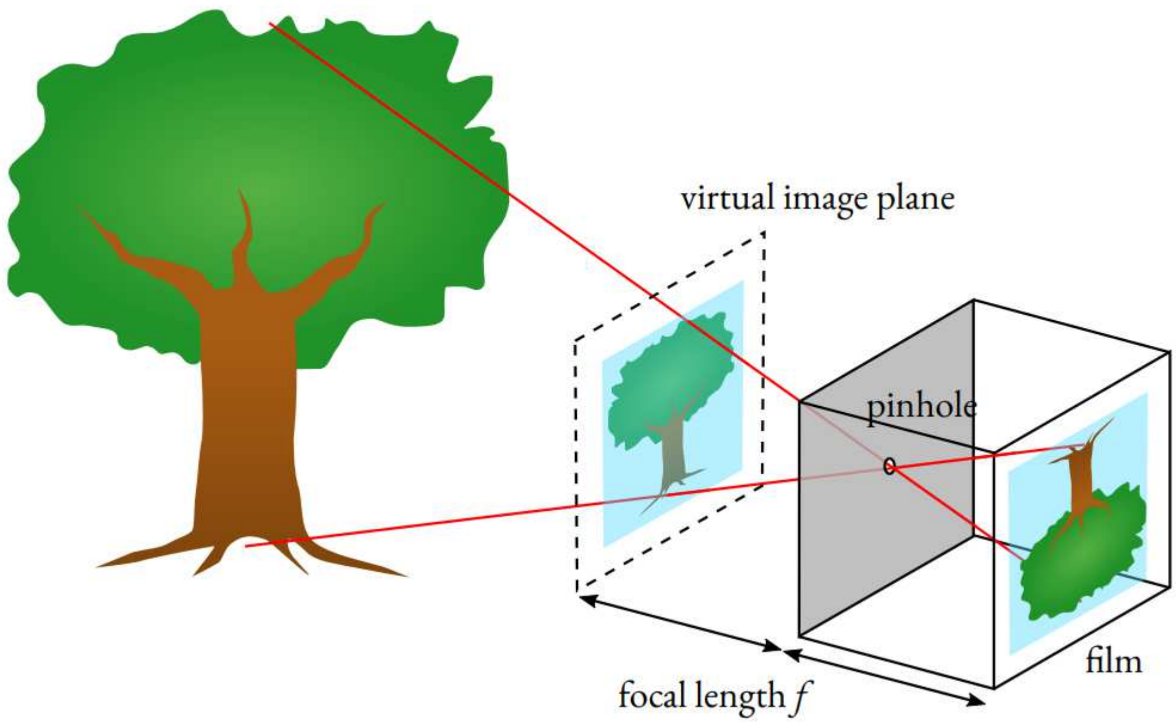
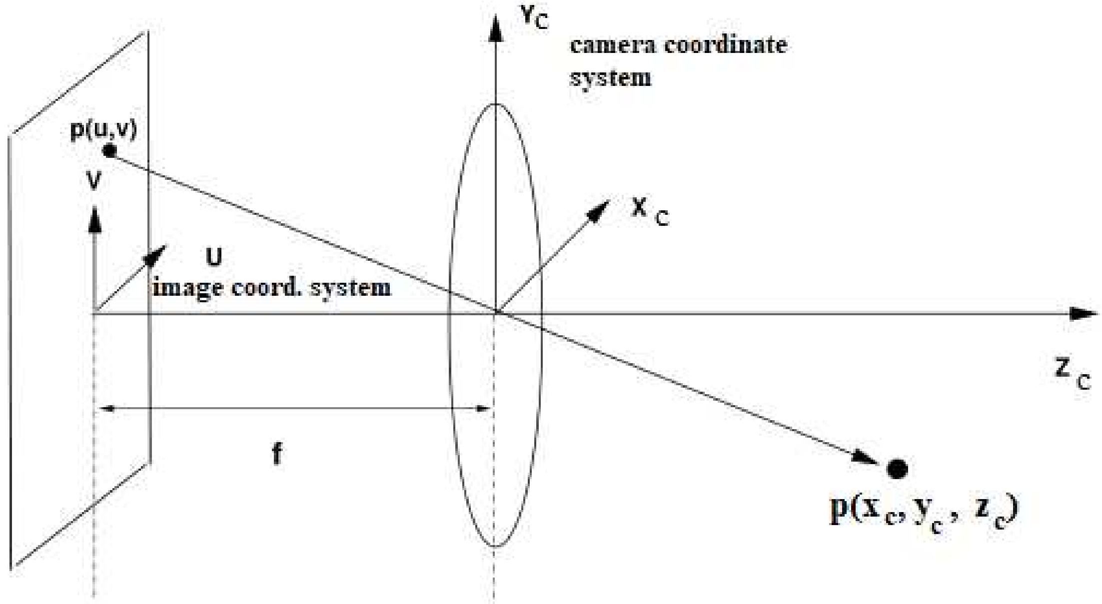
Camera Projection Models

We will introduce different camera projection models that relate the location of an image point to the coordinates of the corresponding 3D points. The projection models include: full perspective projection model, weak perspective projection model, affine projection model, and orthographic projection model.

