Analytical 3d: Reconstruction

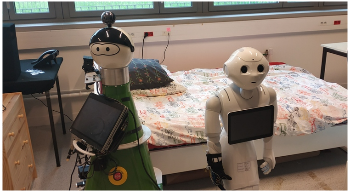
Antoine Manzanera

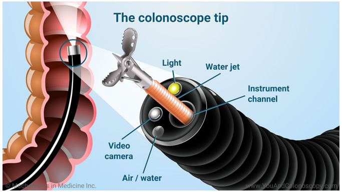
ENSTA Paris

ROB317 - 3d Computer Vision
January 2026

3d Reconstruction from Videos

Reconstructing the scene geometry from videos is useful in many applications: Robot navigation (obstacle detection), Metrology, 3d Cartography, Medicine...





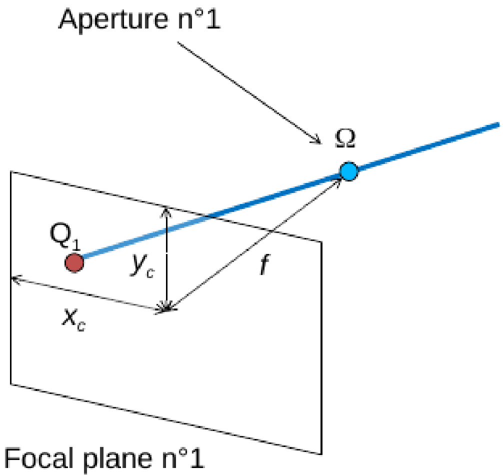
+ It is a cheap and flexible approach: One single passive camera, Adaptive baseline,...
– It strongly relies on scene structure (texture) and precise camera positioning.

Presentation Outline

1. Introduction to analytical methods
2. Projective Geometry and Camera Matrices
3. Epipolar Geometry and the Fundamental Matrix
4. Depth Estimation and Epipolar Flow

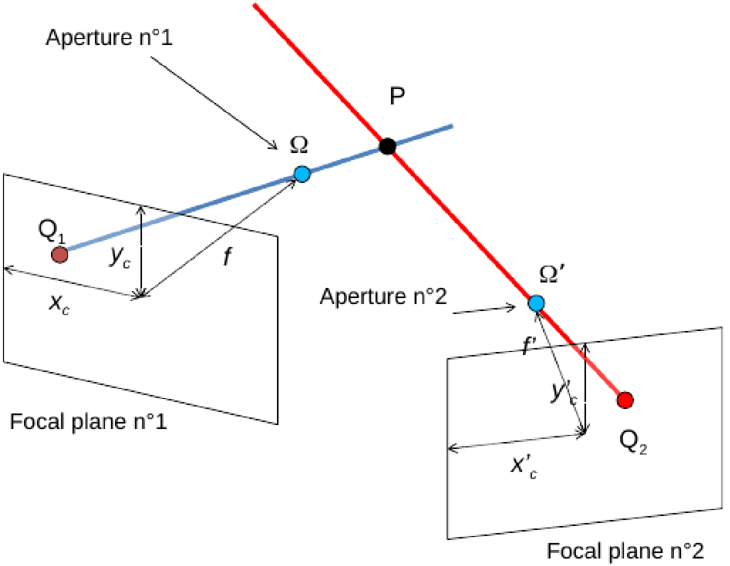
Presentation Outline

Principles of Analytical Methods



The geometry of the camera (intrinsic parameters) identifies the projection line of any point in the focal plane.

