TSBB06 Multidimensional Signal Analysis

Lecture 2D: Stereo Geometry

Per-Erik Forssén, docent
Avdelningen för Datorseende
Institutionen för Systemteknik
Linköpings Universitet

LINKÖPING UNIVERSITY

Epipolar geometry

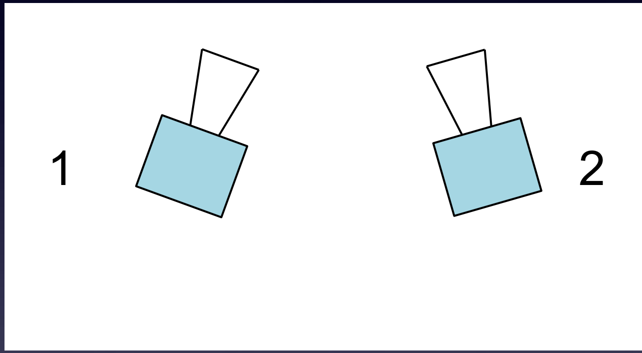
- Epipolar geometry is the geometry of ***two cameras*** (stereo cameras) that image the same scene

- Three or more cameras:
  - Multi-view geometry (see TSBB15 Computer Vision)

- Basic assumption:
  - Images are taken from **different positions**
    $\Rightarrow$ The cameras have **distinct camera centers**

Possible camera configurations

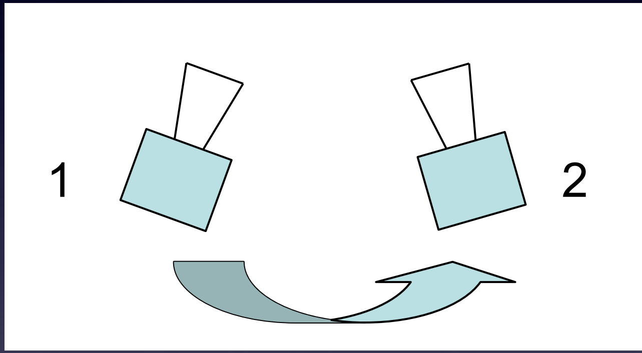
Two cameras:



- different internal parameters
- Images taken at the same time

$\Rightarrow$ Non-static scene is allowed

Motion stereo:



- One moving camera
- Images taken at different times

$\Rightarrow$ Scene must be static

Common camera motion patterns

Turntable: The scene is rotated

Image 1

Image 2

The camera moves along the principal axis

Image 1

Image 2

The camera moves “sideways”

Image 1

Image 2

Epipolar geometry

Two basic issues in epipolar geometry:

- The ***correspondence problem***:
  How can we know if a point in image 1 and another point in image 2 correspond to the same 3D point?

- The ***reconstruction problem***:
  Given that a pair of points in image 1 and 2 correspond, what 3D point do they depict?

Basic setup

- Let $\mathbf{C}_1$ and $\mathbf{C}_2$ be the camera matrices of the two cameras
- Let $\mathbf{x}$ be the homogeneous coordinates of a 3D point
- Let $\mathbf{y}_1$ and $\mathbf{y}_2$ be the homogeneous coordinates of the projections of $\mathbf{x}$ in image 1 and 2
- Let $\mathbf{n}_1$ and $\mathbf{n}_2$ be the homogeneous coordinates of the camera centers

$$\mathbf{y}_1 \sim \mathbf{C}_1\mathbf{x} \qquad \mathbf{C}_1\mathbf{n}_1 = \mathbf{0}$$
$$\mathbf{y}_2 \sim \mathbf{C}_2\mathbf{x} \qquad \mathbf{C}_2\mathbf{n}_2 = \mathbf{0}$$

Projection rays

- If $\mathbf{y}_2$ is known, what can be said about $\mathbf{x}$?
- We know that $\mathbf{x}$ lies somewhere on **a 3D line**:
  - Passes through: $\mathbf{n}_2$
  - Passes through: $\mathbf{C}_2^+\mathbf{y}_2$
- Parametric representation of the line:

$$\mathbf{x} = (1-t)\mathbf{n}_2 + t\mathbf{C}_2^+\mathbf{y}_2$$

- Pseudo-inverse:

$$\mathbf{C}\mathbf{C}^+ = \mathbf{I} \Rightarrow \mathbf{C}_2^+ = \mathbf{C}_2^T(\mathbf{C}_2\mathbf{C}_2^T)^{-1}$$

These two points are always distinct!