The Pinhole Camera Model (cont'd)



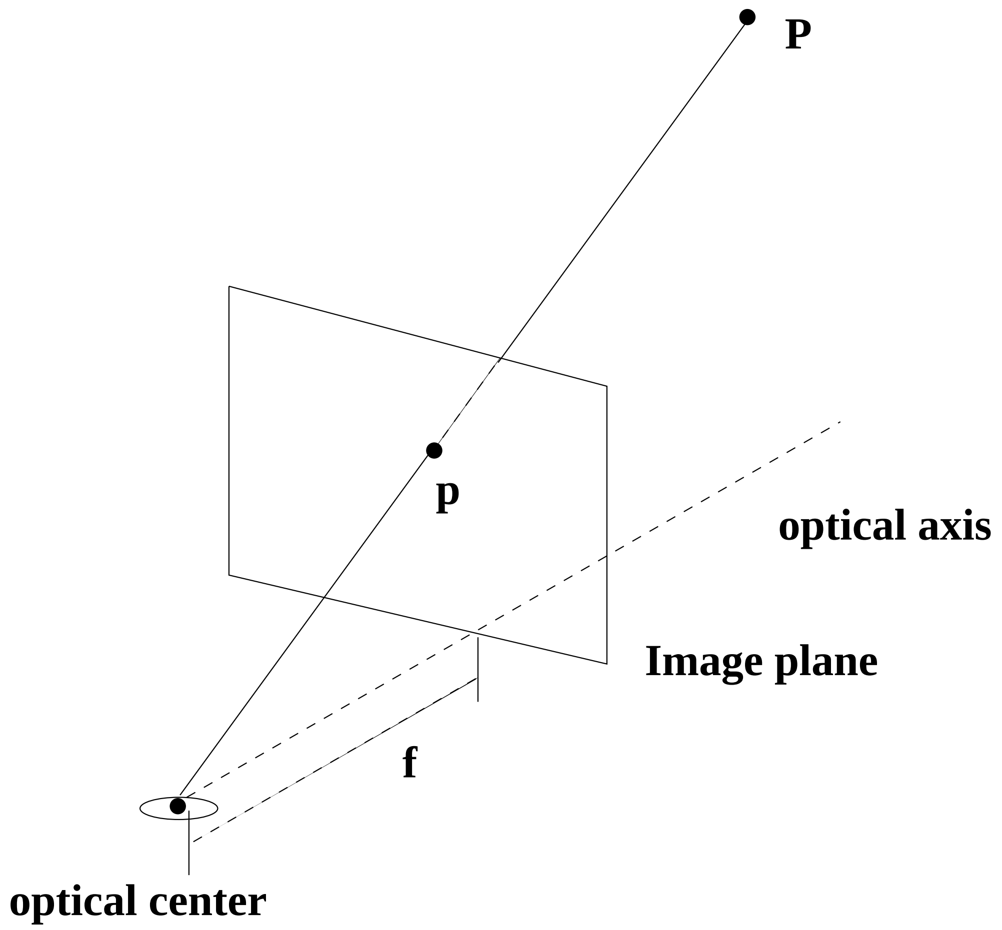
The Pinhole Camera Model



Based on simply trigonometry (or using 3D line equations), we can derive

$$u = \frac{-f x_c}{z_c} \qquad v = \frac{-f y_c}{z_c}$$

The Computer Vision Camera Model



$$u = \frac{f x_c}{z_c} \qquad v = \frac{f y_c}{z_c}$$

where $\frac{f}{z_c}$ is referred to as isotropic scaling.The full perspective projection is non-linear w.r.t 3D coordinates. Please note f is the image distance, i.e., the distance between the lens center and the image plane. It is **NOT** the focus length, though it is close to focus length, in particular for objects far away from camera (see thin lens property discussion).

Weak Perspective Projection

Let $z_c = \bar{z}_c + \delta z_c$, where $\bar{z}_c$ is the average distance to the camera and δz_c is the distance between the point and the center. If the relative distance δz_c (scene depth) is much smaller than the average distance $\bar{z}_c$ to the camera, i.e., $(\delta z < \frac{\bar{z}}{20})$, i.e, $z_c \approx \bar{z}_c$

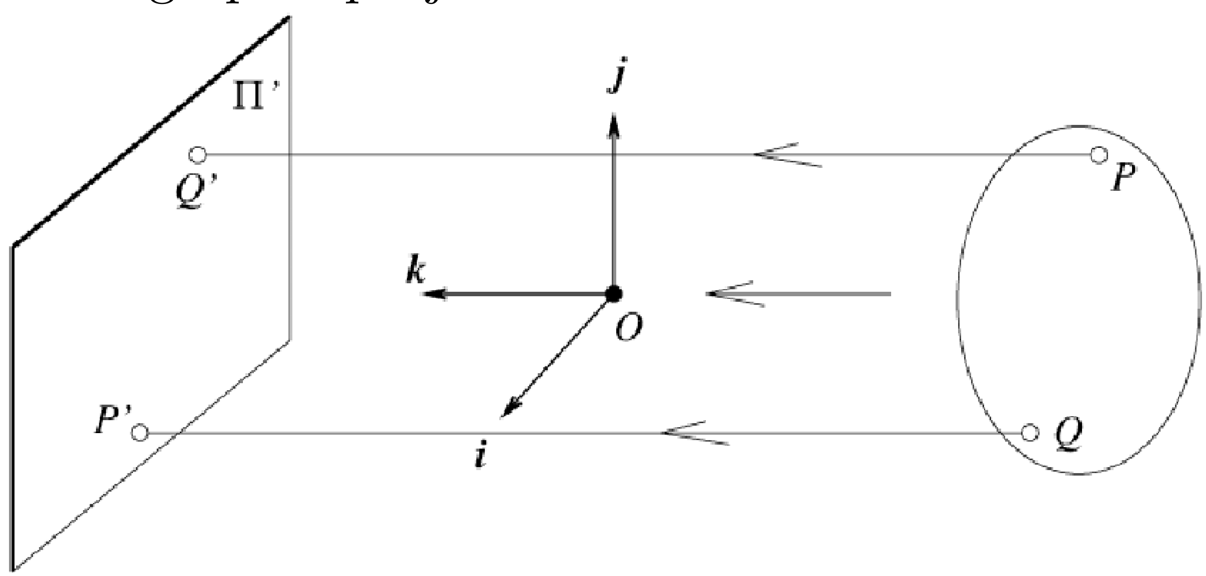
then

$$
\begin{aligned}
u &= f\frac{x_c}{z_c} \approx \frac{f x_c}{\bar{z}_c} \\
v &= f\frac{y_c}{z_c} \approx \frac{f y_c}{\bar{z}_c}
\end{aligned}
$$

We have linear equations since all projections have the same (yet unknown) scaling factor.