Principles of Analytical Methods

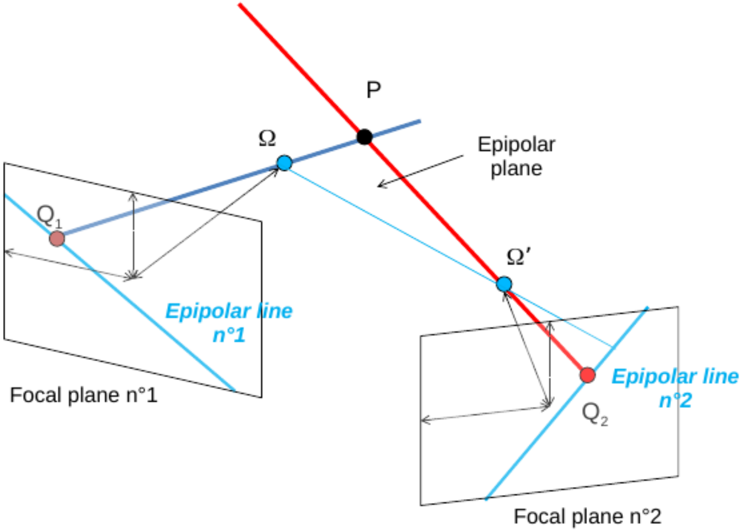


Another position of the camera (extrinsic parameters) allows to recover the 3d position of a point projected on the two focal planes:

$$\Omega P = \Omega\Omega' \frac{\sin \hat{\Omega}'}{\sin \hat{P}}$$

$$\Omega' P = \Omega\Omega' \frac{\sin \hat{\Omega}}{\sin \hat{P}}$$

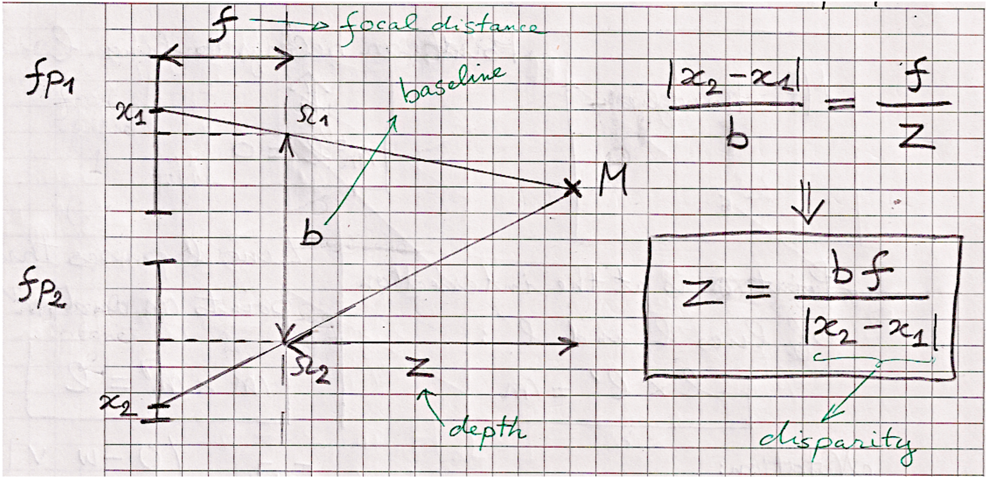
Principles of Analytical Methods



The epipolar constraints may reduce the search area for matching points. It is expressed by the fundamental matrix **F** in the projective geometry framework: $Q_2^t \mathbf{F} Q_1 = 0$.

- $\mathbf{F} Q_1$: epipolar line n.2.
- $Q_2^t \mathbf{F}$: epipolar line n.1.

