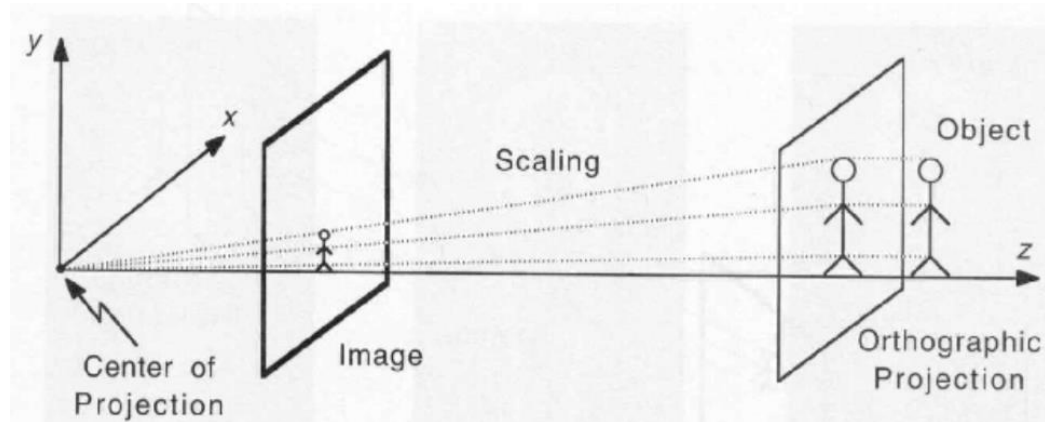
Camera projection matrix:

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

- 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 $P = [M \mid \mathbf{t}]$ 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:

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

- It has **six degrees of freedom**; one additional for the equal scale factors.
- A scaled orthographic projection matrix $P = [M \mid \mathbf{t}]$ 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:

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

A Hierarchy of Affine Cameras

3. Weak perspective projection

Camera projection matrix:

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

- It has **seven degrees of freedom**; one additional for the different scale factors.
- A weak perspective projection matrix $P = [M \mid \mathbf{t}]$ 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**:

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

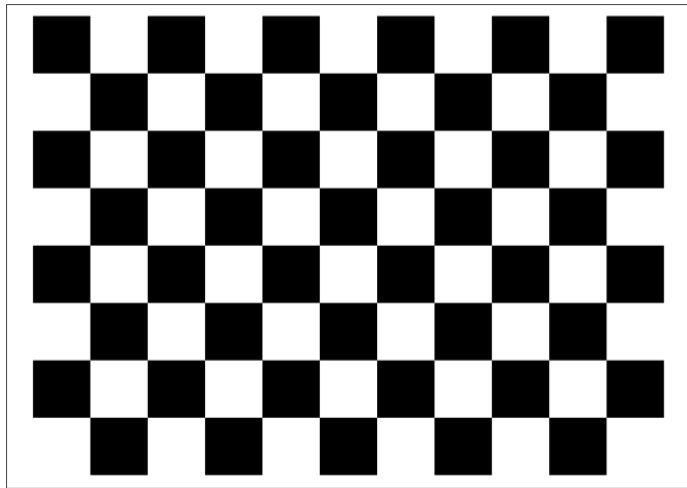
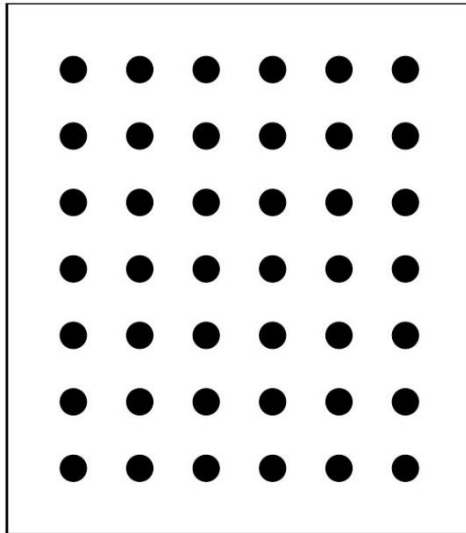
Component	# DOF	Elements	Known As
K	5	f_x, f_y, s, p_x, p_y	Intrinsic Parameters
R	3	α, β, γ	Extrinsic Parameters
\tilde{C} or t	3	(C_x, C_y, C_z) or (t_x, t_y, t_z)	

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 commonly used approach: Use a **2D calibration pattern** (e.g. a checkerboard).