Projection ray

$$\mathbf{x} = (1-t)\mathbf{n}_2 + t\mathbf{C}_2^{+}\mathbf{y}_2$$

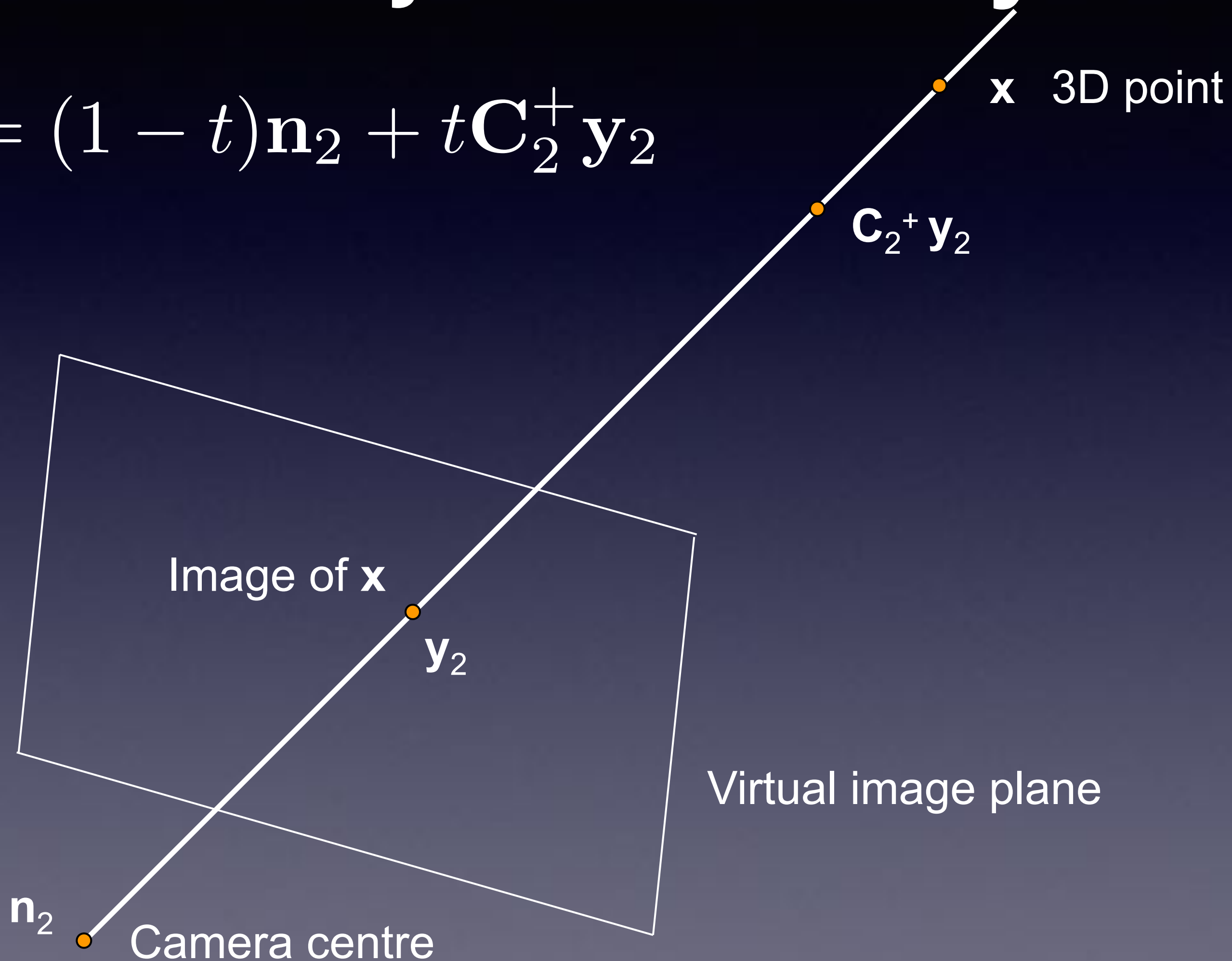


Image of a line

- What is the image of this line in camera 1?
- The parametric 3D point is mapped to $\mathbf{y}'_1(t)$ in image 1:

$$\mathbf{y}'_1(t) \sim \mathbf{C}_1[(1-t)\mathbf{n}_2 + t\mathbf{C}_2^+\mathbf{y}_2]$$

$$\mathbf{y}'_1(t) \sim (1-t)\underline{\mathbf{C}_1\mathbf{n}_2} + t\underline{\mathbf{C}_1\mathbf{C}_2^+\mathbf{y}_2}$$

Two points in image 1

Image of a line

- Form 2D line
  $$\mathbf{l}_1 = (\mathbf{C}_1\mathbf{n}_2) \times (\mathbf{C}_1\mathbf{C}_2^+\mathbf{y}_2)$$
- Easy to see that: $\mathbf{y}_1'(t)^T\mathbf{l}_1 = 0 \text{ for all } t$
- Thus $\mathbf{l}_1$ represents the 2D line $\mathbf{y}_1'(t)$
- This is a general result:
  - The image of a 3D line is a 2D line (**exception?**)
- as $\underline{\mathbf{y}_1} = \mathbf{C}_1\mathbf{x}$ it follows that $\mathbf{y}_1^T\mathbf{l}_1 = 0$

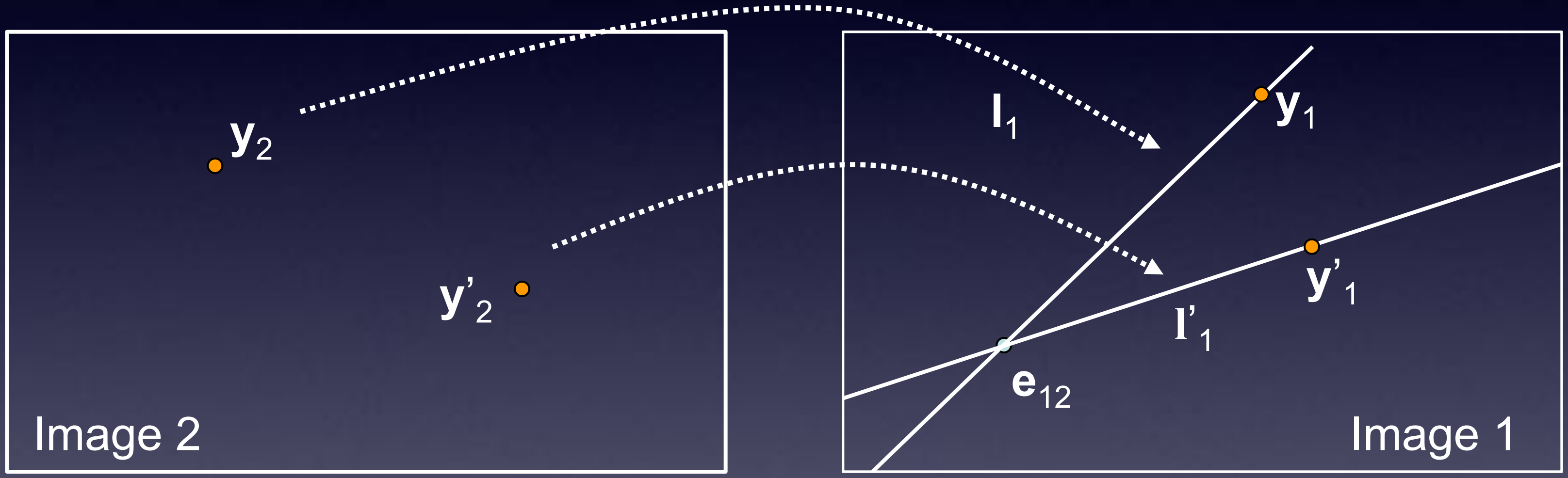
$\mathbf{y}_1$ is the image of $\mathbf{x}$ in image 1

Conclusions

- If we know $\mathbf{y}_2$, with $\mathbf{y}_2 = \mathbf{C}_2\mathbf{x}$, we know that $\mathbf{y}_1$ lies on a line $\mathbf{l}_1$ in image 1
- The line $\mathbf{l}_1$ depends on $\mathbf{y}_2$
- $\mathbf{l}_1$ is called an *epipolar line*
- All epipolar lines in image 1 intersect the point $\mathbf{e}_{12} = \mathbf{C}_1\mathbf{n}_2$
- $\mathbf{e}_{12}$ is called *epipolar point* or *epipole*
- Symmetry between image 1 and image 2

Epipolar lines and points

$\mathbf{y}_1$ and $\mathbf{y}_2$ are corresponding (to the same 3D point $\mathbf{x}$)
$\mathbf{y}'_1$ and $\mathbf{y}'_2$ are corresponding



$\mathbf{y}_2$ generates epipolar line $\mathbf{l}_1$ in image 1
$\mathbf{y}'_2$ generates epipolar line $\mathbf{l}'_1$ in image 1

Both epipolar lines intersect at epipolar point $\mathbf{e}_{12}$

$\mathbf{y}_1$ lies on $\mathbf{l}_1$ and $\mathbf{y}'_1$ lies on $\mathbf{l}'_1$

More conclusions

- The mapping from a point $\mathbf{y}_2$ to a line $\mathbf{l}_1$:

$$\mathbf{l}_1 = (\mathbf{C}_1\mathbf{n}_2) \times (\mathbf{C}_1\mathbf{C}_2^+\mathbf{y}_2)$$

$$\mathbf{l}_1 = [\mathbf{e}_{12}]_\times \mathbf{C}_1\mathbf{C}_2^+\mathbf{y}_2$$

$\mathbf{l}_1$ is directly given by a linear mapping of $\mathbf{y}_2$!

