Robotics - Single view, Epipolar geometry, Image Features

Simone Ceriani

ceriani@elet.polimi.it

Dipartimento di Elettronica e Informazione
Politecnico di Milano

12 April 2012

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Pin hole model - Recall

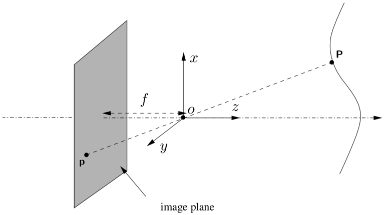
The intrinsic camera matrix

or *calibration matrix*

$$K = \begin{bmatrix} f_x & s & \mathbf{c}_x \\ 0 & f_y & \mathbf{c}_y \\ 0 & 0 & 1 \end{bmatrix}$$

- f_x, f_y: focal lenght (in pixels)
  $f_x/f_y = s_x/s_y = a$: aspect ratio
- s: skew factor
  pixel not orthogonal
  usually 0 in modern cameras
- $\mathbf{c}_x$, $\mathbf{c}_y$: principal point (in pixel)
  usually $\neq$ half image size due to misalignment of CCD

Projection



$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \mathbf{K} & \mathbf{0} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix}$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \frac{f_x X}{Z} + \mathbf{c}_x \\ \frac{f_y Y}{Z} + \mathbf{c}_y \end{bmatrix}$$

Points in the world

Consider

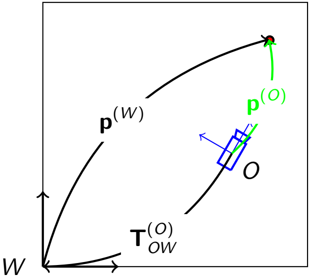
- $\mathbf{p}^{(\mathbf{I}')} = \begin{bmatrix} f_x & s & \mathbf{c}_x & 0 \\ 0 & f_y & \mathbf{c}_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{P}^{(O)}$
- $\mathbf{P}^{(O)} = \mathbf{T}_{OW}^{(O)} \mathbf{P}^{(W)}$

  extrinsic camera matrix

One step

- $\mathbf{p}^{(\mathbf{I}')} = \begin{bmatrix} \mathbf{K} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \mathbf{P}^{(W)}$
- $\pi = \begin{bmatrix} \mathbf{KR} & \mathbf{Kt} \end{bmatrix}$

  complete projection matrix

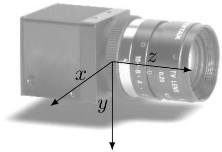


Note

- $\mathbf{R}$ is $\mathbf{R}_{OW}^{(O)}$
- $\mathbf{t}$ is $\mathbf{t}_{OW}^{(O)}$
- i.e., the position and orientation of W in O

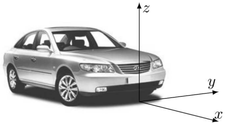
Note on camera reference system

Camera reference system



- z: front
- y: down

World reference system



- x: front
- z: up

Rotation of O w.r.t. W

- Rotate around y of 90°

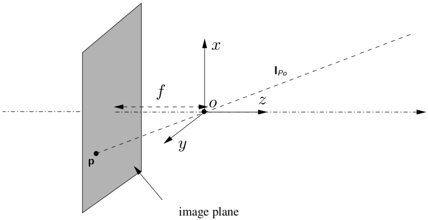
  z' front

- Rotate around z' of -90°

  y'' point down

- $\mathbf{R}_{WO}^{(W)} = \mathbf{R}_y(90^\circ)\,\mathbf{R}_z(-90^\circ)$
- $\mathbf{R}_{WO}^{(W)} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}$
- $\mathbf{R}_{OW}^{(O)} = {\mathbf{R}_{WO}^{(W)}}^T = \begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & -1 \\ 1 & 0 & 0 \end{bmatrix}$

Interpretation line



Given

- $\mathbf{p}^{(\mathbf{I})} = \left[u, v\right]^T$: point in image (pixel)

Calculate $\mathbf{P}^{(O)}$?

- No, only $\mathbf{l}_{Po}$: interpretation line
- $\forall \mathbf{P}_i^{(O)} \in \mathbf{l}_{Po}$ image is $\mathbf{p}^{(\mathbf{I})}$

Calculate $\mathbf{l}_{Po}$

- 3D lines not coded in 3D
  remember duality points $\leftrightarrow$ planes
- $\mathbf{p}^{(\mathbf{I}')} = \left[\mathbf{K} \quad \mathbf{0}\right] \mathbf{P}^{(O)}$
- $\mathbf{p}^{(\mathbf{I}')} = \left[\mathbf{K} \quad \mathbf{0}\right] \left[X, \ Y, \ Z, \ W\right]^T$
- $\mathbf{p}^{(\mathbf{I}')} = \mathbf{K} \left[X, \ Y, \ Z\right]^T$
  W "cancelled" by zeros fourth column
- $\mathbf{d}^{(\mathbf{O})} = \mathbf{K}^{-1} \left[u, v, 1\right]^T$
- $\overline{\mathbf{d}}^{(\mathbf{O})} = \mathbf{d}^{(\mathbf{O})} / \|\mathbf{d}^{(\mathbf{O})}\|$: unit vector
- $\mathbf{P}_i^{(\mathbf{O})} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \lambda \begin{bmatrix} \overline{\mathbf{d}}^{(\mathbf{O})} \\ 0 \end{bmatrix}, \lambda > 0$

$\mathbf{l}_{Po}$ in parametric form

Interpretation line & Normalized image plane

Interpretation line direction

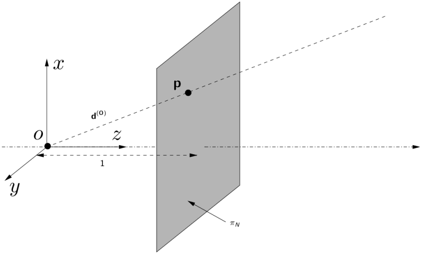
- $\mathbf{d}^{(O)} = \mathbf{K}^{-1}\left[u, v, 1\right]^T$
- $\mathbf{K}^{-1} = \begin{bmatrix} 1/f_x & 0 & -\mathbf{c}_x/fx & 0 \\ 0 & 1/f_y & -\mathbf{c}_y/fy & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

  assume skew $= 0$
- $\mathbf{d}^{(O)} = \left[\frac{u-\mathbf{c}_x}{fx}, \frac{v-\mathbf{c}_y}{fy}, 1\right]^T$
- $\mathbf{P}^{(O)}_{\lambda=\|\mathbf{d}^{(O)}\|} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{d}^{(O)} \\ 0 \end{bmatrix}$

  lies on π_N
- If $u = \mathbf{c}_x$, $v = \mathbf{c}_y$, $\mathbf{d}$ is

  the *principal direction*

Normalized image plane



- Distance 1 from the optical center
- Independent of camera intrinsic

Given a cartesian point $\mathbf{P}^{(O)} = \left[X, Y, Z\right]^T$

$$\mathbf{P}^{(O)}_{\pi_N} = \left[X/Z, Y/Z, 1\right]^T$$

Interpretation line in the world - 1

Consider

- Interpretation line in camera coordinate

$$\mathbf{P}_i^{(\mathbf{O})} = \begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \lambda \begin{bmatrix} \overline{\mathbf{d}}^{(\mathbf{O})} \\ 0 \end{bmatrix}$$