Orthographic Projection

As a special case of the weak perspective projection, when $\frac{f}{z_c}$ factor equals 1, we have $u = x_c$ and $v = y_c$, i.e., the lines (rays) of projection are parallel to the optical axis, i.e., the projection rays meet in the infinite instead of lens center (long f or telephoto). This leads to the sizes of image and the object are the same. This is called orthographic projection.



Note orthographic projection can also a special case of the scaled orthographic projection and para-perspective projection. Scaled orthographic projection scales orthographic projection by a constant scale factor s. Para-perspective projection have the projection lines parallel to each other. (see section 2.1.4 of [2] for details).

Perspective projection geometry

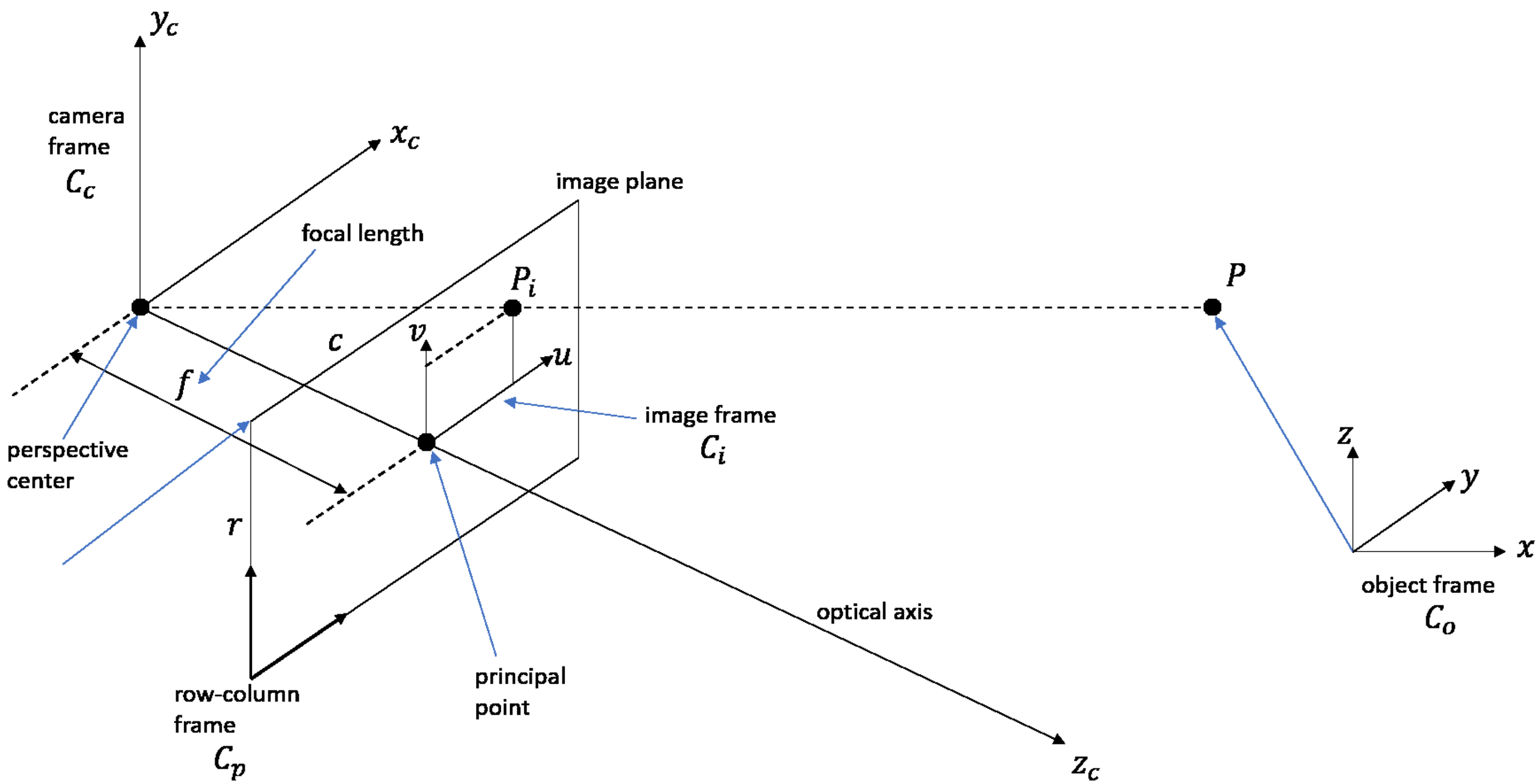


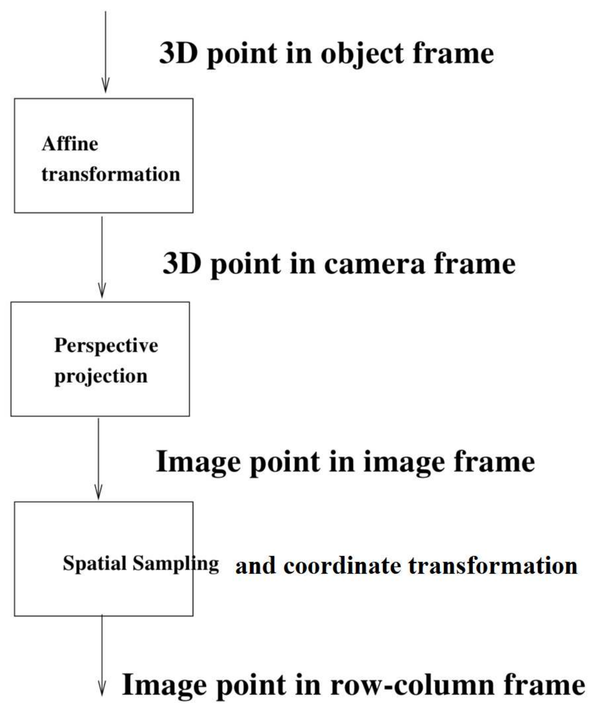
Figure 1: Perspective projection geometry

Notations

Let $P = (x\ y\ z)^T$ be a 3D point in object (world) frame and $U = (u\ v)^T$ the corresponding image point in the image frame before digitization. Let $X_c = (x_c\ y_c\ z_c)^T$ be the coordinates of P in the camera frame and $p = (c\ r)^T$ be the coordinates of U in the row-column frame after digitization.