The fundamental matrix

- This 3 × 3 mapping is called the *fundamental matrix*, denoted **F**.
- usage:

$$\mathbf{l}_1 = \mathbf{F}\mathbf{y}_2$$

- where:

$$\mathbf{F} = [\mathbf{e}_{12}]_{\times}\mathbf{C}_1\mathbf{C}_2^{+}$$

F depends only on the camera matrices $\mathbf{C}_1$ and $\mathbf{C}_2$ ($\mathbf{e}_{12}$ depends on $\mathbf{C}_1$ and $\mathbf{C}_2$)

The epipolar constraint

- If $\mathbf{y}_1$ and $\mathbf{y}_2$ correspond to the same 3D point $\mathbf{x}$ we have:

$$\mathbf{y}_1^T \mathbf{l}_1 = 0$$

- and thus

$$\mathbf{y}_1^T \mathbf{F} \mathbf{y}_2 = 0$$

Epipolar constraint

- This relation must always be satisfied for points $\mathbf{y}_1$ and $\mathbf{y}_2$ if they correspond to the same 3D point.

The epipolar constraint

- The epipolar constraint a necessary ***but not sufficient*** condition for correspondence:

$$\mathbf{y}_1 \leftrightarrow \mathbf{y}_2 \;\Rightarrow\; \mathbf{y}_1^T \mathbf{F} \mathbf{y}_2 = 0$$

- Other points on an epipolar line also satisfy the epipolar constraint.

Summary so far

- If $\mathbf{C}_1$ and $\mathbf{C}_2$ are known, $\mathbf{F}$ can be determined
- Given that $\mathbf{F}$ is known, we can test if a points in image 1 and a point in image 2 correspond to the same 3D point
- Given a point $\mathbf{y}_2$ in image 2, the corresponding point $\mathbf{y}_1$ lies on an epipolar line $\mathbf{l}_1$ in image 1
- All epipolar lines in image 1 intersect at the epipolar point $\mathbf{e}_{12}$
- $\mathbf{l}_1$ is given by $\mathbf{l}_1 = \mathbf{F}\mathbf{y}_2$

Symmetry

- In the previous derivation we saw that a point in image 2 defines an epipolar line in image 1
- Due to symmetry, we can instead start with a point in image 1 and find an epipolar line in image 2

$$\mathbf{l}_2 = \mathbf{F}^T \mathbf{y}_1$$

$$\mathbf{F}^T = [\mathbf{e}_{21}]_\times \mathbf{C}_2 \mathbf{C}_1^+$$

$\mathbf{e}_{21} = \mathbf{C}_2 \mathbf{n}_1$

Properties of F

- The epipoles span the left and right null spaces of **F**:
  - From $\mathbf{F} = [\mathbf{e}_{12}]_\times \mathbf{C}_1 \mathbf{C}_2^+$ follows that $\mathbf{e}_{12}$ is a left null vector
  - and by symmetry: $\mathbf{F}\mathbf{e}_{21} = \mathbf{0}$
- The rank of **F** is 2:
  - rank $[\mathbf{e}_{12}]_\times = 2 \Rightarrow$ rank $\mathbf{F} = 2 \Rightarrow$ det $\mathbf{F} = 0$
- **F** has 7 degrees of freedom (why?)
  - **F** is a projective element in $P(\mathbb{R}^9)$
  - **F** is rank deficient

Two ways to determine F

- The calibrated case:
  - **F** can be computed from $\mathbf{C}_1$ and $\mathbf{C}_2$
- The uncalibrated case:
  - Given a set of *K* corresponding image points, $\mathbf{y}_{1k}$ in image 1 and $\mathbf{y}_{2k}$ in image 2,
  - it is possible to determine **F** from the constraints:

$$\mathbf{y}_{1,k}^T \mathbf{F} \mathbf{y}_{2,k} = 0, k = 1, \ldots, K$$

The uncalibrated case

- No camera matrices need to be known
- Image coordinates can only be determined up to a certain accuracy due to:
  - **detection inaccuracy**
    (e.g. quantization to integer pixel coordinates)
  - **model errors**
    lens distortion, rolling shutter etc.
- This accuracy affects the estimation of **F**

Estimation of **F**

- Let **y** and **y'** be corresponding points in image 1 and image 2

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \quad \mathbf{y}' = \begin{pmatrix} y'_1 \\ y'_2 \\ y'_3 \end{pmatrix} \quad \mathbf{F} = \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix}$$

Estimation of $\mathbf{F}$

- Let **y** and **y'** be corresponding points in image 1 and image 2

$$\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \quad \mathbf{y}' = \begin{pmatrix} y'_1 \\ y'_2 \\ y'_3 \end{pmatrix} \quad \mathbf{F} = \begin{pmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{pmatrix}$$

$$\mathbf{y}'^T\mathbf{F}\mathbf{y} =$$

$$y_1 y'_1 f_{11} + y'_2 y_1 f_{21} + y'_3 y_1 f_{31} +$$
$$y_1 y'_2 f_{12} + y'_2 y_2 f_{22} + y'_3 y_2 f_{32} +$$
$$y_1 y'_3 f_{13} + y'_2 y_3 f_{23} + y'_3 y_3 f_{33}$$

Estimation of $\mathbf{F}$

- The epipolar constraint: $\mathbf{Y}^T \mathbf{F}_{vec} = 0$

$$\mathbf{Y} = \begin{pmatrix} y'_1 y_1 \\ y'_2 y_1 \\ y'_3 y_1 \\ y'_1 y_2 \\ y'_2 y_2 \\ y'_3 y_2 \\ y'_1 y_3 \\ y'_2 y_3 \\ y'_3 y_3 \end{pmatrix} \quad \mathbf{F}_{\text{vec}} = \begin{pmatrix} f_{11} \\ f_{21} \\ f_{31} \\ f_{12} \\ f_{22} \\ f_{32} \\ f_{13} \\ f_{23} \\ f_{33} \end{pmatrix}$$

- One linear constraint on the elements of **F**.