- Interpretation line in world coordinate

$$\begin{aligned} \mathbf{P}_i^{(\mathbf{W})} &= \begin{bmatrix} \mathbf{R}_{OW}^{(O)\,T} & -\mathbf{R}_{OW}^{(O)\,T}\mathbf{t}_{OW}^{(O)} \\ \mathbf{0} & 1 \end{bmatrix} \left(\begin{bmatrix} \mathbf{0} \\ 1 \end{bmatrix} + \lambda \begin{bmatrix} \overline{\mathbf{d}}^{(\mathbf{O})} \\ 0 \end{bmatrix} \right) \\ &= \begin{bmatrix} -\mathbf{R}_{OW}^{(O)\,T}\mathbf{t}_{OW}^{(O)} \\ 1 \end{bmatrix} + \begin{bmatrix} \lambda\,\mathbf{R}_{OW}^{(O)\,T}\overline{\mathbf{d}}^{(\mathbf{O})} \\ 0 \end{bmatrix} \\ &= \mathbf{O}^{(W)} + \lambda\,\overline{\mathbf{d}}^{(\mathbf{W})} \end{aligned}$$

- Camera center in world coordinate + direction rotated as world reference

Interpretation line in the world - 2

Consider

- Interpretation line in world coordinate

$$\mathbf{P}_i^{(\mathbf{W})} = \lambda\, \mathbf{R}_{OW}^{(O)\,T}\, \overline{\mathbf{d}}^{(\mathbf{O})} - \mathbf{R}_{OW}^{(O)\,T}\, \mathbf{t}_{OW}^{(O)}$$

$$= \lambda\, \mathbf{R}_{OW}^{(O)\,T}\, \mathbf{K}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} - \mathbf{R}_{OW}^{(O)\,T}\, \mathbf{t}_{OW}^{(O)}$$

- Complete projection matrix

$$\boldsymbol{\pi} = \begin{bmatrix} \mathbf{K}\mathbf{R}_{OW}^{(O)} & \mathbf{K}\mathbf{t}_{OW}^{(O)} \end{bmatrix} = \begin{bmatrix} \mathbf{M} & \mathbf{m} \end{bmatrix}$$

- $\mathbf{M}^{-1} = \mathbf{R}_{OW}^{(O)\,T}\mathbf{K}^{-1}$: direction in world coordinate given pixel coordinate
- $-\mathbf{M}^{-1}\mathbf{m} = -\mathbf{R}_{OW}^{(O)\,T}\, \mathbf{K}^{-1}\, \mathbf{K}\, \mathbf{t}_{OW}^{(O)} = -\mathbf{R}_{OW}^{(O)\,T}\, \mathbf{t}_{OW}^{(O)} = \mathbf{t}_{WO}^{(W)}$:

  camera center in world coordinate $\mathbf{O}^{(W)}$

Principal ray

Interpretation line of principal point

- $\mathbf{d}^{(\mathbf{O})} = \mathbf{K}^{-1} \begin{bmatrix} \mathbf{c}_x \\ \mathbf{c}_y \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

  z-axis of the camera reference system

- $\mathbf{d}^{(\mathbf{W})} = \mathbf{R}_{OW}^{(O)\,T} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} \mathbf{c}_x \\ \mathbf{c}_y \\ 1 \end{bmatrix}$

  z-axis of the camera in world reference system

- $\mathbf{P}_i^{(\mathbf{W})} = \lambda\, \mathbf{M}^{-1} \begin{bmatrix} \mathbf{c}_x \\ \mathbf{c}_y \\ 1 \end{bmatrix} - \mathbf{M}^{-1}\mathbf{m}$

  parametric line of z-axis of the camera in world reference system

Vanishing points & Origin

Vanishing points

- $\mathbf{V}_x^{(\mathbf{W})} = \begin{bmatrix}1, 0, 0, 0\end{bmatrix}^T$
- $\mathbf{V}_y^{(\mathbf{W})} = \begin{bmatrix}0, 1, 0, 0\end{bmatrix}^T$
- $\mathbf{V}_z^{(\mathbf{W})} = \begin{bmatrix}0, 0, 1, 0\end{bmatrix}^T$

Projection on the image

- $\mathbf{p}^{(\mathbf{I}')} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}\begin{bmatrix}\mathbf{d} \\ 0\end{bmatrix} = \mathbf{Md}$
- $\mathbf{p}_x^{(\mathbf{I}')} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}\mathbf{V}_x^{(\mathbf{W})} = \mathbf{M}^{(1)}$
- $\mathbf{p}_x^{(\mathbf{I}')} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}\mathbf{V}_y^{(\mathbf{W})} = \mathbf{M}^{(2)}$
- $\mathbf{p}_z^{(\mathbf{I}')} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}\mathbf{V}_z^{(\mathbf{W})} = \mathbf{M}^{(3)}$

Origin

- $\mathbf{O}^{(\mathbf{W})} = \begin{bmatrix}0, 0, 0, 1\end{bmatrix}^T$

Projection on the image

- $\mathbf{p}^{(\mathbf{I}')} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}\begin{bmatrix}\mathbf{0} \\ 1\end{bmatrix} = \mathbf{m}$

Note

- Col 1 of $\boldsymbol{\pi}$ is image of x vanishing point
- Col 2 of $\boldsymbol{\pi}$ is image of y vanishing point
- Col 3 of $\boldsymbol{\pi}$ is image of w vanishing point
- Col 4 of $\boldsymbol{\pi}$ is image of $\mathbf{O}^{(\mathbf{W})}$

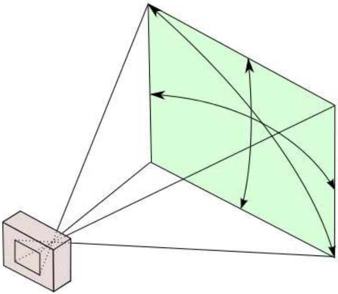
Angle of View

Given

- Image size: $[w, h]$
- Focal lenght: f_x (assume $f_x = f_y$)

Angle of view

- $\theta = 2\text{atan2}(w/2, f_x)$
- $\theta < 180^\circ$



Examples

14mm

20mm

28mm

35mm

50mm

From past exam

Ex. 4 - 20 November 2006

Problem

- Given $\boldsymbol{\pi} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 3 & 1 & 0 & 4 \\ 1 & 2 & 3 & 1 \end{bmatrix}$
- Where is the camera center in world reference frame?