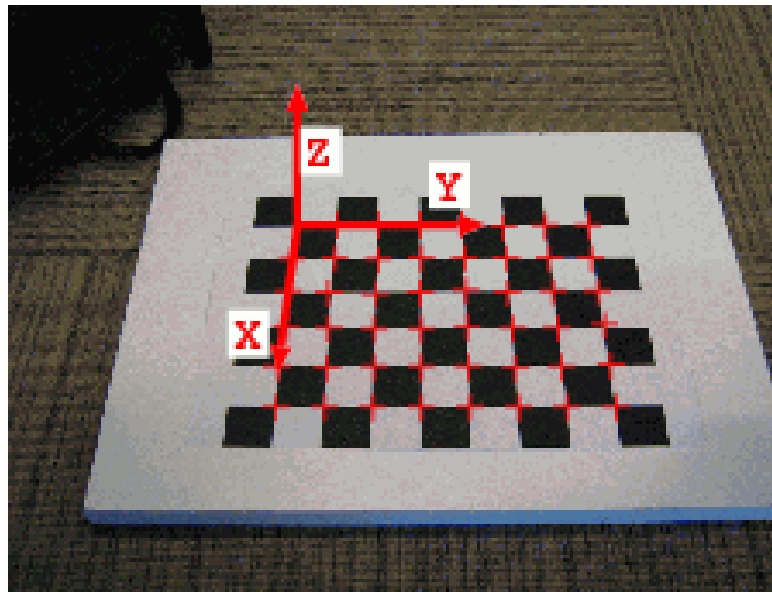


Camera Calibration: Open Source

- 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:

$$\text{Scale factor} \rightarrow s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}$$

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**:

$$s \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K \underbrace{\begin{bmatrix} r_1 & r_2 & t \end{bmatrix}}_{\text{Homography: } H = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix}} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \Rightarrow s \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} = K \begin{bmatrix} r_1 & r_2 & t \end{bmatrix}$$

where h_i is the i^{th} column H

Calibration of Projective Camera

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

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

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

$$r_1^T r_2 = 0 \Rightarrow h_1^T K^{-T} K^{-1} h_2 = 0 \quad (1)$$

$$\|r_1\| = \|r_2\| \Rightarrow h_1^T K^{-T} K^{-1} h_1 = h_2^T K^{-T} K^{-1} h_2 \quad (2)$$

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

Calibration of Projective Camera

- Let us denote:

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

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

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

Calibration of Projective Camera

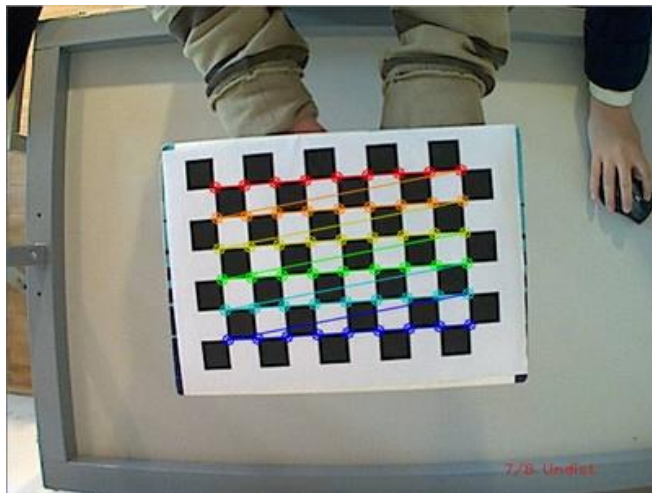
- Putting the 6-vector b into Equations (1) and (2).
- Re-arranging the homography terms, we get:

$$ab=0$$

- a is a 2×6 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.