Estimation of $\mathbf{F}$

- Conclusion: each pair of corresponding points $\mathbf{y}_{1k}$, $\mathbf{y}_{2k}$ in the two images gives us ***a linear homogeneous constraint*** on $\mathbf{F}_{\text{vec}}$. Stack these:

$$\begin{pmatrix} - & \mathbf{Y}_1^T & - \\ & \vdots & \\ - & \mathbf{Y}_K^T & - \end{pmatrix} \mathbf{F}_{\text{vec}} = 0$$

Estimation of $\mathbf{F}$

- Conclusion: $\mathbf{F}_{\text{vec}}$ must satisfy the linear homogeneous equation

$$\mathbf{A}\mathbf{F}_{\text{vec}} = \mathbf{0} \quad \Rightarrow \quad \mathbf{A}^T\mathbf{A}\mathbf{F}_{\text{vec}} = \mathbf{0}$$

where $\mathbf{A}$ is the $K \times 9$ matrix that contains $\mathbf{Y}_k^{\mathrm{T}}$ for $k = 1, \ldots, K$ in its rows

- $\mathbf{F}_{\text{vec}}$ is an eigenvector of $\mathbf{A}^{\mathrm{T}}\mathbf{A}$, of eigenvalue zero
  Or: $\mathbf{F}_{\text{vec}}$ is a right singular vector of $\mathbf{A}$, of singular value zero

The 8-point algorithm

Given K pairs of corresponding points $\mathbf{y}_{1k}$, $\mathbf{y}_{2k}$

1. Form $\mathbf{Y}_k$ from these pairs for $k = 1, \ldots, K$ and then $\mathbf{A}$ from all $\mathbf{Y}_k$ (row-wise)

2. $\mathbf{F}_{\text{vec}}$ = the eigenvector corresponding to the smallest eigenvalue of $\mathbf{A}^T\mathbf{A}$
(or the right singular vector corresponding to the smallest singular value of $\mathbf{A}$)

3. Reshape $\mathbf{F}_{\text{vec}}$ to a 3×3 matrix $\mathbf{F}$.

This $\mathbf{F}$ is an estimate of the fundamental matrix

The 8-point algorithm: Details

- Since **A** is $K \times 9$

$$\mathbf{A}^T\mathbf{A}\,\mathbf{F}_{vec} = \mathbf{0}$$

has a unique solution $\mathbf{F}_{vec}$ if $K \geq 8$.

This is why it is called the 8-point algorithm

- Special configurations of $\mathbf{x}_k$ make **F** not unique
- The 3D points $\mathbf{x}_k$ that generate $\mathbf{y}_{1k}$, $\mathbf{y}_{2k}$ must be in general positions (e.g. not in a plane, or all at infinity)

The 8-point algorithm: Details

- We know: det $\mathbf{F} = 0$
- In practice, the image coordinates $\mathbf{y}_1$, $\mathbf{y}_2$ cannot be measured exactly
  - det $\mathbf{F} = 0$ is not valid automatically when $\mathbf{F}$ is estimated according to above
- If det $\mathbf{F} \neq 0$:
  - $\mathbf{F}$ cannot be related to some camera matrices
  - $\mathbf{F}$ does not describe well-defined epipoles
- We need to *enforce* det $\mathbf{F} = 0$
  - Find $\mathbf{F}_0$ that is closest to $\mathbf{F}$, with det $\mathbf{F}_0 = 0$

Enforcement of det **F** = 0

F is a 3 × 3 matrix with det **F** ≠ 0

An SVD of **F** gives us: **F** = **U S V**^{T}

U and **V** are orthogonal matrices
S is a diagonal matrix that holds the singular values σ_1 , σ_2 , $\sigma_3 > 0$

det **F** = $\pm\, \sigma_1 \cdot \sigma_2 \cdot \sigma_3$

In normal cases:
σ_1 and σ_2 are relatively large and σ_3 is small but not = 0

Enforcement of det $\mathbf{F} = 0$

Set the smallest singular value to zero and recombine:

$$\mathbf{F}_0 = \mathbf{U} \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \mathbf{V}^T$$

- $\mathbf{F}_0$ is the closest approximation of **F** with det $\mathbf{F}_0 = 0$ (in Frobenius norm)

The 8-point algorithm
Full picture

Given $K \geq 8$ pairs of corresponding points $\mathbf{y}_{1k}$, $\mathbf{y}_{2k}$

1. Form $\mathbf{Y}_k$ from each pair for $k = 1, \ldots, K$ and stack these to form $\mathbf{A}$
2. $\mathbf{F}_{\text{vec}}$ = the right singular vector corresponding to the smallest singular value of $\mathbf{A}$ (ideally zero)
3. Reshape $\mathbf{F}_{\text{vec}}$ to a 3×3 matrix $\mathbf{F}$.
4. Enforce $\det \mathbf{F} = 0 \Rightarrow \mathbf{F}_0$
5. This $\mathbf{F}_0$ is our estimate of the fundamental matrix

The uncalibrated case, summary

- Given a set of $K \geq 8$ correspondences, we can estimate an **F** that fits these points
  - The 8-point algorithm

- As the image coordinates are perturbed by noise, the estimated **F** will not satisfy $\mathbf{Y}_k^T \mathbf{F} = 0$ exactly, but $\mathbf{F}_{vec}$ minimizes

$$\epsilon = \|\mathbf{A}\mathbf{F}_{\text{vec}}\|$$

(at least before the constraint enforcement)

- Note that this is an *algebraic error* (**What is this?**)

Hartley normalisation

- To get useful estimates of **F**, we need to use Hartley normalisation of the image coordinates:
  - Translate origin to the centroid of the points in each image
  - Scale each image so that average distance to origin = $2^{1/2}$
- Estimate **F** in the transformed coordinates and then transform **F** back to standard coordinates
- More advanced methods: 7-point algorithm, gold-standard algorithm (TSBB15)

BREAK

Stereo rig

- A general stereo rig consists of two cameras with
  - distinct camera centers
  - general orientations of the camera principal axes (although often toward a common scene!)