Solution

- $\boldsymbol{\pi} = \begin{bmatrix} \mathbf{M} & \mathbf{m} \end{bmatrix}$
- $\mathbf{O}^{(W)} = -\mathbf{M}^{-1}\mathbf{m}$
- $\mathbf{M}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 1 & 0 \\ 5/3 & -2/3 & 1/3 \end{bmatrix}$
- $\mathbf{O}^{(W)} = \begin{bmatrix} -1 \\ -1 \\ 2/3 \end{bmatrix}$

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

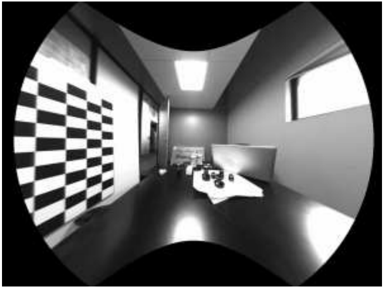
Distortion

Distortion

- Deviation from rectilinear projection
- Lines in scene don't remains lines in image

Original image

Corrected



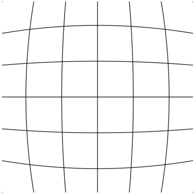
Distortion

Distortion

- Can be irregular
- Most common is *radial* (radially symmetric)

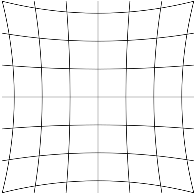
Radial Distortion

Barrel distortion



Magnification decrease with distance from optical axis

Pincushion distortion



Magnification increase with distance from optical axis

Brown distortion model

Consider

- $\mathbf{P}^{(O)} = \begin{bmatrix} X, & Y, & Z \end{bmatrix}^T$ in camera reference system
- Calculate $\mathbf{p}^{(\mathbf{I})} = \begin{bmatrix} x, & y, & 1 \end{bmatrix}^T = \begin{bmatrix} X/Z, & Y/Z, & 1 \end{bmatrix}^T$ on the normalized image plane

Distortion model

- $\tilde{\mathbf{p}}^{(\mathbf{I})} = \left(1 + \mathbf{k}_1 r^2 + \mathbf{k}_2 r^4 + \mathbf{k}_3 r^6\right) \mathbf{p}^{(\mathbf{I})} + d_x$
- $r^2 = x^2 + y^2$: distance wrt optical axis $(0, 0)$
- $d_x = \begin{bmatrix} 2p_1 xy + p_2(r^2 + 2x^2) \\ p_1(r^2 + 2y^2) + 2p_2 xy \end{bmatrix}$: tangential distortion compensation

Image coordinate

- $\tilde{\mathbf{p}}^{(\mathbf{I}')} = \mathbf{K}\tilde{\mathbf{p}}^{(\mathbf{I})}$: pixel coordinate of $\mathbf{P}^{(O)}$ considering distortion

Brown distortion model

From image points

- $\tilde{\mathbf{p}}^{(\mathbf{I}')}$ in image (pixel)
- Calculate $\tilde{\mathbf{p}}^{(\mathbf{I})} = \mathbf{K}^{-1}\tilde{\mathbf{p}}^{(\mathbf{I}')} = \left[x,\, y,\, 1\right]^T$ on the (distorted) normalized image plane
- Undistort: $\mathbf{p}^{(\mathbf{I})} = dist^{-1}(\tilde{\mathbf{p}}^{(\mathbf{I})})$
- Image projection: $\mathbf{p}^{(\mathbf{I}')} = \mathbf{K}\mathbf{p}^{(\mathbf{I})}$

Evaluation of $dist^{-1}(\cdot)$

- No analytic solution
- Iterative solution ($N = 20$ is enough):

$\mathbf{p}^{(\mathbf{I})} = \tilde{\mathbf{p}}^{(\mathbf{I})}$: initial guess

for $i = 1$ to N **do**

$$r^2 = x^2 + y^2,\ k_r = \left(1 + \mathbf{k}_1 r^2 + \mathbf{k}_2 r^4 + \mathbf{k}_3 r^6\right),\ d_x = \begin{bmatrix} 2p_1xy + p_2(r^2 + 2x^2) \\ p_1(r^2 + 2y^2) + 2p_2xy \end{bmatrix}$$

$$\mathbf{p}^{(\mathbf{I})} = \left(\tilde{\mathbf{p}}^{(\mathbf{I})} - d_x\right)/k_r$$

end for

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Camera calibration

Intrinsic calibration

- Find parameters of **K**
- Nominal values of optics are not suitable
- Differences between different exemplar of same camera/optic system
- Include distortion coefficient estimation

Extrinsic calibration

- Find parameters of $\boldsymbol{\pi} = \begin{bmatrix} \mathbf{M} & \mathbf{m} \end{bmatrix}$
- i.e., find **K** and **R**, **t**

Camera calibration - Approaches

Calibration

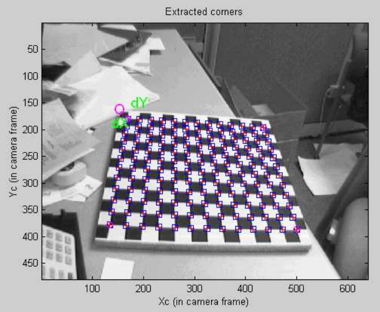
- Very large literature!
- Different approaches

Known 3D pattern

Methods

- Based on correspondances
- Need for a pattern

Planar pattern



Camera calibration - Formulation

Formulation

- $\mathbf{M}_i$: model points on the pattern
- $\mathbf{p}_{ij}$: observation of model point i in image j
- $\mathbf{p} = \left[f_x, f_y, s, \cdots k_1, k_2, \cdots \right]^T$: intrinsic parameters
- $\mathbf{R}_j, \mathbf{t}_j$: pose of the patter wrt camera reference frame j

  i.e., $\mathbf{R}^{(C)}_{CP\,j}, \mathbf{t}^{(C)}_{CP\,j}$
- $\hat{\mathbf{m}}(\mathbf{p}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{M}_i)$: estimated projection of $\mathbf{M}_i$ in image j.