Projection Process

Our goal is to go through the projection process to understand how an image point (c, r) is generated from the 3D point (x, y, z).



Relationships between different frames

Between camera frame (C_c) and object frame (C_o)

$$X_c = RX + T \qquad (1)$$

X is the 3D coordinates of P w.r.t the object frame. R is the rotation matrix and T is the translation vector. R and T specify the orientation and position of the object frame relative to the camera frame. They are often collectively called the pose of the object,

R and T can be parameterized as

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \qquad T = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

$r_i = (r_{i1}, r_{i2}, r_{i3})$ be a 1 x 3 row vector, R can be written as

$$R = \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \\ \mathbf{r}_3 \end{pmatrix}$$

Substituting the parameterized T and R into equation 1 yields

$$\begin{pmatrix} x_c \\ y_x \\ z_c \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (2)$$

- Between image frame (C_i) and camera frame (C_c)

  Perspective Projection:

$$\begin{aligned} u &= \frac{f x_c}{z_c} \\ v &= \frac{f y_c}{z_c} \end{aligned}$$

  Hence,

$$X_c = \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \lambda \begin{pmatrix} u \\ v \\ f \end{pmatrix} \tag{3}$$

  where $\lambda = \frac{z_c}{f}$ is a scalar and f is the camera focal length.

Relationships between different frames (cont'd)

- Between image frame (C_i) and row-col frame (C_p) (spatial quantization process)

$$\begin{pmatrix} c \\ r \end{pmatrix} = \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} c_0 \\ r_0 \end{pmatrix} \tag{4}$$

where s_x and s_y are scale factors ($pixels/mm$) due to spatial quantization. c_0 and r_0 are the coordinates of the principal point in pixels relative to C_p.

Collinearity Equations

Combining equations 1 to 4 yields

$$c = s_x f \frac{r_{11}x + r_{12}y + r_{13}z + t_x}{r_{31}x + r_{32}y + r_{33}z + t_z} + c_0$$

$$r = s_y f \frac{r_{21}x + r_{22}y + r_{23}z + t_y}{r_{31}x + r_{32}y + r_{33}z + t_z} + r_0$$

Homogeneous Coordinate System

In homogeneous coordinate system,

$$\begin{pmatrix} c \\ r \end{pmatrix} \text{ is changed to } \begin{pmatrix} c \\ r \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \text{ is changed to } \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

Homogeneous system: perspective projection

In homogeneous coordinate system, equation 3 may be rewritten as

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \tag{5}$$

Note $\lambda = z_c$.

Homogeneous System: Spatial Quantization

Similarly, in homogeneous system, equation 4 may be rewritten as

$$\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x & 0 & c_0 \\ 0 & s_y & r_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \tag{6}$$

Homogeneous system: quantization + projection

Substituting equation 5 into equation 6 yields

$$
\begin{aligned}
\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} &= \begin{pmatrix} s_x f & 0 & c_0 \\ 0 & s_y f & r_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \\
&= W \begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} \qquad (7)
\end{aligned}
$$

where $\lambda = z_c$.

Homogeneous system: Affine Transformation

In homogeneous coordinate system, equation 2 can be expressed as

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{8}$$

Homogeneous system: full perspective

Combining equation 8 with equation 7 yields

$$\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} s_x f \mathbf{r}_1 + c_0 \mathbf{r}_3 & s_x f t_x + c_0 t_z \\ s_y f \mathbf{r}_2 + r_0 \mathbf{r}_3 & s_y f t_y + r_0 t_z \\ \mathbf{r}_3 & t_z \end{pmatrix}}_{P} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{9}$$

where $\mathbf{r}_1$, $\mathbf{r}_2$, and $\mathbf{r}_3$ are the row vectors of the rotation matrix R, $\lambda = z_c$ is a scaler and matrix P is called the homogeneous projection matrix.

$$P = WM$$

where

$$W = \begin{pmatrix} fs_x & 0 & c_0 \\ 0 & fs_y & r_0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$M = \begin{pmatrix} R & T \end{pmatrix}$$

W is often referred to as the intrinsic matrix and M as exterior matrix.

Since $P = WM = [WR \;\; WT]$, for P to be a projection matrix, $Det(WR) \neq 0$, i.e., $Det(W) \neq 0$.

Full Perspective Projection Camera Model

Eq. 9 can be alternatively re-written as

$$\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x f \mathbf{r}_1 + c_0 \mathbf{r}_3 & s_x f t_x + c_0 t_z \\ s_y f \mathbf{r}_2 + r_0 \mathbf{r}_3 & s_y f t_y + r_0 t_z \\ \mathbf{r}_3 & t_z \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} s_x f \mathbf{r}_1 & c_0 \mathbf{r}_3 \mathbf{X} + s_x f t_x + c_0 t_z \\ s_y f \mathbf{r}_2 & r_0 \mathbf{r3X} + s_y f t_y + r_0 t_z \\ \mathbf{0}^{1\times 3} & \mathbf{r}_3 \mathbf{X} + t_z \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} s_x f \mathbf{r}_1 & s_x f t_x + c_0 z_c \\ s_y f \mathbf{r}_2 & s_y f t_y + r_0 z_c \\ \mathbf{0}^{1\times 3} & z_c \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{10}$$

where $\mathbf{X} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$

Weak Perspective Camera Model

For weak perspective projection, we have $z_c \approx \bar{z}_c$, i.e.,

$$\bar{z}_c \approx z_c = \mathbf{r}_3 \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t_z = \mathbf{r}_3 \mathbf{X} + t_z \tag{11}$$