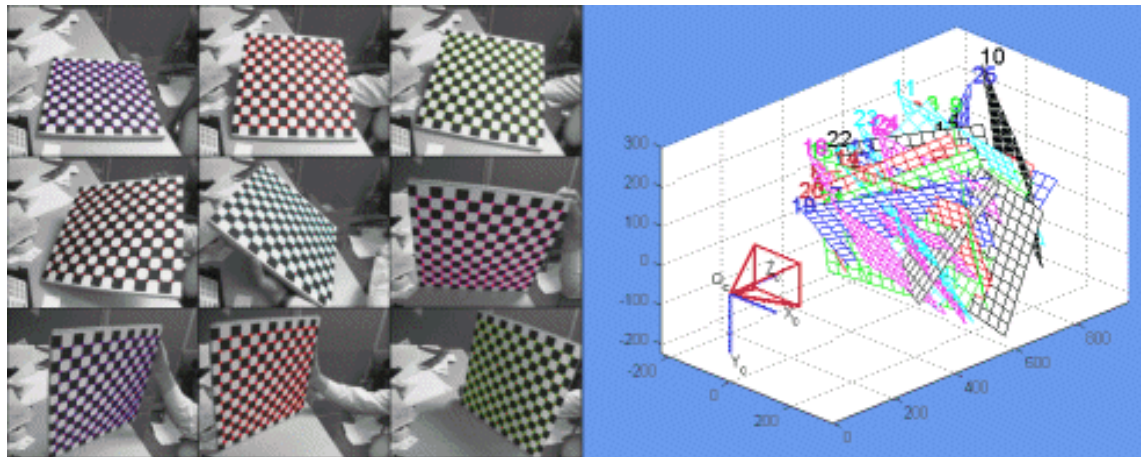


OpenCV detects the point correspondences automatically

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

Calibration of Projective Camera

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



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:

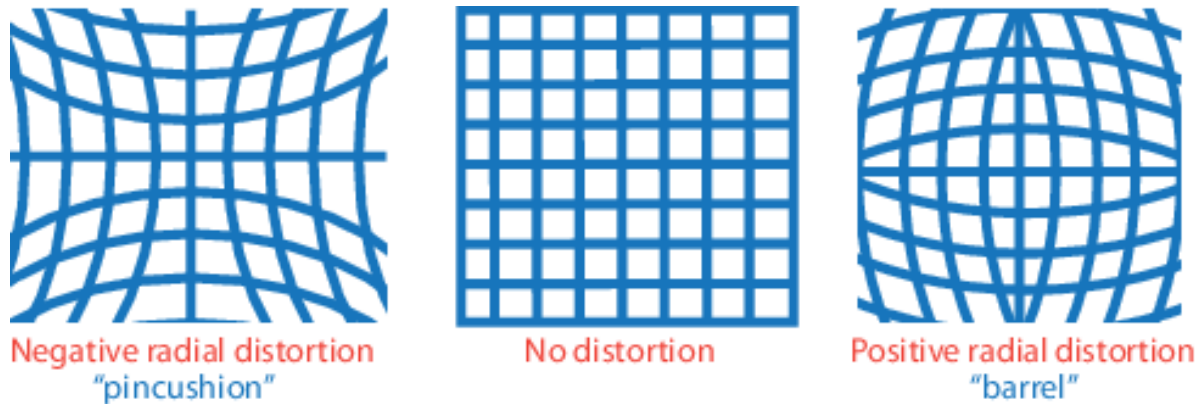
$$r_1 = sK^{-1}h_1, \quad r_2 = sK^{-1}h_2, \quad r_3 = r_1 \times r_2, \quad t = sK^{-1}h_3$$

where

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

Lens Distortion

1. Radial distortion (More common)



2. Tangential distortion (Less common)

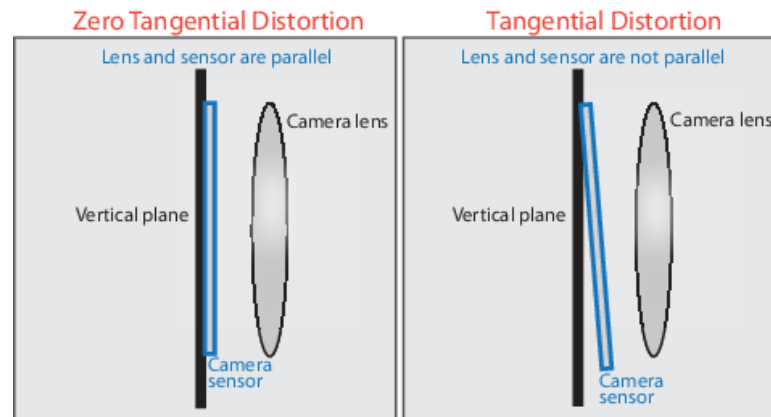


Image source: <http://www.mathworks.com/help/vision/ug/camera-calibration.html>

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:

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

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:

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

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:

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

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:

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

Use Levenberg-Marquardt to minimize this!

Lens Distortion: Maximum Likelihood Estimation

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

views $\rightarrow n$ m \leftarrow # 3D points

- 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
- $\pi(\cdot)$: projection function + lens distortion

$$\check{x}_{ij} = K[R_i \quad t_i]X_j \quad \longrightarrow \quad \pi(\cdot) = x_{dij} = x_r(\check{x}_{ij}) + dx(\check{x}_{ij})$$

Lens Distortion Correction

Before and after lens distortion correction



Summary

- We have looked at how to:
 1. Describe the **pinhole model** for camera projection with the camera **intrinsic** and **extrinsic** parameters.
 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.