Estimation

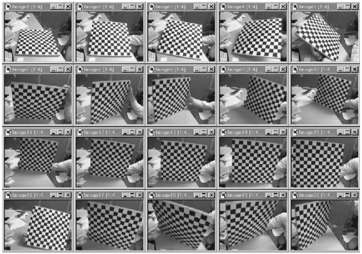
- $\underset{\mathbf{p}, \mathbf{R}_j, \mathbf{t}_j}{\operatorname{argmin}} \sum_j \sum_i \mathbf{p}_{ij} - \mathbf{m}_{ij}(\mathbf{p}, \mathbf{R}_j, \mathbf{t}_j, \mathbf{M}_i)$: observation of model point i in image j
- Gives both *intrinsic* (unique) and *extrinsic* (one for each image) calibration

Z. Zhang, "A flexible new technique for camera calibration", 2000

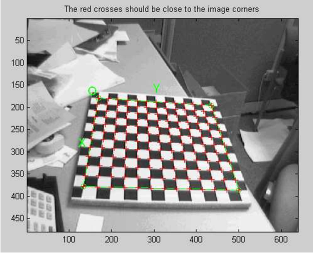
Heikkila, Silvén, "A Four-step Camera Calibration Procedure with Implicit Image Correction", 1997

Camera Calibration Toolbox for Matlab

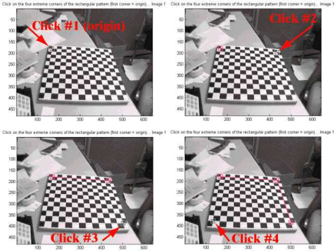
Collect images



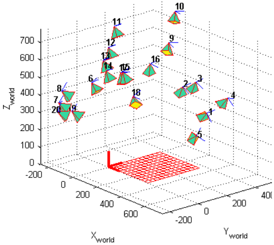
Automatic corners identification



Find chessboard external corners



Calibration



Camera Calibration Toolbox for Matlab - *http://www.vision.caltech.edu/bouguetj/calib_doc/*

Outline

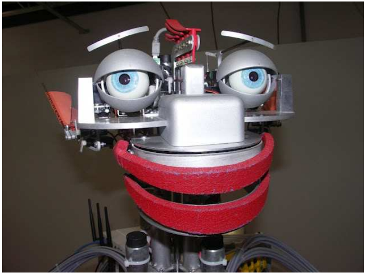
1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. **Two views geometry**
5. Image features
6. Edge, corners
7. Exercise

Epipolar geometry introduction

Epipolar geometry

- Projective geometry between two views
- Independent of scene structure
- Depends only on
  - Cameras parameters
  - Cameras relative position
- Two views
  - Simultaneously (stereo)
  - Sequentially (moving camera)

Bumblebee camera



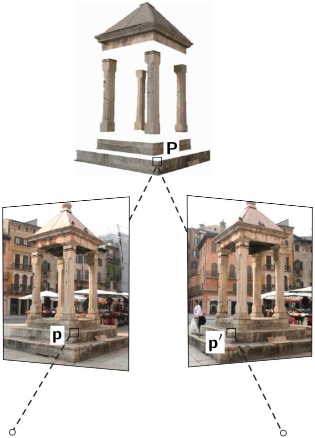
Robot head with two cameras

Correspondence

Consider

- $\mathbf{P}$ a 3D point in the scene
- Two cameras with $\boldsymbol{\pi}$ and $\boldsymbol{\pi}'$
- $\mathbf{p} = \boldsymbol{\pi}\mathbf{P}$ image on first camera
- $\mathbf{p}' = \boldsymbol{\pi}'\mathbf{P}$ image on second camera
- $\mathbf{p}$ and $\mathbf{p}'$: images of the same point

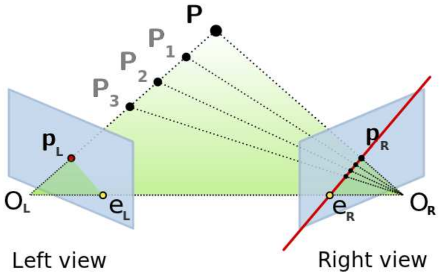
  $\rightarrow$ *correspondence*



Correspondence geometry

- $\mathbf{p}$ on first image
- How $\mathbf{p}'$ is constrained by $\mathbf{p}$?

Correspondences and epipolar geometry



Suppose (1)

- $\mathbf{P}$, a 3D point imaged in two views
- $\mathbf{p}_L$ and $\mathbf{p}_R$ image of $\mathbf{P}$
- $\mathbf{P}$, $\mathbf{p}_L$, $\mathbf{p}_R$, O_L, O_R are coplanar on π
- π is the *epipolar plane*

Suppose (2)

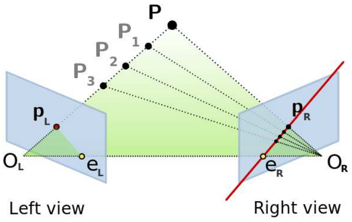
- $\mathbf{P}$ is unknown
- $\mathbf{p}_L$ is known
- Where is $\mathbf{p}_R$?

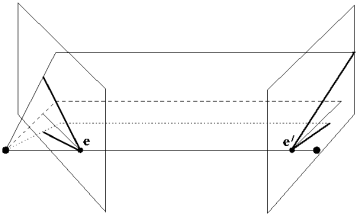
  or how is constrained $\mathbf{p}_R$?
- $\mathbf{P}_i = O_L + \lambda \mathbf{d}_{\mathbf{p}_L O_L}$ is the interpretation line of $\mathbf{p}_L$
- $\mathbf{p}_R$ lies on a line:

  intersection of π with the 2^{nd} image

  $\rightarrow$ *epipolar line*

Epipolar geometry - Definitions





Base line

- Line joining O_L and O_R

Epipoles ($\mathbf{e}_L$, $\mathbf{e}_R$)

- Intersection of base line with image planes
- Projection of camera centres on images
- Intersection of all epipolar lines

Epipolar line

- Intersection of epipolar plane with image plane

Epipolar plane

- A plane containing the baseline
- It's a pencil of planes
- Given an *epipolar line* is possible to identify a unique epipolar plane

Epipolar constraints

Correspondences problem

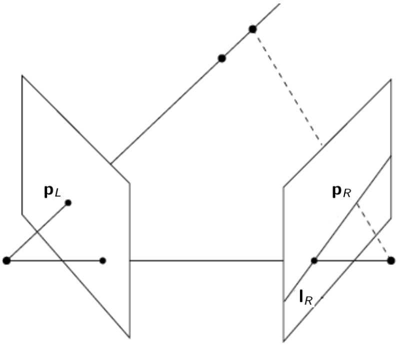
- Given $\mathbf{p}_L$ in one image
- Search on second image along the *epipolar line*
- 1D search!
- A point in one image "generates" a line in the second image