Substituting $z_c \approx z_c$ into Eq. 10 yields

$$\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x f \mathbf{r}_1 & c_0 \bar{z}_c + s_x f t_x \\ s_y f \mathbf{r}_2 & r_0 \bar{z}_c + s_y f t_y \\ \mathbf{0}^{1\times 3} & \bar{z}_c \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{12}$$

Weak Perspective Camera Model

The weak perspective projection matrix is

$$P_{weak} = \begin{pmatrix} fs_x\mathbf{r}_1 & fs_xt_x + c_0\bar{z}_c \\ fs_y\mathbf{r}_2 & fs_yt_y + r_0\bar{z}_c \\ \mathbf{0}^{1\times3} & \bar{z}_c \end{pmatrix} \tag{13}$$

where $\mathbf{r}_1$ and $\mathbf{r}_2$ are the first two rows of R , $\bar{z}_c = \mathbf{r}_3\mathbf{X} + t_z$, and $\lambda = \bar{z}_c$.

Orthographic Projection Camera Model

Under orthographic projection, projection is parallel to the camera optical axis, we have

$$\begin{aligned} u &= x_c \\ v &= y_c \end{aligned}$$

which can be approximated by $\frac{f}{z_c} \approx 1$.

Dividing both sides of Eq. 10 by $\lambda = z_c$ and applying $\frac{f}{z_c} = 1$ yields

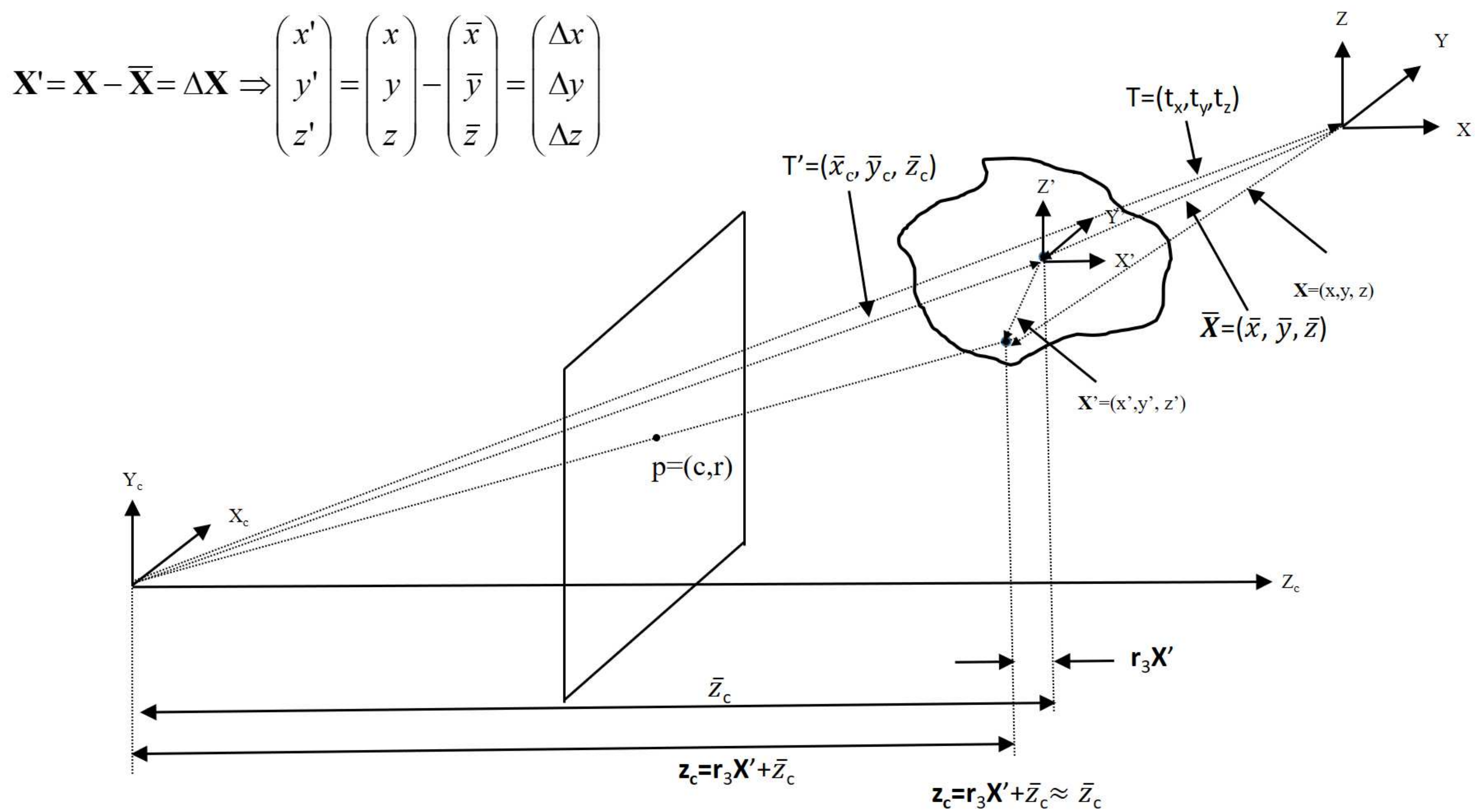
$$\begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x \mathbf{r}_1 & c_0 + s_x t_x \\ s_y \mathbf{r}_2 & r_0 + s_y t_y \\ \mathbf{0}^{1\times 3} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{14}$$

The orthographic projection matrix can therefore be obtained as

$$P_{orth} = \begin{pmatrix} s_x \mathbf{r}_1 & c_0 + s_x t_x \\ s_y \mathbf{r}_2 & r_0 + s_y t_y \\ \mathbf{0}^{1\times 3} & 1 \end{pmatrix} \tag{15}$$

Affine Camera Model

A further simplification from weak perspective camera model is the affine camera model, which is often assumed by computer vision researchers due to its simplicity. The affine camera model assumes that the object frame is located on the centroid of the object being observed and that the projection follows the weak perspective projection.



