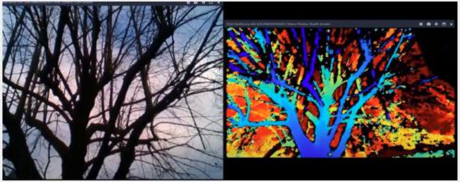
Autonomous Mobile Robotics

Multi-view Geometry

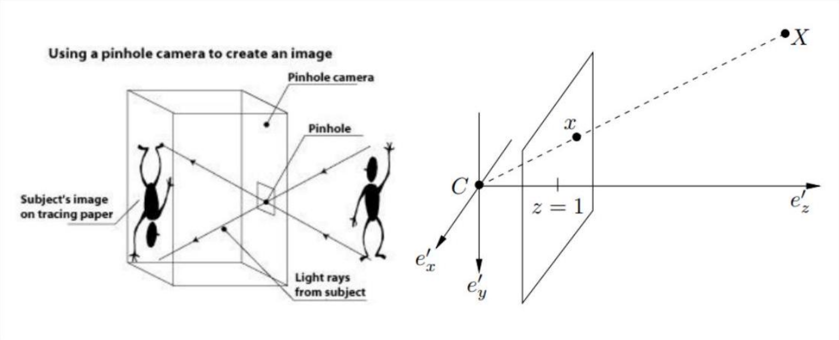
Geesara Kulathunga

February 8, 2023



Contents

Pinhole camera model



http://www.ctr.maths.lu.se/media/FMA270/2015/alllectures.pdf

- Idea is that light rays enter through **a small hole (the pinhole)** and project an image on **the back of the camera wall**
- If camera coordinate system, defined as $\left\{e'_x, e'_y, e'_z\right\}$. The coordinate of **camera center** or **pinhole** of the camera(C) is at $(0, 0, 0)$
- The projection of $\mathbf{X} = (X_w, Y_w, Z_w)$ scene point into the image plane $\mathbf{x}' = (x', y', z')$ while assuming $z' = 1$ has the **normal** e_z lies at the distance 1 from the camera center. e_z can be defined as the viewing direction since the $\mathbf{X} - \mathbf{C}$ is the direction vector of viewing ray

$$\mathbf{C} + s(\mathbf{X} - \mathbf{C}) = \mathbf{sX}, \mathbf{s} \in \mathbb{R}$$

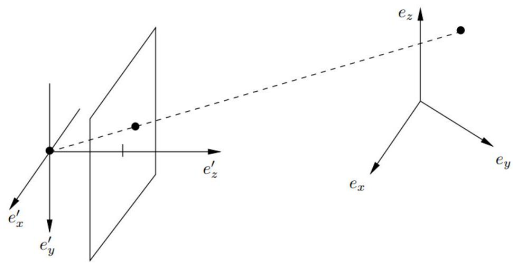
Thus, where will the intersection of this vector be, if $e_z = 1$?

$$\mathbf{x}' = \begin{bmatrix} X_w/Z_w \\ Y_w/Z_w \\ 1 \end{bmatrix}$$

Example 01

Compute the projection of the cube with corners: $(\pm 1, \pm 1, 2)$ and $(\pm 1, \pm 1, 4)$ in image plane?

Global coordinate system and camera coordinate system



In real-world examples, the camera can **undergo a series of rotations and translations**. Hence, it is required to transform the **world coordinate system into a camera coordinate system**.
http://www.ctr.maths.lu.se/media/FMA270/2015/alllectures.pdf

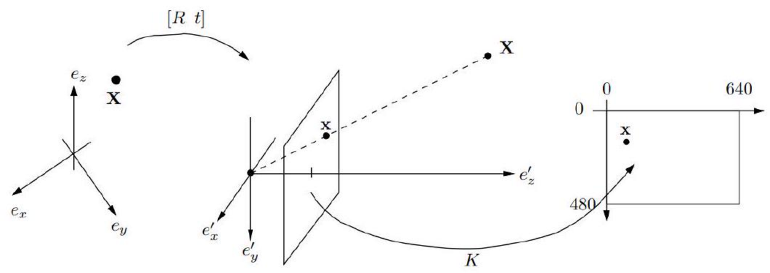
A given point in the global coordinate system can be represented with respect to the camera coordinate system:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = [Rt] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

Example 02

Compute the projection of $\mathbf{X} = (0, 0, 1)$ in the cameras coordinate system if R is $\begin{bmatrix} 1 & 0 & -1 \\ 0 & \sqrt{2} & 0 \\ 1 & 0 & 1 \end{bmatrix}$ and t vector equals to $[0, 0, \sqrt{2}]$.
Also, how do you assume for a given point is in the front of the camera or not?