Correspondences example

The fundamental matrix $\mathbf{F}$

Epipolar geometry

- Given $\mathbf{p}_L$ on one image
- $\mathbf{p}_R$ lies on $\mathbf{l}_R$, i.e. the *epipolar line*
- $\mathbf{p}_R \in \mathbf{l}_R \leftrightarrow \mathbf{p}_R^T \mathbf{l}_R = 0$
- Thus, there is a map $\mathbf{p}_L \to \mathbf{l}_R$



The fundamental matrix $\mathbf{F}$

- $\mathbf{l}_R = \mathbf{F}\,\mathbf{p}_L$
- $\mathbf{F}$ is the fundamental matrix
- $\mathbf{p}_R \in \mathbf{l}_R \leftrightarrow \mathbf{p}_R^T \mathbf{F}\,\mathbf{p}_L = 0$

The fundamental matrix $\mathbf{F}$ properties

Properties

- If $\mathbf{p}_L$ correspond to $\mathbf{p}_R$ $\rightarrow \mathbf{p}_L \mathbf{F} \mathbf{p}_R = 0$, necessary condition for correspondence
- If $\mathbf{p}_L \mathbf{F} \mathbf{p}_R = 0$ interpretation lines (a.k.a. *viewing ray*) are coplanar
- $\mathbf{F}$ is a 3x3 matrix
- $\det(\mathbf{F}) = 0$
- $\text{rank}(\mathbf{F}) = 2$
- $\mathbf{F}$ has 7 dof (1 homogeneous, 1 rank deficient)
- $\mathbf{l}_R = \mathbf{F}\,\mathbf{p}_L$, $\mathbf{l}_L = \mathbf{F}^T\,\mathbf{p}_R$
- $\mathbf{F}\mathbf{e}_L = 0$, $\mathbf{F}^T\mathbf{e}_R = 0$, i.e.: epipoles are the right null vector of $\mathbf{F}$ and $\mathbf{F}^T$

  Proof: $\forall \mathbf{p}_L \neq \mathbf{e}_L, \quad \mathbf{l}_R = \mathbf{F}\,\mathbf{p}_L$ and $\mathbf{e}_R \in \mathbf{l}_R \rightarrow \forall \mathbf{p}_L \quad \mathbf{e}_R^T \mathbf{F}\,\mathbf{p}_L = 0 \rightarrow \mathbf{F}^T\mathbf{e}_R = 0$

F calculus

From calibrated cameras

- $\boldsymbol{\pi}_L$ and $\boldsymbol{\pi}_R$ are known
- $\mathbf{F} = [\mathbf{e}_R]_\times \boldsymbol{\pi}_R \boldsymbol{\pi}_L^+$

  where

  - $\mathbf{e}_R = \boldsymbol{\pi}_R \mathbf{O}_L = \boldsymbol{\pi}_R \left(-\mathbf{M}_L^{-1}\mathbf{m}_L\right)$
  - $\boldsymbol{\pi}_L^+ = \boldsymbol{\pi}_L^T \left(\boldsymbol{\pi}_L \boldsymbol{\pi}_L^T\right)^{-1}$: pseudo-inverse
  - $\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_\times \mathbf{b}$ where $[\mathbf{a}]_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$

Calibrated cameras with $\mathbf{K}$ and $\mathbf{R}, \mathbf{t}$

- $\boldsymbol{\pi}_L = \mathbf{K}_L \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}$: origin in the left camera
- $\boldsymbol{\pi}_R = \mathbf{K}_R \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}$
- $\mathbf{F} = \mathbf{K}_R^{-T} [\mathbf{t}]_\times \mathbf{R}\, \mathbf{K}_L^{-1}$
- Special forms with pure translations, pure rotations, ...

F estimation - procedure sketch

From uncalibrated images

- Get point correspondances ("by hand" or automatically)
- Compute **F** by consider that
  - $\mathbf{p}_R^T \mathbf{F} \mathbf{p}_L = 0$
  - At least 7 correspondances are needed

    but the 8-point algorithm is the simplest
  - Impose rank($\mathbf{F}$) = 2

Details

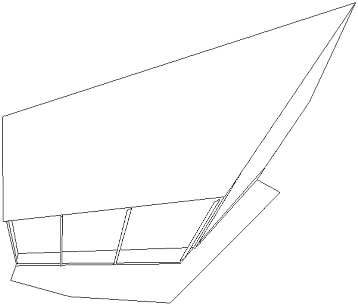
"Multiple View Geometry in computer vision"

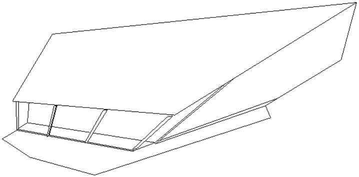
Hartley Zisserman. Chapters 9,10,11,12.

Projective reconstruction

F is not unique

- If estimated by correspondences
- Without any additional constraints allow at least a projective reconstruction

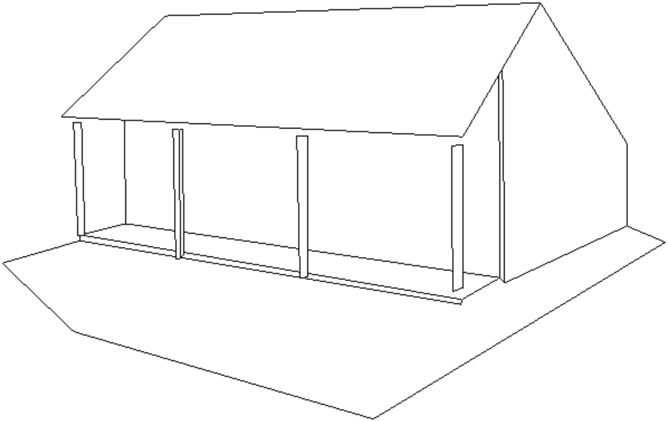




Metric reconstruction and reconstruction

Additional constraints

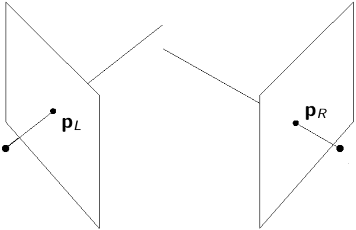
- Parallelism, measures of some points, ...
- $\rightarrow$ allow affine/similar/metric reconstruction