As shown in the figure, object frame located at the object centroid implies $t_x = \bar{x}_c$, $t_y = \bar{y}_c$, and $t_z = \bar{z}_c$, where $(\bar{x}_c, \bar{y}_c, \bar{z}_c)$ are the coordinates of the object centroid relative to the camera frame. Weak perspective projection means $z_c \approx \bar{z}_c$. Substituting them into Eq. 10 yields

$$\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} s_x f \mathbf{r}_1 & c_0 \bar{z}_c + s_x f \bar{x}_c \\ s_y f \mathbf{r}_2 & r_0 \bar{z}_c + s_y f \bar{y}_c \\ \mathbf{0}^{1\times 3} & \bar{z}_c \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{16}$$

Hence

$$P_{affine} = \begin{pmatrix} s_x f \mathbf{r}_1 & s_x f \bar{x}_c + c_0 \bar{z}_c \\ s_y f \mathbf{r}_2 & s_y f \bar{y}_c + r_0 \bar{z}_c \\ 0 & \bar{z}_c \end{pmatrix} \tag{17}$$

Affine Camera Model

Affine camera model represents the first order approximation of the full perspective projection camera model around the object central point $(\bar{x}_c, \bar{y}_c, \bar{z}_c)$.

$$\lambda\begin{pmatrix}c\\r\\1\end{pmatrix}=\mathbf{WRX}+\mathbf{WT} \qquad \begin{pmatrix}c\\r\end{pmatrix}=\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{RX}+\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{T}}{\mathbf{r}_3\mathbf{X}+\mathbf{w}_3\mathbf{T}}=f(\mathbf{X})$$

First order Talyor expansion wrt to $\overline{\mathbf{X}}$

$$\begin{pmatrix}c\\r\end{pmatrix}=f(\overline{\mathbf{X}})+\frac{\partial f(\overline{\mathbf{X}})}{\partial\mathbf{X}}\Delta\mathbf{X}$$

$$=\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{R}\overline{\mathbf{X}}+\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{T}}{\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T}}+\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{R}}{\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T}}\Delta\mathbf{X}-\left(\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{R}\overline{\mathbf{X}}\mathbf{r}_3}{(\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T})^2}+\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{T}\mathbf{r}_3}{(\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T})^2}\right)\Delta\mathbf{X}$$

$$=\begin{pmatrix}\bar{c}\\\bar{r}\end{pmatrix}+\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{R}}{\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T}}\Delta\mathbf{X}-\left(\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{R}\overline{\mathbf{X}}\mathbf{r}_3^{T}}{(\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T})^2}+\frac{\begin{bmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{bmatrix}\mathbf{T}\mathbf{r}_3^{T}}{(\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T})^2}\right)\Delta\mathbf{X}$$

$$=\begin{pmatrix}\bar{c}\\\bar{r}\end{pmatrix}+\frac{\begin{bmatrix}\mathbf{I}_2 & -\begin{pmatrix}\bar{c}\\\bar{r}\end{pmatrix}\end{bmatrix}\begin{bmatrix}\begin{pmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{pmatrix}\mathbf{R}\\\mathbf{r}_3\end{bmatrix}\Delta\mathbf{X}}{\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T}}\Rightarrow\begin{pmatrix}\Delta c\\\Delta r\end{pmatrix}=\begin{pmatrix}c-\bar{c}\\r-\bar{r}\end{pmatrix}=\frac{\begin{bmatrix}\mathbf{I}_2 & -\begin{pmatrix}\bar{c}\\\bar{r}\end{pmatrix}\end{bmatrix}\begin{bmatrix}\begin{pmatrix}\mathbf{w}_1\\\mathbf{w}_2\end{pmatrix}\mathbf{R}\\\mathbf{r}_3\end{bmatrix}\Delta\mathbf{X}}{\mathbf{r}_3\overline{\mathbf{X}}+\mathbf{w}_3\mathbf{T}}$$

as shown in the figure, $(\bar{c}, \bar{r})$ are the projection of $\bar{X}$ and $\Delta X = \mathbf{X}'$ are the 3D coordinates relative to the object frame centered in the object centroid. It approximates well when object is far from camera to satisfy $z_c \approx \bar{z}_c$. It is no longer useful when the object is close to the camera or the camera has a wide angle of view.

Non-full perspective Projection Camera Model

The weak perspective projection, affine, and orthographic camera model can be collectively classified as *non-perspective projection* camera model. In general, the projection matrix for the non-perspective projection camera model, where p_{34} is a constant for all points.

$$\lambda \begin{pmatrix} c \\ r \\ 1 \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ 0 & 0 & 0 & p_{34} \end{pmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix}$$

Dividing both sides by p_{34} (note $\lambda = p_{34}$) yields

$$\begin{pmatrix} c \\ r \end{pmatrix} = M_{2\times 3} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} v_x \\ v_y \end{pmatrix}$$

where $m_{ij} = p_{ij}/p_{34}$ and $v_x = p_{14}/p_{34}$, $v_y = p_{24}/p_{34}$

For any given reference point (c_r, r_r) in image and (x_r, y_r, z_r) in space, the relative coordinates (c', r') in image and (x', y', z') in space are

$$\begin{pmatrix} c' \\ r' \end{pmatrix} = \begin{pmatrix} c - c_r \\ r - r_r \end{pmatrix} \text{ and } \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} x - x_r \\ y - y_r \\ z - z_r \end{pmatrix}$$

It follows that the basic projection equation for the affine and weak perspective model in terms of relative coordinates is

$$\begin{pmatrix} c' \\ r' \end{pmatrix} = M_{2\times 3} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix}$$

An non-perspective projection camera $M_{2\times 3}$ has 3 independent parameters. The reference point is often chosen as the centroid since centroid is preserved under either affine or weak perspective projection.