Global coordinate system and camera coordinate system



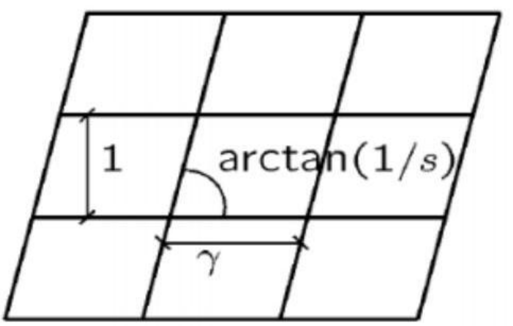
http://www.ctr.maths.lu.se/media/FMA270/2015/alllectures.pdf

In the image plane, the **center of the image** is located in, i.e., 0,0,? $(c_x, c_y, 0)$. However, in the camera plane, 0,0 starts from the upper left corner. This transformation is given by the camera matrix, i.e., the inner parameters of the camera. This transformation matrix is denoted as K where it is **invertible**. In general, K is expressed as:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & c_x \\ 0 & 1 & c_y \\ 0 & 0 & 1 \end{bmatrix}}_{2d \text{ translation}} \times \underbrace{\begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{2d \text{ scaling}} \times \underbrace{\begin{bmatrix} 1 & s/f_y & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{2d shear}}$$

where f is called focal length, c_x and c_y is denoted the principle point of the camera, γ is the aspect ratio.

The skew parameter (s) corrects non-rectangular pixels and γ is used correct the aspect ratio issue



When the **pixels** are **not square values**, γ will not be equal to one. Otherwise, it will be equal to 1. The final parameter is s which is defined as **skew**. This parameter is used to tilt the pixels

Projection matrix

The relationship between a point in the camera and in the world:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K[R \mid t]X = PX$$

where $R \mid t$ is the homogeneous transformation which is composed out of a rotation matrix R, and a translation vector t.

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}.$$

All in all, we can define the projective transformation that maps world coordinates points in $\mathbf{R}^3$ to $\mathbf{R}^2$ image coordinate system followed by normalized camera coordinate system.

Projective transformation

The projective transformation that maps world coordinates points in $\mathbf{R}^3$ to $\mathbf{R}^2$ image coordinate system followed by normalized camera coordinate system.

$$Z_c \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = [R \mid t] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix},$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} f_x X_c/Z_c + c_x \\ f_y Y_c/Z_c + c_y \end{bmatrix}$$

where $x' = X_c/Z_c$ and $y' = Y_c/Z_c$.

Example 03

How we are going to find the λ, K, R and t? Which one of these belongs to intrinsic parameters? As stated before, the task is to find the projection matrix P. Thus, how many unknowns do we have and how many equations do we need to solve in this problem?

Finding P using the direct linear transformation (DLT)

Example 03

Let's say we have N number of points in which correspondence is known between the world and the camera frame.

$$\lambda_i x_i = PX_i, \quad i = 1, ..N$$

In order to find P, can you try to derive an expression for minimum value for N to be satisfied? And prove that N should be equal to or higher than 6.

Finding P using the direct linear transformation (DLT)

Example 03

Let $p_i, i = 1, 2, 3$ be vectors containing the rows of P, that is,

$$P = \begin{bmatrix} p_1^T \\ p_2^T \\ p_3^T \end{bmatrix}$$

then, Equ. 9 can be reformulated as follows:

$$\begin{aligned} X_i^\top p_1 - \lambda_i x_i &= 0 \\ X_i^\top p_2 - \lambda_i y_i &= 0 \\ X_i^\top p_3 - \lambda_i &= 0 \end{aligned}$$

Example 03

Can you convert the previous formulation into a matrix form?

$$\underbrace{\begin{bmatrix} X_1^T & 0 & 0 & -x_1 & 0 & 0 & \cdots \\ 0 & X_1^T & 0 & -y_1 & 0 & 0 & \cdots \\ 0 & 0 & X_1^T & -1 & 0 & 0 & \cdots \\ X_2^T & 0 & 0 & 0 & -x_2 & 0 & \cdots \\ 0 & X_2^T & 0 & 0 & -y_2 & 0 & \cdots \\ 0 & 0 & X_2^T & 0 & -1 & 0 & \cdots \\ X_3^T & 0 & 0 & 0 & 0 & -x_3 & \cdots \\ 0 & X_3^T & 0 & 0 & 0 & -y_3 & \cdots \\ 0 & 0 & X_3^T & 0 & 0 & -1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{M} \underbrace{\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \vdots \end{bmatrix}}_{v} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \vdots \end{pmatrix}$$