Triangulation

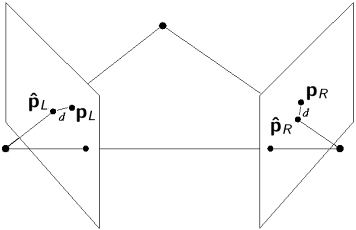
Suppose

- $\mathbf{p}_R$ and $\mathbf{p}_L$ are correspondent points
- π_L and π_R are known
- Due to noise is possible that interpretation lines don't intersect
- $\mathbf{p}_L \mathbf{F} \mathbf{p}_R \neq 0$



3D point computation

- $\underset{\hat{\mathbf{p}}_L, \hat{\mathbf{p}}_R}{\operatorname{argmin}} \quad d(\hat{\mathbf{p}}_L, \mathbf{p}_L)^2 + d(\hat{\mathbf{p}}_R, \mathbf{p}_R)^2$
- subject to $\hat{\mathbf{p}}_L \mathbf{F} \hat{\mathbf{p}}_R = 0$



Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Features in image

What is a feature?

- No common definition
- Depends on problem or application
- Is an *interesting part* of the image

Types of features

- Edges
  - Boundary between regions
- Corners / interest points
  - Edge intersection
  - Corners
  - Point-like features
- Blobs
  - Smooth areas that define regions

Edges

Corners

Blob

Black & White threshold

Thresholding

- On a gray scale image $I(u, v)$
- If $I(u, v) > T \quad I'(u, v) = white$
- else $I'(u, v) = black$

Properties

- Simplest method of image segmentation
- Critical point: threshold T value
  - Mean value of $I(u, v)$
  - Median value of $I(u, v)$
  - (Local) adaptive thresholding

Filtering

Kernel matrix filtering

- Given an image $I(i,j), i = 1 \cdots h, j = 1 \cdots w$
- A kernel $H(k,z), k = 1 \cdots r, z = 1 \cdots c$
- $I'(i,j) = \sum_k \sum_z I(i - \lfloor r/2 \rfloor + k - 1, j - \lfloor c/2 \rfloor + z - 1) * H(k,z)$
- special cases on borders

$I(i,j)$

2	2	2	3
2	1	3	3
2	2	1	2
1	3	2	2

$H(k,z)$

1	1	1
-1	2	1
-1	-1	1

2	2	2	3
-2	2	3	3
-2	-2	1	2
1	3	2	2

$I'(2,2)$

5	4	4	-2
9	6		

Final result

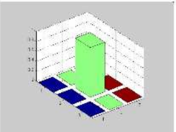
5	4	4	-2
9	6	14	5
11	7	6	5
9	12	8	5

Filter Examples - 1

IDENTITY

$*$

0	0	0
0	1	0
0	0	0

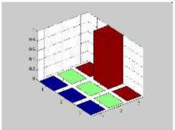


$=$

TRANSLATION

$*$

0	0	0
0	0	1
0	0	0

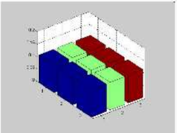


$=$

Filter Examples - 2

AVERAGE (3×3)

$$* \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$



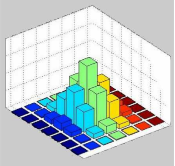
AVERAGE (5×5)

$$* \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} =$$

Filter Examples - 3

Gaussian - $\sim N(0, \sigma)$

$*$



$=$

Gaussian vs Average

Smoothing

Generally expect

- Pixels to "be like" neighbours
- Noise independent from pixel to pixel

Implies

- Smoothing suppress noises
- Appropriate noise model (?)

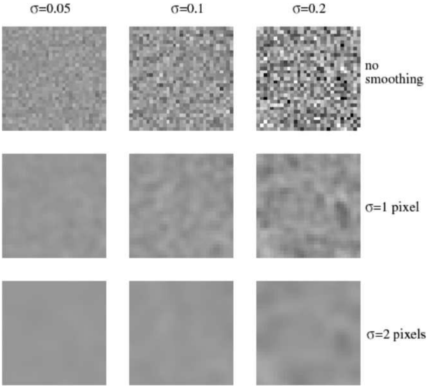


Image Gradient - 1

HORIZONTAL DERIVATIVES (∇I_x)

$*$

1	0	-1
1	0	-1
1	0	-1

$=$

Image Gradient - 2

Vertical derivatives (∇I_y)

$*$

1	1	1
0	0	0
-1	-1	-1

$=$

Rough edge detector

$$\underline{\nabla I_x^2 + \nabla I_y^2}$$

^{2} $+$

^{2} $=$

then apply threshold...

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Canny Edge Detector

Criterion

- *Good detection*: minimize probability of false positive and false negatives
- *Good localization*: Edges as closest as possible to the true edge
- *Single response*: Only one point for each edge point (thick = 1)

Procedure

- Smooth by Gaussian ($S = I * G(\sigma)$)
- Compute derivatives (∇S_x, ∇S_y)

  Alternative in one step: Filter with derivative of Gaussian
- Compute magnitude and orientation of gradient

  $(\|\nabla S_x\| = \sqrt{\nabla S_x^2 + \nabla S_y^2},\ \theta_{\nabla S} = \operatorname{atan2} \nabla S_y, \nabla S_x)$
- Non maxima suppression

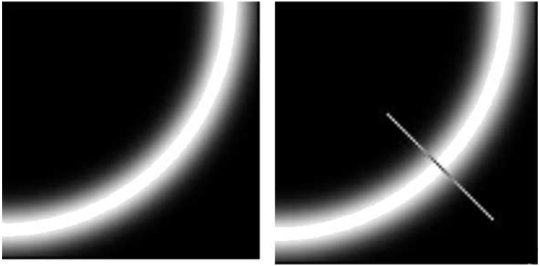
  Search for local maximum in the gradient direction $\theta_{\nabla S}$
- Hysteresis threshold

  Weak edges (between the two thresholds) are edges

  if connected to strong edges (greater than high threshold)

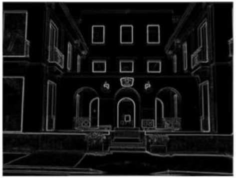
Canny Edge Detector - Non Maxima Suppression

Non Maxima Suppression



Example

Original image



Gradient magnitude

Non-maxima suppressed

Canny Edge Detector - Hysteresis threshold

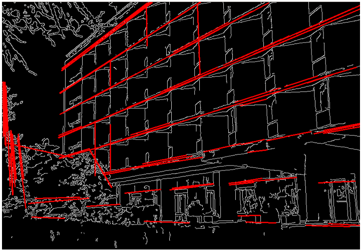
Hysteresis threshold example

Lines from edges

How to find lines?