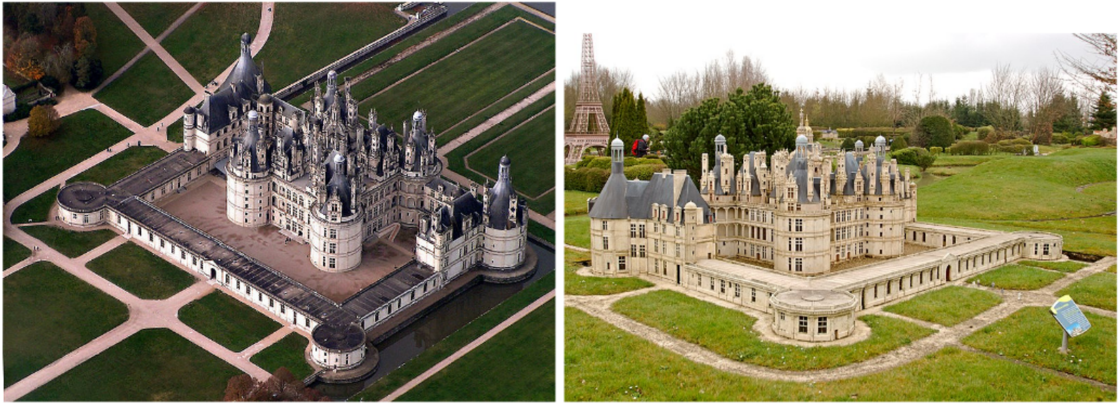
In-plane Ideal Stereovision



Ideal or Rectified or Plenoptic (Single-Lens) Stereovision

Scale Ambiguity

Without knowledge of focal and baseline, depth can at best be estimated up to scale factor!
(But look at the contextual clues...):

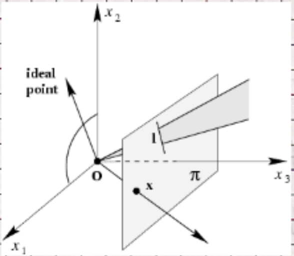


Aerial view of Chambord Castle and 1/30-scale model miniature model in La France Miniature

From **[PhD C. Pinard 2019]**

Presentation Outline

Projective Geometry in $\mathbb{P}^2$: Reminder



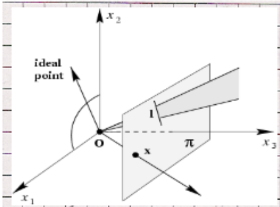
$$\mathbb{R}^2 \qquad\qquad \mathbb{P}^2$$

$$\begin{pmatrix}x\\y\end{pmatrix} \longrightarrow \begin{pmatrix}x\\y\\1\end{pmatrix}$$

$$\begin{pmatrix}u/w\\v/w\end{pmatrix} \longleftarrow \begin{pmatrix}u\\v\\w\end{pmatrix}$$

- Equivalence Classes: $\forall \lambda \neq 0 \quad \lambda x \equiv x$
- Duality point / line:
  $m = (x, y, 1)^t \qquad \ell = (a, b, c)^t$
- Ideal points: $(x, y, 0)^t$
- Line at infinity: $(0, 0, 1)^t$

Projective Geometry in $\mathbb{P}^2$: Reminder



Point m belongs to line ℓ

$$\Updownarrow$$

$$\boxed{m^t \ell = 0}$$

Point m is at the intersection of lines ℓ and ℓ':

$$\boxed{\ell \times \ell' = m}$$

Line ℓ passes through points m and m':

$$\boxed{m \times m' = \ell}$$

Notation:

pre-vector product: $[U]_\times = \begin{pmatrix} 0 & -w & v \\ w & 0 & -u \\ -v & u & 0 \end{pmatrix}$

with $U = (u, v, w)^t$

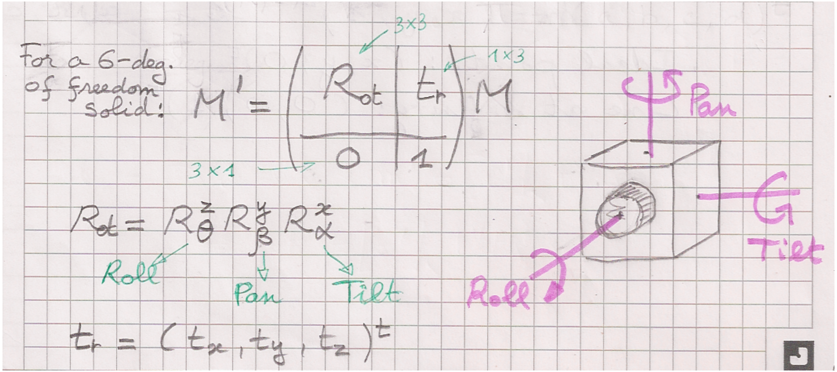
Then:

$$\boxed{U \times U' = [U]_\times U'}$$

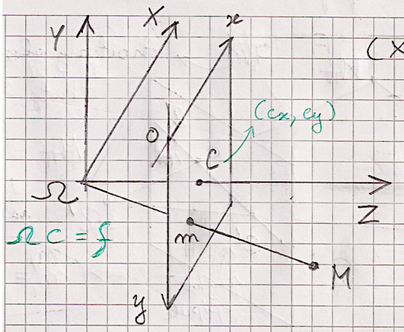
Projective Geometry in $\mathbb{P}^3$

- $\mathbb{R}^3 \leftrightarrow \mathbb{P}^3$: $(X, Y, Z) \rightarrow (X, Y, Z, 1)$; $(u/h, v/h, w/h) \leftarrow (u, v, w, h)$
- Duality point / plane: $M = (X, Y, Z, 1)^t$ / $\Pi = (a, b, c, d)$.
- Lines are defined from 2 points or from 2 planes!

$\mathbb{P}^3$ allows to express linearly affine transformations:



Camera (Calibration) Matrix: Intrinsics



$$(X,Y,Z)^t \mapsto \underbrace{\left(\frac{fX}{Z}+c_x,\ \frac{fY}{Z}+c_y\right)^t}_{m}$$

$$K = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}$$

Calibration (intrinsics) Matrix

$$\underset{\in \mathbb{P}^2}{\tilde{m}} = \begin{pmatrix} fX + Zc_x \\ fY + Zc_y \\ Z \end{pmatrix} = \underbrace{\left(\underset{3\times 3}{K} \,\middle|\, \underset{1\times 3}{0_3}\right)}_{P} \underset{\in \mathbb{P}^3}{\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}}$$

Projection and Back-Projection Matrices

$M = (X, Y, Z)^t \in \mathbb{R}^3$
$m = (x, y)^t \in \mathbb{R}^2$, and $\tilde{m} = (x, y, 1)^t \in \mathbb{P}^2$

Camera (Projection) Matrix

$$m = \pi(M) = \left(f\frac{X}{Z} + c_x, f\frac{X}{Z} + c_x\right)$$

Equivalent to:

$$\tilde{m} = KM$$

with: $K = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}$

Back-Projection Matrix

$$M = \pi^{-1}(m, Z) = \left(Z\frac{x-c_x}{f}, Z\frac{y-c_y}{f}, Z\right)$$

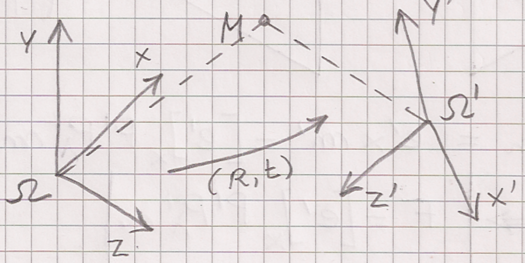
Equivalent to:

$$M = \underbrace{Z}_{\text{Depth}} \underbrace{K^{-1}\tilde{m}}_{\text{Direction}}$$

with: $K^{-1} = \begin{pmatrix} \frac{1}{f} & 0 & -\frac{c_x}{f} \\ 0 & \frac{1}{f} & -\frac{c_y}{f} \\ 0 & 0 & 1 \end{pmatrix}$

Displacement Matrix: Extrinsics

$$\tilde{m}' = \underbrace{\left(K \,\middle|\, 0_3 \right) \left(\begin{array}{c|c} R & -Rt \\ \hline 0_3^t & 1 \end{array} \right)}_{P'} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