Stereo rig

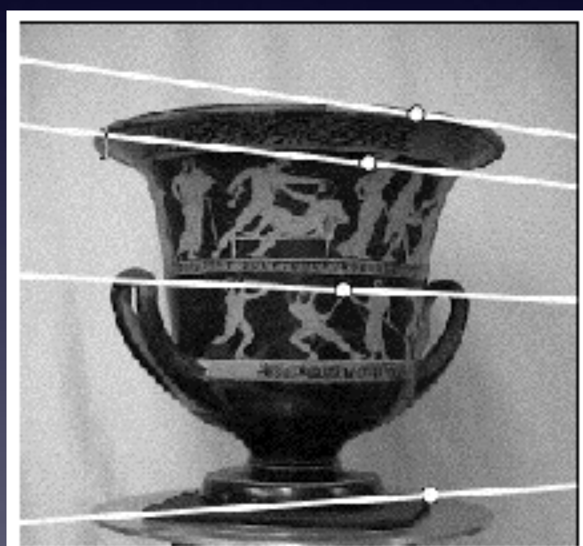
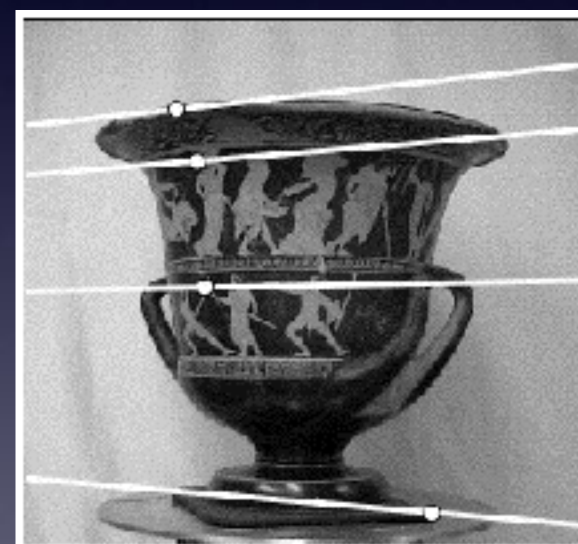
Research stereo rig, Eddie, ISY/LiU

FujiFilm consumer stereo camera
Finepix W3

Point Grey, Bumblebee classic stereo camera

For a general stereo rig

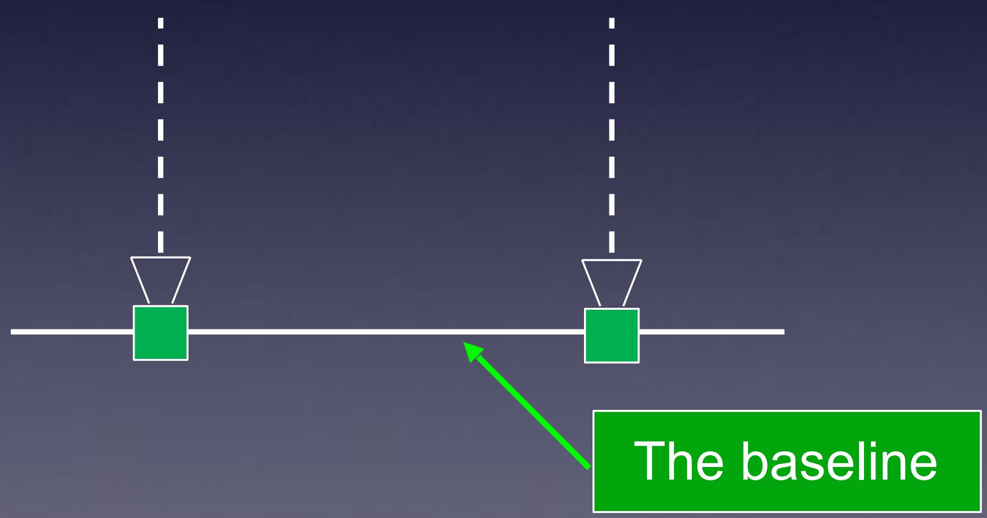
- In general the epipolar lines are not parallel
  - Intersect at the epipole

  - In this example, the cameras are *convergent* (or ”inwards pointing”).

Rectified stereo rig

- In a *rectified stereo rig*, the principal directions of the cameras are parallel and orthogonal to the baseline and the cameras have identical intrinsics



Rectified stereo images

For a rectified stereo rig, corresponding image points lie on the same row. This means that

- The epipolar points are points at infinity
- The epipolar lines are parallel
- More precisely:

$$\mathbf{e}_{12} = \mathbf{e}_{21} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Infinitely far to the left ***and*** to the right!

Rectified stereo images

- The corresponding fundamental matrix is

$$\mathbf{F}_{\mathsf{R}} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

- Note that $\mathbf{y}_1^T \mathbf{F}_R \mathbf{y}_2 = 0$ for all vectors $\mathbf{y}_1$, $\mathbf{y}_2$ with

$$\mathbf{y}_1 = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \quad \mathbf{y}_2 = \begin{pmatrix} u + d \\ v \\ 1 \end{pmatrix}$$

Rectified stereo rig

- A rectified stereo rig is difficult to build in practise:
  - requires expensive high precision mechanical alignment: E:g. 1/100 of a degree for a 4K camera with 45° hfov
  - a mechanical rig may lose its rectification if bumped into
- Typically one instead sets up an approximate rectified stereo rig, and does rectification in software.

Equivalent cameras

- Let **C** and **C’** be the camera matrices of two pinhole cameras with the *same camera centre,* **n**:

$$\mathbf{y} \sim \mathbf{C}\mathbf{x} \qquad \mathbf{C}\mathbf{n} = \mathbf{0}$$

$$\mathbf{y}' \sim \mathbf{C}'\mathbf{x} \quad \mathbf{C}'\mathbf{n} = \mathbf{0}$$

- Given **y**’, we have the following parametric form for projection ray:

$$\mathbf{x} = t\mathbf{n} + (1-t)\mathbf{C}'^{+}\mathbf{y}'$$

Equivalent cameras

- This set of points is projected into camera **C** as

$$\mathbf{y} \sim \mathbf{C}[t\mathbf{n} + (1-t)\mathbf{C}'^{+}\mathbf{y}']$$

$$\mathbf{y} \sim (1-t)\mathbf{C}\mathbf{C}'^{+}\mathbf{y}' = (1-t)\mathbf{H}\mathbf{y}' \sim \mathbf{H}\mathbf{y}'$$

- There is a homography mapping **H** from **y** to **y'** defined by the cameras **C** and **C'** (**how come?**)

$$\mathbf{y} \sim \mathbf{H}\mathbf{y}' \quad \Leftrightarrow \quad \mathbf{y}' \sim \mathbf{H}^{-1}\mathbf{y}$$

- The images in the two cameras are identical except for a homography mapping of the coordinates.
⇒ One can be converted to the other by resampling.