Finding P using the direct linear transformation (DLT)

Example 03

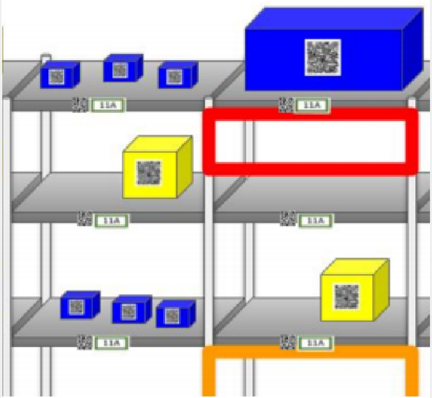
In order to find vector v, we have to find the **null space vector of M**. Basically, need to solve system $Mv = 0$. Can we actually solve this? I would say no! what are you up to?

Camera Calibration

Camera calibration is all about finding project matrix ($P = K[Rt]$). More information can be found here: `https://www.mathworks.com/help/vision/camera-calibration.html` or `http://wiki.ros.org/camera_calibration`.

Depth estimation using monocular camera

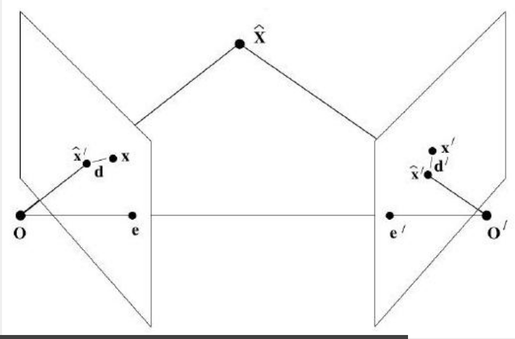
Can you use monocular camera for depth estimation?



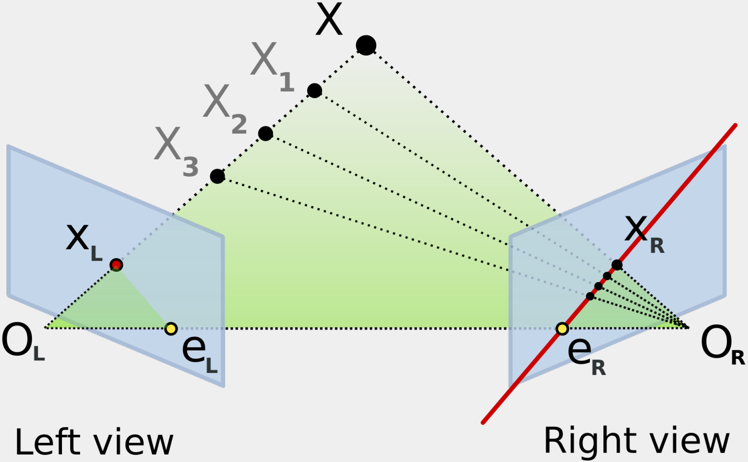
https://www.semanticscholar.org/paper/
Warehouse-Management-Using-Real-Time-QR-Code-and-Saha-Udayagiri/
4c6c478b7ba8c46dca35dcba5d69648610c2742b

Simple Stereo

If the **camera matrices are known** (the triangulation problem) Direct Linear Transformation (**DLT**) to find the projection matrix (**P**). On the contrary, if the scene points and camera matrices are not known problems get complicated. The main intuition is to find **some similarities between considered two images** where part of those are overlapping each other. The technique used to solve this problem is called epipolar geometry.

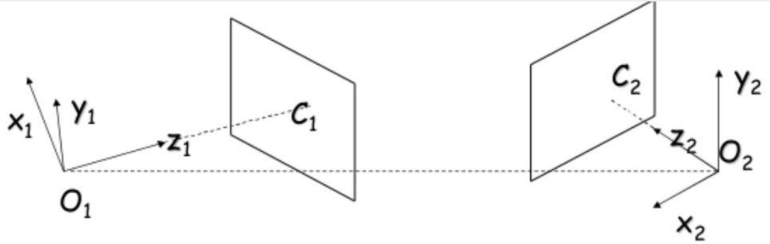


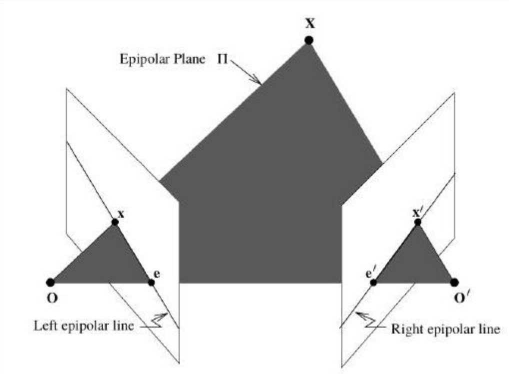
If both projection matrices, i.e., P_1 and P_2, are known, how can we estimate the $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ for a known $\hat{\mathbf{X}}$?



https://en.wikipedia.org/wiki/Epipolar_geometry

In general, there are two types of problems that belong to general stereo: matrix K is known, need to find $[\mathrm{R}t]$ matrix (Essential Matrix) or matrix K also unknown or has different focal lengths (Fundamental Matrix).