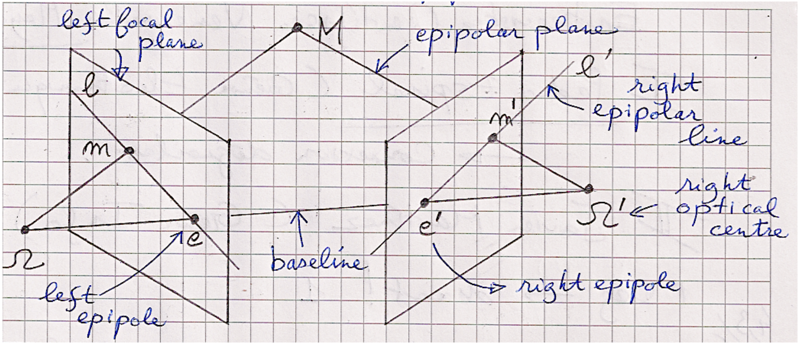


$m = PM$

$m' = P'M$

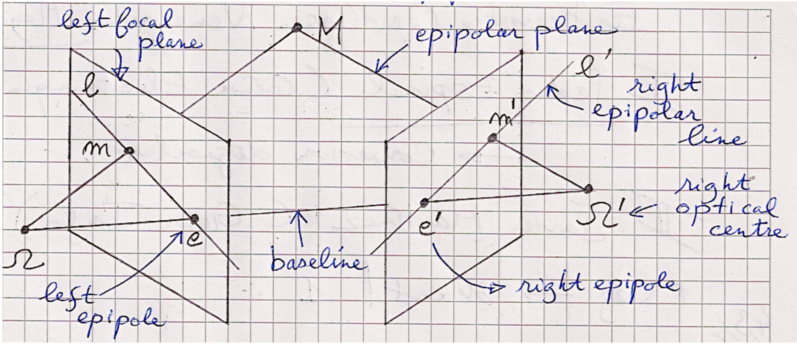
Presentation Outline

Epipolar Geometry



- Ω, m, M, m' and Ω' are coplanar.
- The epipolar plane cuts each focal plane through the epipolar line.
- Each point M has its own epipolar plane.
- All epipolar planes (epipolar pencil) intersect at the baseline $(\Omega\Omega')$

Epipolar Geometry



- The right (resp. left) epipole is the projection of the left (resp. right) optical centre on the right (resp. left) focal plane.
- All epipolar lines intersect at the epipole.

Example 1: Converging Cameras

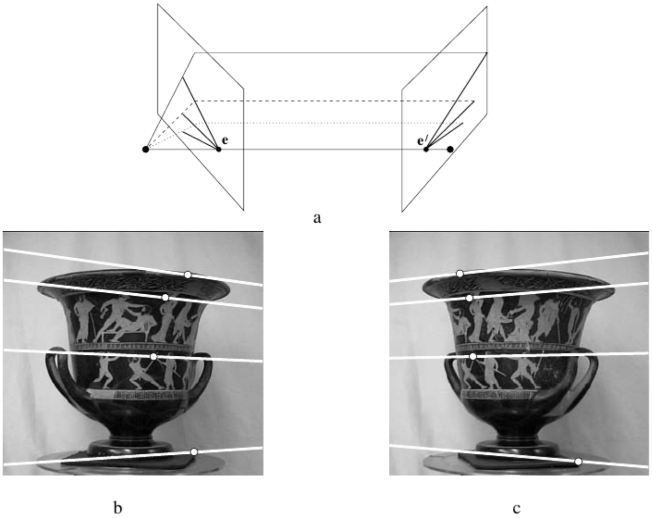


Figure from [Hartley and Zissermann 2003]

Example 2: In-Focal-Plane Moving Camera

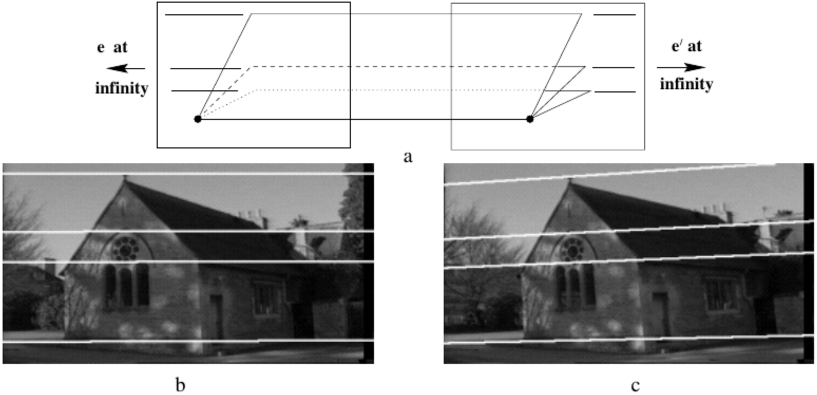


Figure from [Hartley and Zissermann 2003]