Rectified stereo rig

Consequently:

- All cameras that share the same camera center are in this sense "equivalent"
- E.g. if a camera rotates about its center by 3D rotation **R**, the image transforms according to a homography $\mathbf{H} = \mathbf{K}\,\mathbf{R}\,\mathbf{K}^{-1}$
  - where **K** is the intrinsic camera matrix

Rectified images

Consequence:

- If the principal axis of a camera is not exactly pointing in the right direction, this can be compensated for by applying a suitable homography **H** on the image coordinates
  - Implies a rotation of the principal axis
  - This can make the epipolar lines parallel
- The result is a *rectified image*

Stereo rectification

- We now wish to determine homographies $\mathbf{H}_1$ for image 1 and $\mathbf{H}_2$ for image 2 that rectify the two images
- Estimate $\mathbf{F}$ from corresponding points in the two images
  - The 8-point algorithm
- Find $\mathbf{H}_1$, $\mathbf{H}_2$ such that $(\mathbf{H}_1^{-1})^T \mathbf{F} \mathbf{H}_2^{-1} \sim \mathbf{F}_\mathrm{R}$

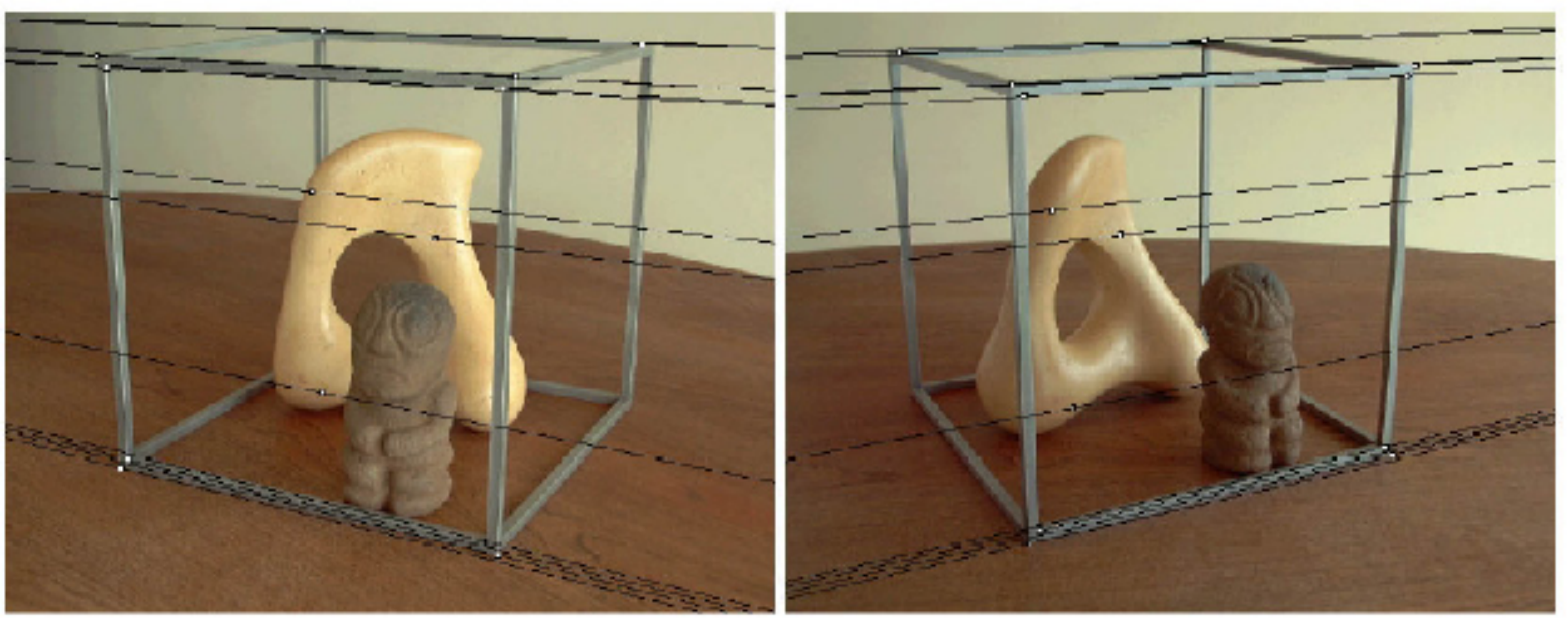
Stereo rectification

- This relation in $\mathbf{H}_1$ and $\mathbf{H}_2$ has multiple solutions, many of which are unwanted, e.g.:
  - horizontal mirroring
  - extreme geometric distortion

- Several methods for determining useful $\mathbf{H}_1$ and $\mathbf{H}_2$ from $\mathbf{F}$ exist, for example:
  - Loop & Zhang, **Computing Rectifying Homographies for Stereo Vision**, **ICPR** 1999
    Determines $\mathbf{H}_1$ and $\mathbf{H}_2$ by minimising geometric distortion, see computer exercise D

Stereo rectification

Example of an unrectified stereo image pair



Black lines are epipolar lines. Not parallel

From Loop & Zhang

Stereo rectification

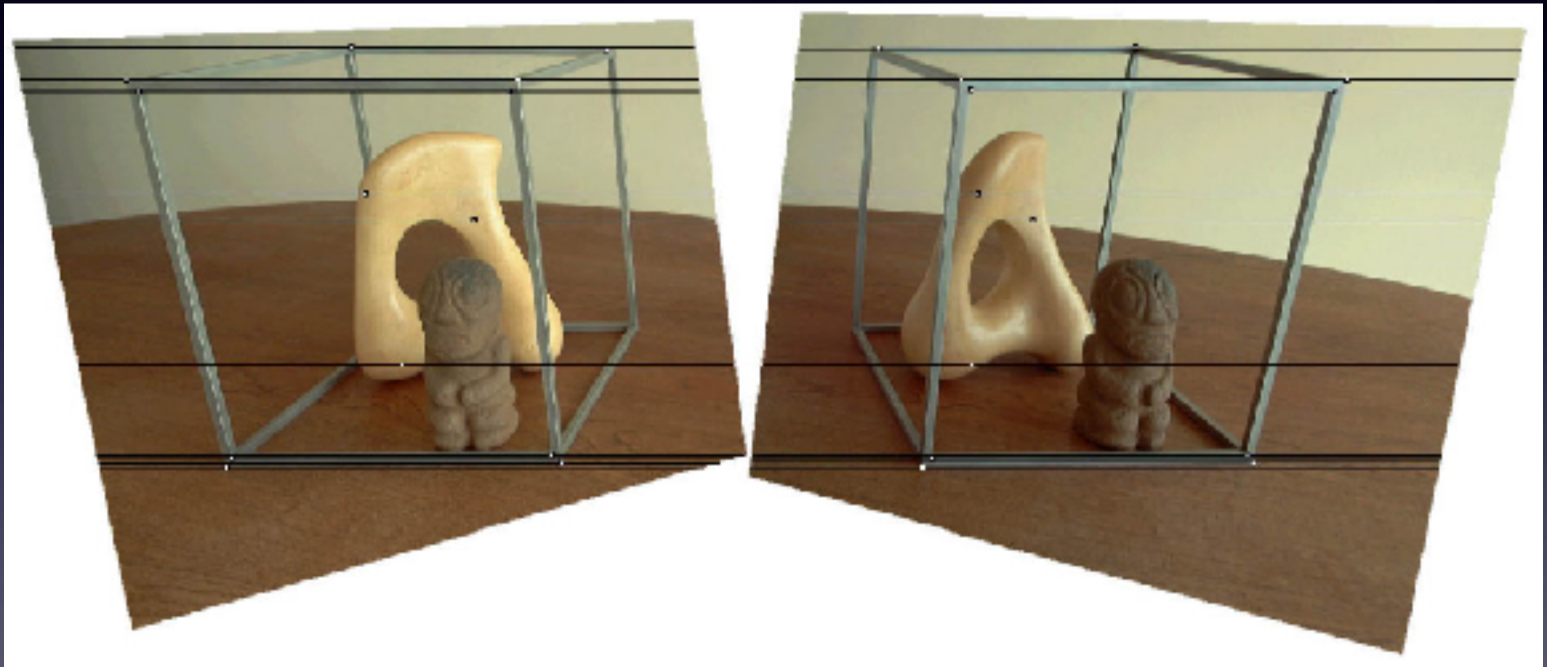
Example of a rectification

Epipolar lines are parallel and aligned!