- *Hough Transformation* after Canny edges extraction
- Use a *voting procedure*
- Generally find imperfect instances of a shape class
- Classical Hough Transform for lines detection
- Later extended to other shapes (most common: circles, ellipses)

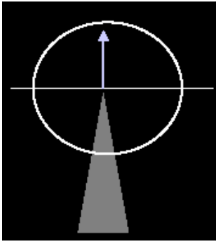
Example



Corners - Harris and Shi Tomasi

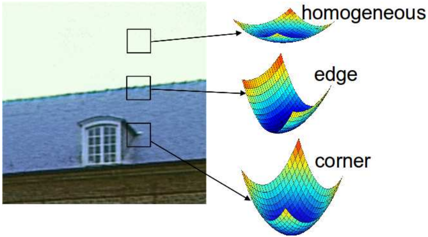
Edge intersection

- At intersection point gradient is ill defined
- Around intersection point gradient changes in "all" directions
- It is a "good feature to track"



Corner detector

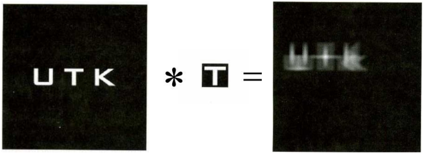
- Examine gradient over window
- $C = \sum_w \sum \begin{bmatrix} \nabla I_x^2 & \nabla I_x \nabla I_y \\ \nabla I_x \nabla I_y & \nabla I_y^2 \end{bmatrix}$
- Shi-Tomasi: corner if $\min \text{eigenvalue}(C) > T$
- Harris: approximation of eigenvalues



Template matching - Patch

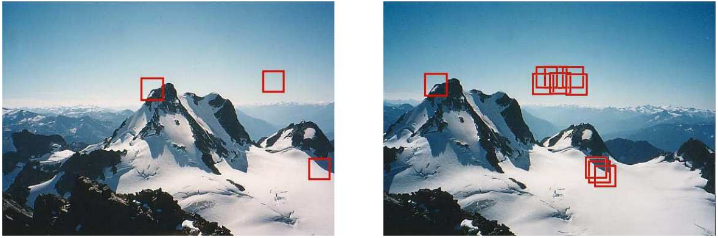
Filtering with a template

- Correlation between template (patch) and image
- Maximum where template matches
- Alternatives with normalizations for illumination compensation, etc.



Good features

- On corners: higher repeatability (homogeneous zone and edges are not distinctive)



Template matching - SIFT

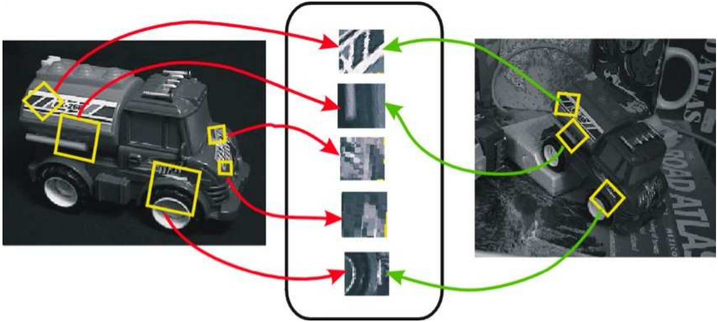
Template matching issues

- Rotations
- Scale change

SIFT

- Scale Invariant Feature Transform
- Alternatives descriptor to patch
- Performs orientation and scale normalization
- *See also SURF (Speeded Up Robust Feature)*

SIFT example



D. Lowe - "Object recognition from local scale-invariant features" - 1999

Outline

1. Pin Hole Model
2. Distortion
3. Camera Calibration
4. Two views geometry
5. Image features
6. Edge, corners
7. Exercise

Camera matrix - 1

Given

- $$\mathbf{P} = \begin{bmatrix} 122.5671 & -320.0000 & -102.8460 & 587.3835 \\ -113.7667 & 0.0000 & -322.2687 & 350.6050 \\ 0.7660 & 0 & -0.6428 & 4.6711 \end{bmatrix}$$
- $f_x = f_y = 320$
- $c_x = 160$
- $c_y = 120$

Questions

- Where is the camera in the world?
- Compute the coordinate of the vanishing point of x, y plane in the image
- Where is the origin of the world in the image?
- Write the parametric 3D line of the principal axis in world coordinates
- ...

Camera matrix - 2

Where is the camera in the world?

- $\mathbf{P} = \begin{bmatrix} \mathbf{KR} & \mathbf{Kt} \end{bmatrix}$
- $\mathbf{K} = \begin{bmatrix} 320 & 0 & 1600 & 320 & 1200 & 0 & 1 \end{bmatrix}$
- $\mathbf{R} = \mathbf{K}^{-1}\mathbf{P}(1:3, 1:3)$
- $\mathbf{t} = \mathbf{K}^{-1}\mathbf{P}(1:3, 4)$
- $\mathbf{T}_{WC}^{(W)} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix}^{-1}$

  P contains the world wrt camera
- Camera is at $[-4, -0.5, 2.5]$
- Rotation around axis x, y, z is $[-130°, 0.0°, -90°]$
- To be more clear, remove the rotation of camera reference frame
- $\mathbf{T}_{WC}^{(W)}\mathbf{R}_z(90°), \mathbf{R}_y(-90°)$

  rotation around axis x, y, z is $[0°, 40.0°, 0°]$

Camera matrix - 3

Vanishing point of x, y plane in the image

- $\mathbf{v}_x = \mathbf{P}\begin{bmatrix}1, 0, 0, 0\end{bmatrix}^T \equiv \begin{bmatrix}160.0, & -148.5, 1\end{bmatrix}^T$
- $\mathbf{v}_y = \mathbf{P}\begin{bmatrix}0, 1, 0, 0\end{bmatrix}^T \equiv \begin{bmatrix}1, & 0, 0\end{bmatrix}^T$ (improper point)
- Remember: they are the 1^{st} and 2^{nd} column of $\mathbf{P}$

Where is the origin of the world in the image?