Given the weak projection matrix P,

$$P = \begin{pmatrix} f s_x \mathbf{r}_1 & f s_x t_x + c_0 \bar{z}_c \\ f s_y \mathbf{r}_2 & f s_y t_y + r_0 \bar{z}_c \\ 0 & \bar{z}_c \end{pmatrix}$$

The M matrix is

$$
\begin{aligned}
M &= \begin{pmatrix} \frac{f s_x \mathbf{r}_1}{\bar{z}_c} \\ \frac{f s_y \mathbf{r}_2}{\bar{z}_c} \end{pmatrix} \\
&= \frac{f}{\bar{z}_c} \begin{pmatrix} s_x \mathbf{r}_1 \\ s_y \mathbf{r}_2 \end{pmatrix} \\
&= \frac{f}{\bar{z}_c} \begin{pmatrix} s_x & 0 \\ 0 & s_y \end{pmatrix} \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{pmatrix}
\end{aligned}
$$

For affine projection, $\bar{z}_c = t_z$, for orthographic projection, $\frac{f}{\bar{z}_c} = 1$. If we assume $s_x = s_y$, then

$$
M = \frac{f s_x}{\bar{z}_c} \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{pmatrix}
$$

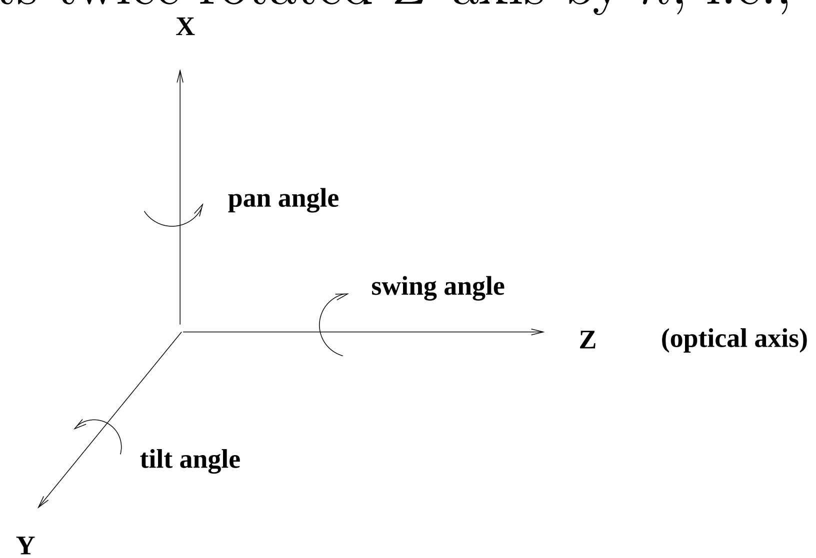
Then, we have only four parameters: three rotation angles and a scale factor.

Rotation Matrix Representation: Euler angles

Rotation matrix R is a 3×3 matrix but it has only 3 degrees of freedom (DOF). The 3 DOFs can be represented differently.

Rotation Matrix Representation: Euler angles

Assume rotation matrix R results from successive Euler rotations of the camera frame around its X axis by ω, its once rotated Y axis by ϕ, and its twice rotated Z axis by κ, i.e.,



$$R(\omega, \phi, \ \kappa) = R_X(\omega)R_Y(\phi)R_Z(\kappa)$$

where ω, ϕ, and κ are often referred to as pan, tilt, and swing angles respectively.

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

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

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

Rotation Matrix: Rotation by a general axis

Let the general axis be $\omega = (\omega_x, \omega_y, \omega_z)$ and the rotation angle be θ. The rotation matrix R resulting from rotating around ω by θ can be expressed by the Rodrigues' rotation formula

$$\begin{bmatrix} \cos\theta + \omega_x^2(1-\cos\theta) & \omega_x\omega_y(1-\cos\theta) - \omega_z\sin\theta & \omega_y\sin\theta + \omega_x\omega_z(1-\cos\theta) \\ \omega_z\sin\theta + \omega_x\omega_y(1-\cos\theta) & \cos\theta + \omega_y^2(1-\cos\theta) & -\omega_x\sin\theta + \omega_y\omega_z(1-\cos\theta) \\ -\omega_y\sin\theta + \omega_x\omega_z(1-\cos\theta) & \omega_x\sin\theta + \omega_y\omega_z(1-\cos\theta) & \cos\theta + \omega_z^2(1-\cos\theta) \end{bmatrix}$$

It gives an efficient method for computing the rotation matrix. The DOF remains 3 as $||\omega||_2 = 1$.

Quaternion Representation of R

The relationship between a quaternion $q = [q_0, q_1, q_2, q_3]$ and the equivalent rotation matrix is

$$R = \begin{pmatrix} q_0q_0 + q_1q_1 - q_2q_2 - q_3q_3 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_2q_1 + q_0q_3) & q_0q_0 - q_1q_1 + q_2q_2 - q_3q_3 & 2(q_2q_3 - q_0q_1) \\ 2(q_3q_1 - q_0q_2) & 2(q_3q_2 + q_0q_1) & q_0q_0 - q_1q_1 - q_2q_2 + q_3q_3 \end{pmatrix}.$$

Here the quaternion is assumed to have been scaled to unit length, i.e., $||q||_2 = 1$ to ensure the DOF to be 3.

The axis/angle representation ω/θ is strongly related to a quaternion, according to the formula

$$\begin{pmatrix} \cos(\theta/2) \\ \omega_x \sin(\theta/2) \\ \omega_y \sin(\theta/2) \\ \omega_z \sin(\theta/2) \end{pmatrix}$$

Among the three representations, Euler angles are the minimal but it is least preferred as the results depend on the rotation orders and the motion is not smooth in the parameter space. The general axis/angle is complex and highly non-linear. The quaternion is often used and it is better for smooth motion.

R's Orthnormality

The rotation matrix is an orthnormal matrix, which means its rows (columns) are normalized to one and they are orthonal to each other. The orthnormality property produces

$R^T = R^{-1}$

Interior Camera Parameters

Parameters (c_0, r_0), s_x, s_y, and f are collectively referred to as *interior camera parameters*. They do not depend on the position and orientation of the camera. Interior camera parameters allow us to perform metric measurements, i.e., to convert pixel measurements to inch or mm.