Example 3: Radially Moving Camera

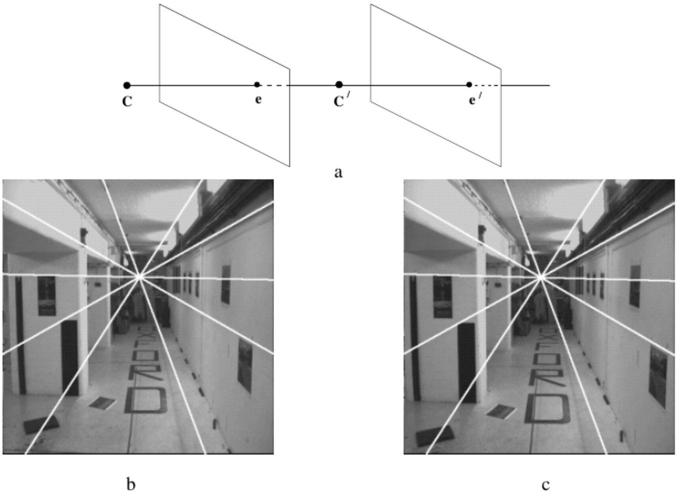
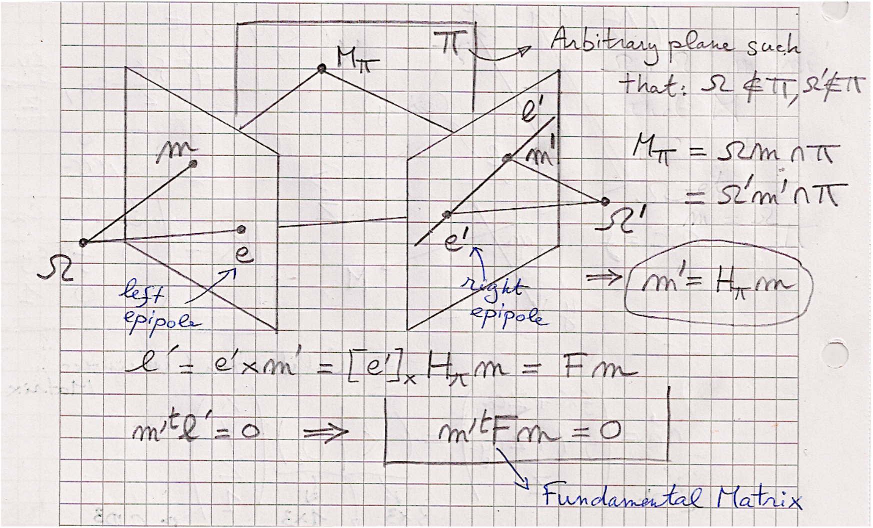
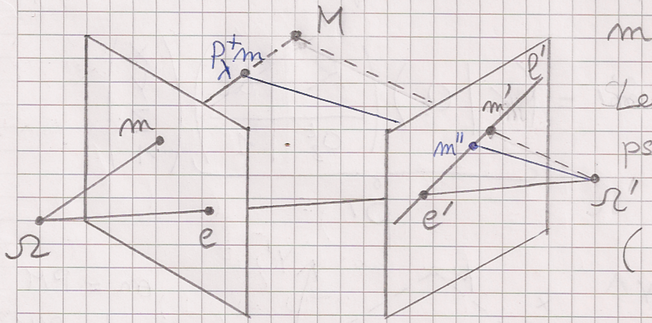