From Loop & Zhang

Stereo rectification

Another example, with less geometric distortion than the previous one



Epipolar lines are parallel and aligned!

From Loop & Zhang

Stereo rectification, summary

- A pair of stereo images that are not rectified:
  - the principal axes are not parallel and not perpendicular to the baseline
- can be rectified by homographies such that
  - corresponding points are found on the same row
- Multiple solutions to the rectification exist

Reconstruction

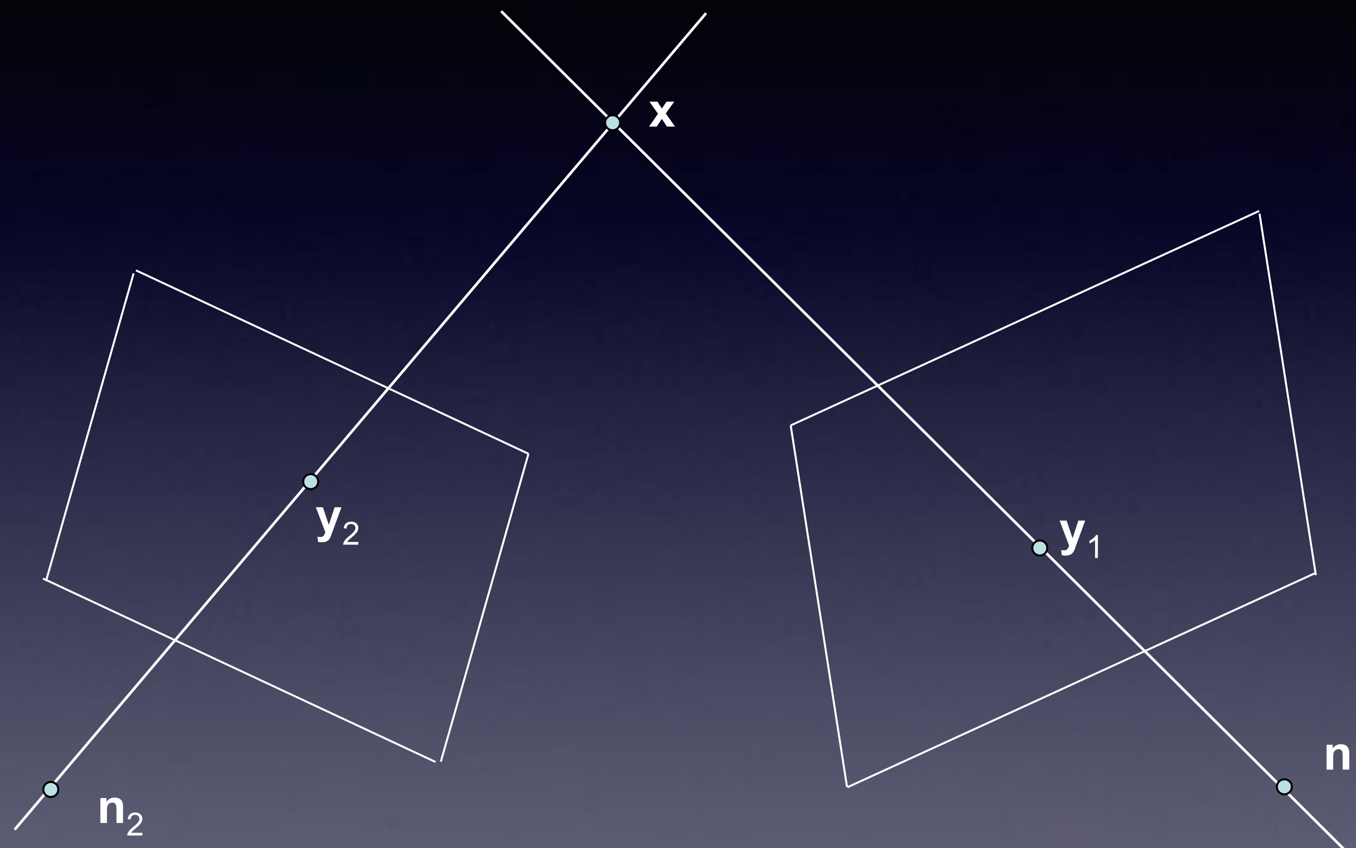
- Given a pair of corresponding image points $\mathbf{y}_1$ and $\mathbf{y}_2$

$$\begin{cases} \mathbf{y}_1 \sim \mathbf{C}_1\mathbf{x} \\ \mathbf{y}_2 \sim \mathbf{C}_2\mathbf{x} \end{cases}$$

we know that: $\mathbf{y}_1^T\mathbf{F}\mathbf{y}_2 = 0$

- But what about **x**? Can **x** be determined?
- This problem is called ***triangulation***.

Reconstruction



The epipolar constraint ⇔ the two projection rays intersect

In this case: there is a unique **x** that projects to $\mathbf{y}_1$ and $\mathbf{y}_2$