Terms, **e** and **e**′, are considered as **epipoles**, **epipolar plane** is defined by points **o**′, **O** and **X**. Besides, assume f and f' are the focal lengths of left and right cameras, respectively

Some homogeneous properties

- Point x on a line

$$\mathbf{l}^T\mathbf{x} = \mathbf{x}^T\mathbf{l} = 0, l_1x + l_2y + l_3 = 0$$

- Two points define a line

$$l = X_1 \times X_2$$

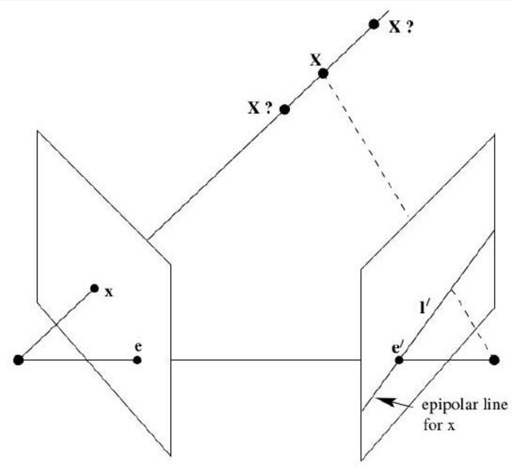
- Intersection of two lines defines a point

$$x = l_1 \times l_2$$

where cross product between two vectors can be written as a matrix multiplication

$$\mathbf{v} \times \mathbf{u} = [\mathbf{v}]_\times \mathbf{u}, \mathbf{v}_\times = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}$$

How can we see the location we see from left camera from the right camera



Since we have two cameras, two projection matrices with respect to left and right cameras have to be identified:

$$\mathbf{x} = \lambda_1 P_1 \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}$$

$$\mathbf{x}' = \lambda_2 P_2 \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}$$

where $P_1 = K_1(I \mid 0)$ and $P_2 = K_2(R \mid t)$, and baseline between the two cameras is denoted by t. Let's start assuming K_1 and K_2 are known. Then if

$$\hat{\mathbf{x}}' = K_2^{-1}\mathbf{x}' = \lambda_2(R \mid t) \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}$$

$$\hat{\mathbf{x}} = K_1^{-1}\mathbf{x} = \lambda_1(I \mid 0) \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix}$$

Now let's project **X** on the left and right images

$$\hat{\mathbf{x}} = \lambda_1(I \mid \mathbf{0}) \begin{pmatrix} \mathbf{X} \\ 1 \end{pmatrix} \Rightarrow \mathbf{X} = \lambda_1^{-1}\hat{\mathbf{x}}$$

$$\lambda_2(R \mid t) \begin{pmatrix} \lambda_1^{-1}\hat{\mathbf{x}} \\ 1 \end{pmatrix} = \lambda_2\lambda_1^{-1}R\hat{\mathbf{x}} + \lambda_2 t = \lambda_2 \left(\lambda_1^{-1}R\hat{\mathbf{x}} + t\right)$$

And this will be the epipolar line with respect to right camera in our setup. Let's take corresponding points when $\lambda_1 = 1$ and $\lambda_1 = \pm\infty$

$$(R\hat{\mathbf{x}} + t), t$$

Thus, we can define the right epipolar line:

$$\begin{aligned} \mathbf{l}' = t \times (R\hat{\mathbf{x}} + t) = t \times R\hat{\mathbf{x}} + t \times t = t \times R\hat{\mathbf{x}} \\ = [t]_\times R\hat{\mathbf{x}} = E\hat{\mathbf{x}} \end{aligned}$$

This matrix E is called the **Essential matrix**, which map point in the left image to a line in the right image. Thus, we can define the epipolar constraint that $\hat{\mathbf{x}}'$ lies on $\mathbf{l}'$ can be written as

$$\hat{\mathbf{x}}'^T \mathbf{l}' = \hat{\mathbf{x}}'^T E\hat{\mathbf{x}} = 0$$