Figure from **[Hartley and Zissermann 2003]**

Fundamental matrix derived from a plane



Fundamental matrix derived from the camera matrices



$m = PM$

Let P_λ^+ the pseudo-inverse of P

$(P_\lambda^+ P = I)$

$$l' = Fm = e' \times m' = [e']_\times P' P_\lambda^+ m$$

$$\Rightarrow \boxed{F = [e']_\times P' P_\lambda^+}$$

Fundamental matrix from the camera matrices - Essential matrix

Starting from the equation $F = [e']_\times P' P^+_\lambda$, if we consider one single moving camera with projection matrix K, and right pose given by displacement matrix R, we use $e' = KR\Omega$, $P' = KR$, and $P^+_\lambda = K^{-1}m$, and then:

$$\begin{aligned} l' &= [KR\Omega]_\times KRK^{-1}m \\ &= (KR)^{-t}[\Omega]_\times K^{-1}m \end{aligned}$$

And so:

$$\boxed{F = (KR)^{-t}[\Omega]_\times K^{-1}}$$

In the calibrated case (i.e. when K is known beforehand), we can use the *essential* matrix, which only depends on the displacement of the camera, and is defined as:

$$\boxed{E = K^t F K = R^{-t}[\Omega]_\times}$$

Fundamental Matrix Summary

For 2 images captured by cameras with distinct optical centres, the fundamental matrix is the unique 3×3 rank 2 matrix F that satisfies $m'^t F m = 0$, for all corresponding pairs of points (m, m').

- **Epipolar lines**: $l' = Fm$ and $l = m'^t F$ are the right and left epipolar lines respectively.
- **Epipoles**: Since $e' \in l'$, we have $\forall m,\ e'^t F m = 0$. Then $e'^t F = 0$. Similarly, $Fe = 0$.
- **Rank**: F is an homogeneous (8 DoF) 3×3 matrix, and has rank 2 ($\det F = 0$), so it actually has 7 DoF.

Estimation of Fundamental Matrix F

Each correspondence $m \leftrightarrow m'$ provides one scalar equation:

$$m'^t F m = 0$$

The developed equation writes:

$$xx'f_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + yy'f_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0$$

Or, by separating data and unknowns:

$$\underbrace{\begin{pmatrix} x'x & x'y & x' & y'x & y'y & y' & x & y & 1 \end{pmatrix}^t}_{\mathbf{d}} \underbrace{\begin{pmatrix} f_{11} & f_{12} & f_{13} & f_{21} & f_{22} & f_{23} & f_{31} & f_{32} & f_{33} \end{pmatrix}}_{\mathbf{f}} = 0$$

And, by using N correspondence pairs $\{m_i \leftrightarrow m'_i\}_{1 \leq i \leq N}$:

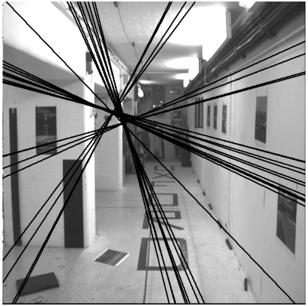
$$\mathbf{Df} = \begin{pmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_N \end{pmatrix} \mathbf{f} = \mathbf{O}_N$$

Estimation of Fundamental Matrix F

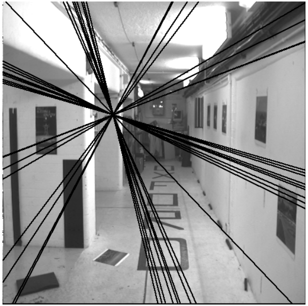
- The system $\mathbf{Df} = \mathbf{O}_N$ is solved using SVD.
- Since the columns of $\mathbf{D}$ range over several order of magnitudes, it is better to normalise the data, for numerical stability purposes.
- Once F is estimated, it is usually imposed that: $e'^t F = 0$, $Fe = 0$, and $\text{rank}(F) = 2$.
    - This is done by finding F' such that $F' = \arg \min_{G;\text{rank}(G)=2} ||F - G||_{\mathrm{F}}$
- RANSAC is used to minimise the number of outliers in the N correspondences $\{m_i \leftrightarrow m'_i\}_{1 \leq i \leq N}$.