- $\mathbf{o} = \mathbf{P}\begin{bmatrix}0, 0, 0, 1\end{bmatrix}^T \equiv \begin{bmatrix}125.75, & 75.05, 1\end{bmatrix}^T$
- Remember: it is the 4^{st} column of $\mathbf{P}$

Principal axis in world coordinates

- $\mathbf{P} = \begin{bmatrix}\mathbf{M} & \mathbf{m}\end{bmatrix}$
- $\mathbf{O}^{(W)} = -\mathbf{M}^{-1}\mathbf{m} = \begin{bmatrix}-4, -0.5, 2.5\end{bmatrix}^T$
- $\mathbf{d}^{(W)} = \mathbf{M}^{-1}\begin{bmatrix}c_x, c_y, 1\end{bmatrix}^T = \begin{bmatrix}0.766, 0, -0.6428\end{bmatrix}^T$
- $\mathbf{a}^{(W)} = \mathbf{O}^{(W)} + \lambda\mathbf{d}^{(W)}$

Camera matrix - 5

Questions

- Where is the intersection between principal axis and the floor?
- Calculate the field of view of the camera (image size is 320×240)

  i.e., the portion of the plane imaged by the camera

Camera matrix - 6

Intersection between principal axis and the floor

- $\mathbf{a}^{(W)} = \mathbf{O}^{(W)} + \lambda \mathbf{d}^{(W)}$
- $\mathbf{a}_{z=0}^{(W)} = \left[X, Y, 0\right]^T$
- $\lambda_{z=0} = -\mathbf{O}_z^{(W)} / \mathbf{d}_z^{(W)} = 3.89$
- $\mathbf{a}_{z=0}^{(W)} = \left[-1.02, -0.5, 0\right]^T$

Field of view (1)

- $\mathbf{a}_1^{(W)} = \mathbf{O}^{(W)} + \lambda_1 \mathbf{M}^{-1} \left[0, 0, 1\right]^T$
- $\mathbf{a}_2^{(W)} = \mathbf{O}^{(W)} + \lambda_2 \mathbf{M}^{-1} \left[320, 0, 1\right]^T$
- $\mathbf{a}_3^{(W)} = \mathbf{O}^{(W)} + \lambda_3 \mathbf{M}^{-1} \left[0, 240, 1\right]^T$
- $\mathbf{a}_4^{(W)} = \mathbf{O}^{(W)} + \lambda_4 \mathbf{M}^{-1} \left[320, 240, 1\right]^T$
- calculate λ_i such that $\mathbf{a}_i^{(W)}$ has $z = 0$
- ...

Camera matrix - 7

Field of view (2)

- Transformation between plane $z = 0$ and image plane is a 2D homography
- Consider $\mathbf{p}_{z=0} = \left[x, y, 0, 1\right]^T$;
- Projection $\mathbf{P}\mathbf{p}_{z=0}$
- Notice that $\mathbf{p}_{z=0} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{W}} \left[x, y, 1\right]^T$
- $\mathbf{H} = \mathbf{P}\mathbf{W}$ is the homography

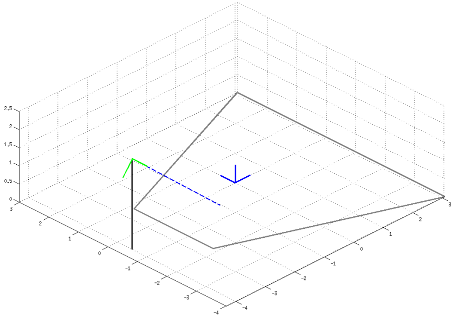
  maps points (x, y) on the $z = 0$ plane to the image
- $\mathbf{H}' = \mathbf{H}^{-1}$ is the inverse homography

  maps image points (u, v) to the $z = 0$ plane

Camera matrix - 8

Field of view (2) (continue)

- $\mathbf{H} = \begin{bmatrix} 122.56 & -320.00 & 587.38 \\ -113.76 & 0.00 & 350.60 \\ 0.76 & 0 & 4.67 \end{bmatrix}$
- $\mathbf{p}_{1\,z=0} = \mathbf{H}' \left[0, 0, 1\right]^T$
- $\mathbf{p}_{2\,z=0} = \mathbf{H}' \left[320, 0, 1\right]^T$
- $\mathbf{p}_{3\,z=0} = \mathbf{H}' \left[0, 240, 1\right]^T$
- $\mathbf{p}_{4\,z=0} = \mathbf{H}' \left[320, 240, 1\right]^T$



The 3D world with the camera reference system (green), the world reference system (blue), the principal axis (dashed blue) and the Field of view (FoV) (grey)

Camera matrix - 9

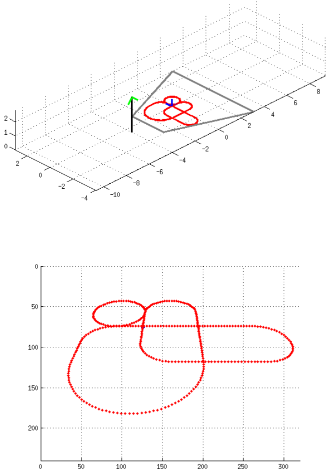
Questions

- There is a "flat robot" moving on the floor, imaged by the camera
- Two distinct and coloured point are drawn on the robot.
  $\mathbf{p}_1^{(R)} = \left[-.3, 0\right]^T$, $\mathbf{p}_2^{(R)} = \left[.3, 0\right]^T$
- Could you calculate the robot position and orientation?

Camera matrix - 10

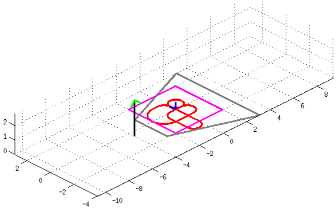
Robot pose

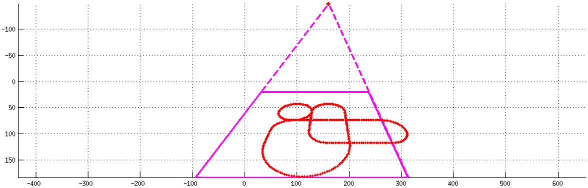
- Call $\mathbf{p}'_1$, $\mathbf{p}'_2$ the points in the image
- $\mathbf{p}_{1_{z=0}} = \mathbf{H}'\mathbf{p}'_1$
- $\mathbf{p}_{2_{z=0}} = \mathbf{H}'\mathbf{p}'_2$
- Position: $\frac{1}{2}(\mathbf{p}_{1_{z=0}} + \mathbf{p}_{2_{z=0}})$
- $\mathbf{d} = \mathbf{p}_{2_{z=0}} - \mathbf{p}_{1_{z=0}}$
- Orientation: $\mathrm{atan2}(\mathbf{d}_y, \mathbf{d}_x)$



A complete execution with a robot trajectory depicted in the world (top) and in the image (bottom)

Camera matrix - 11





A complete execution with a robot trajectory depicted in the world (top) and in the image (bottom). A Square is drawn on the floor to check correctness of the calculated vanishing point x (see previous questions)