Exterior Camera Parameters

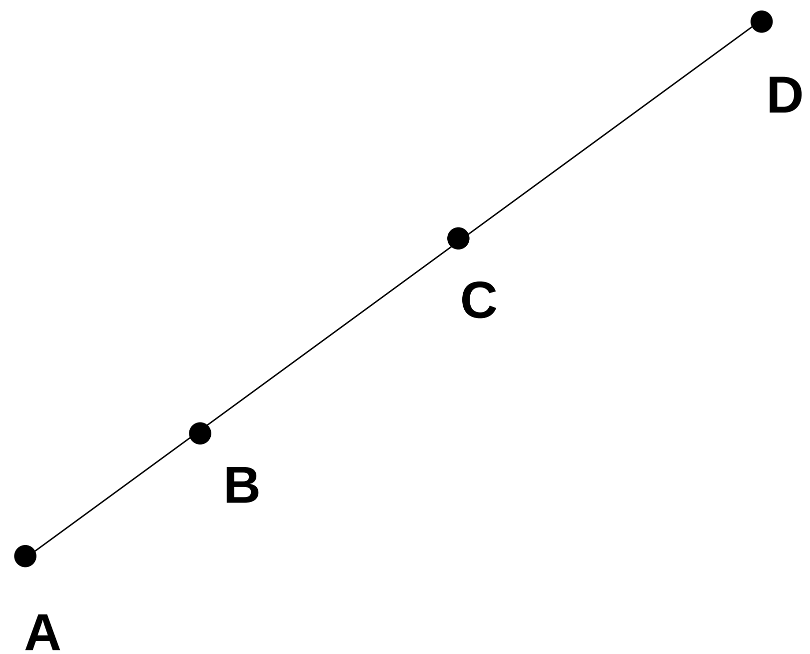
Parameters like Euler angles ω, ϕ, κ, t_x, t_y, and t_z are collectively referred to as *exterior camera parameters*. They determine the position and orientation of the camera.

Perspective Projection Invariants

Certain geometric properties are not preserved under full perspective projection. Distances and angles are no longer preserved under perspective projection.

Some properties are invariant under full perspective projection. Perspective projection of 3D lines remain 2D lines. Projection 3D conic curves remain 2D conic curves though 3D circle projection may no longer be a 2D circle.

The most important invariant with respect to perspective projection is called *cross ratio*. It is defined as follows:

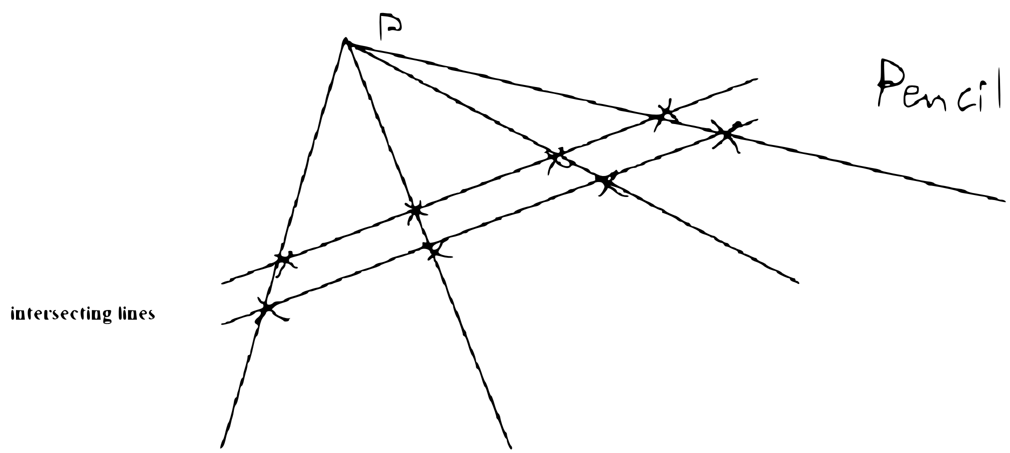


$$\tau(A,B,C,D) = \frac{AC}{BC} \Big/ \frac{AD}{BD}$$

Cross-ratio is preserved under perspective projection.

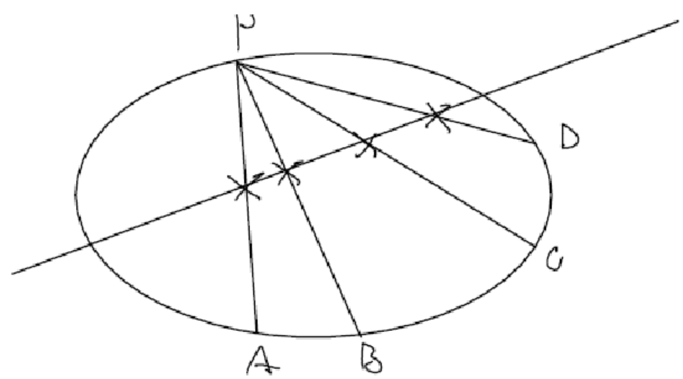
Cross ratios for pencil lines

Cross ratio of intersection points between 4 pencil lines and another intersecting line are only function of the angles among the pencil lines, independent of the interacting line. cross-ratio may be used for ground plane detection from multiple image frames.



Cross ratios for conic points

Chasles' theorem: A, B, C, D are distinct points on a (non-singular) conic (ellipse, circle, ..) and P is another point on the conic, the cross-ratio of intersections points on the pencil PA, PB, PC, PD does not depend on the point P. This means given A,B,C, and D, all points P on the same ellipse should yield the same cross ratios. This theorem may be used for ellipse detection.



See section 19.3 and 19.4 of [1].

References

[1] E Roy Davies. *Computer vision: principles, algorithms, applications, learning*. Academic Press, 2017.

[2] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Science & Business Media, 2021.