Reconstruction

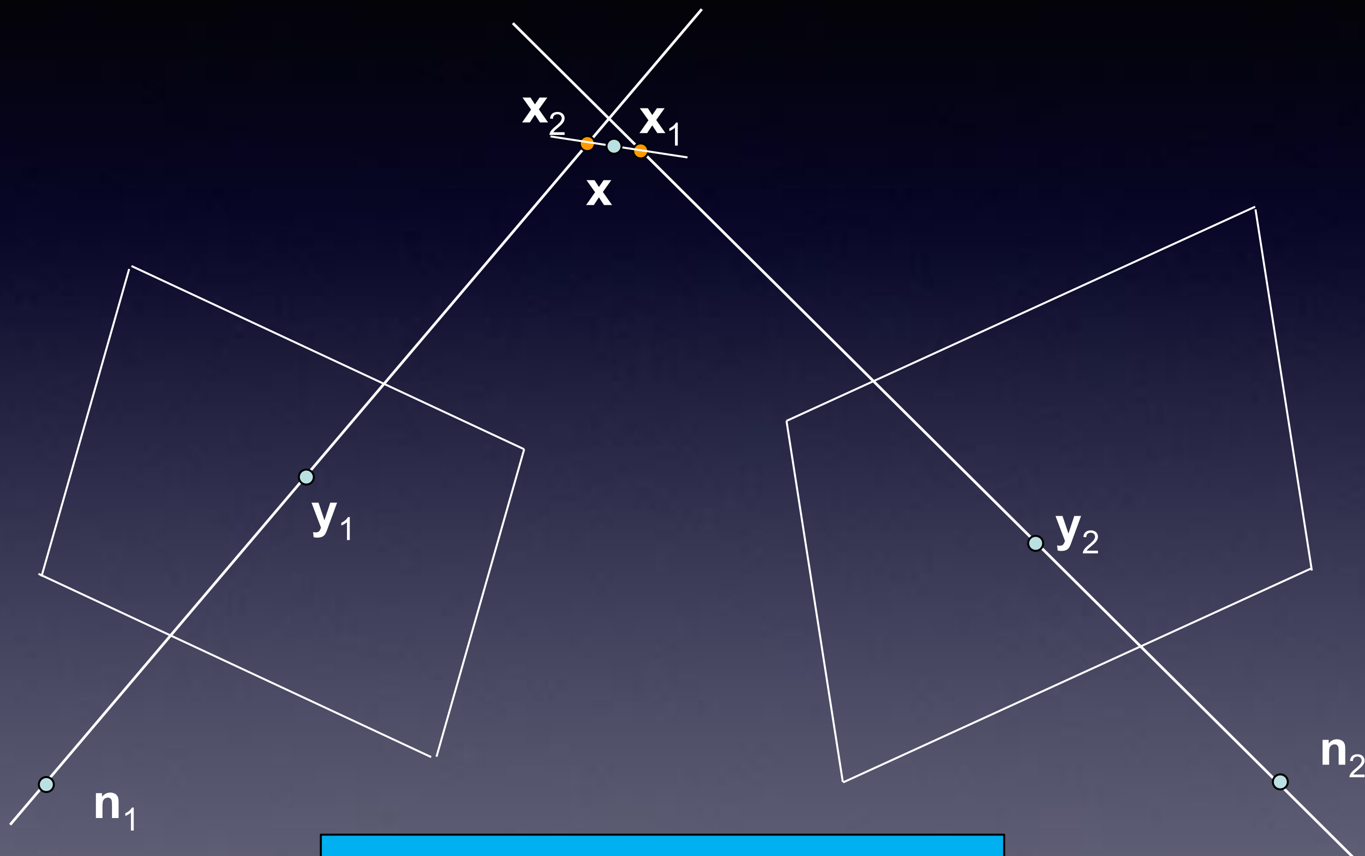
- In reality, the image points $\mathbf{y}_1$ and $\mathbf{y}_2$ do not satisfy $\mathbf{y}_1^T\mathbf{F}\mathbf{y}_2 = 0$ exactly
  - Lens distortion
  - Coordinate quantization
  - Estimation inaccuracy

- The two projection rays do not intersect
  In this case: $\mathbf{x}$ is not well defined
  It has somehow to be estimated

The mid-point method

- Find the unique points $\mathbf{x}_1$ and $\mathbf{x}_2$ on the two projection rays that are closest to the other ray
- Set $\mathbf{x}$ = the mid-point between $\mathbf{x}_1$ and $\mathbf{x}_2$
- If $\mathbf{x}_1 = \mathbf{x}_2$, $\mathbf{y}_1^T\mathbf{F}\mathbf{y}_2 = 0$

The mid-point method



$\mathbf{y}_1$ and $\mathbf{y}_2$ approximately satisfy the epipolar constraint

Linear triangulation

From
$$\begin{cases} \mathbf{y}_1 \sim \mathbf{C}_1\mathbf{x} \\ \mathbf{y}_2 \sim \mathbf{C}_2\mathbf{x} \end{cases}$$

follows
$$\begin{cases} \mathbf{0} = \mathbf{y}_1 \times \mathbf{C}_1\mathbf{x} \\ \mathbf{0} = \mathbf{y}_2 \times \mathbf{C}_2\mathbf{x} \end{cases}$$

or
$$\begin{cases} \mathbf{0} = [\mathbf{y}_1]_\times \mathbf{C}_1\mathbf{x} \\ \mathbf{0} = [\mathbf{y}_2]_\times \mathbf{C}_2\mathbf{x} \end{cases}$$

3+3 = 6 linear homogeneous equations in x

Linear triangulation

Since $[\mathbf{y}_1]_\times$ has rank 2: one of the 3 equations is linearly dependent to the other two:

$$\begin{cases} \mathbf{0} = [\mathbf{y}_1]_\times \mathbf{C}_1 \mathbf{x} \\ \mathbf{0} = [\mathbf{y}_2]_\times \mathbf{C}_2 \mathbf{x} \end{cases}$$

In total: 4 linear independent homogeneous equations in x

This can be written

$$\mathbf{B}\mathbf{x} = \mathbf{0}$$

B is a 6 × 4 matrix

Linear triangulation

- In practice (with noise) **Bx** = **0** cannot be solved exactly, so we resort to finding an **x** that minimizes

$$\epsilon(\mathbf{x}) = \|\mathbf{B}\mathbf{x}\|$$

  with the constraint $\|\mathbf{x}\| = 1 \Rightarrow$ choose **x** as

  - the right singular vector of **B** with smallest singular value.
  - This approach is simple, but minimizes an algebraic error.

Optimal triangulation

- There is also a ***maximum likelihood*** (ML) approach to triangulation: Find the most likely 3D point **x** that could have generated the observations:

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{y}_1|\mathbf{x})p(\mathbf{y}_2|\mathbf{x})$$

- If we assume normal i.i.d. image noise we get:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} d^2(\mathbf{y}_1, \mathbf{l}_1(t))^2 + d^2(\mathbf{y}_2, \mathbf{l}_2(t))$$

- Leads to a cubic polynomial equation in *t*. This solution is known as ***optimal triangulation***. See e.g. the Hartley&Zisserman book for details.

Reconstruction, summary

- Given that $\mathbf{C}_1$ and $\mathbf{C}_2$ are known and $\mathbf{y}_1$ and $\mathbf{y}_2$ correspond to the same $\mathbf{x}$
  - they satisfy the epipolar constraint
- $\mathbf{x}$ can be determined, for example, by
  - the mid-point method (a geometric method)
  - the linear method (an algebraic method)
  - optimal triangulation (a statistical method)
- In the noise free case, these methods give the same $\mathbf{x}$
- In the real and noisy case, they do not

Computer Lab on Tuesday

On the lab you will get to try:

- Stereo rectification
- Fundamental matrix estimation
- Triangulation

Note: Extensive preparations are needed.
Check the lab sheet and review these slides.