Estimation of Fundamental Matrix F - Rank Constraint

- Once F is estimated, it is usually imposed that: $e'^t F = 0$, $Fe = 0$, and $\text{rank}(F) = 2$.
  - This is done by finding F' such that $F' = \arg \min_{G;\text{rank}(G)=2} ||F - G||_{\mathrm{F}}$



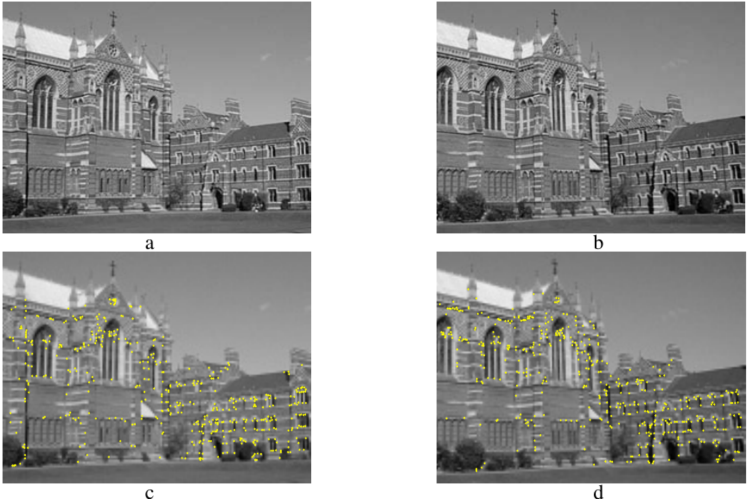
Rank 3



Rank 2

Figure from [Hartley and Zissermann 2003]

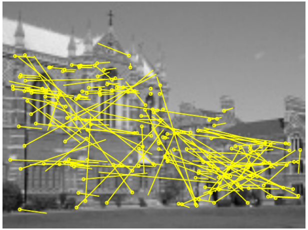
Estimation of Fundamental Matrix F - RANSAC



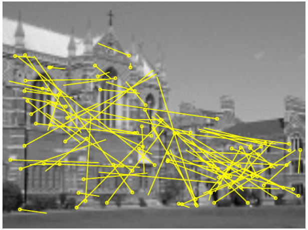
From [Hartley and Zissermann 2003] - There are ≈ 500 keypoints on each image.

Estimation of Fundamental Matrix F - RANSAC

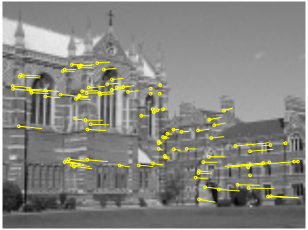
- RANSAC is used to minimise the number of outliers in the N correspondences $\{m_i \leftrightarrow m'_i\}_{1 \leq i \leq N}$.



188 Matches (<< 500!)



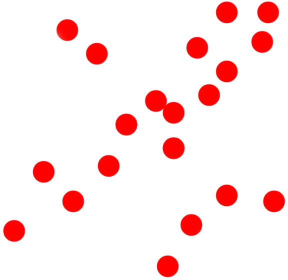
89 Outliers



99 Inliers

Figure from **[Hartley and Zissermann 2003]**

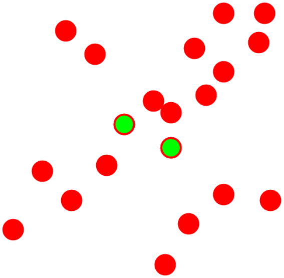
Details on RANSAC: Algorithm



- Randomly sample the number of points required to fit the model
- Solve for model parameters using sample
- Score the model by the ratio of *inliers*
  - $\rightarrow$ points that fit the model up to a certain threshold
- Repeat 1-3 until the best model is found with high confidence

These slides are from Gianni Franchi

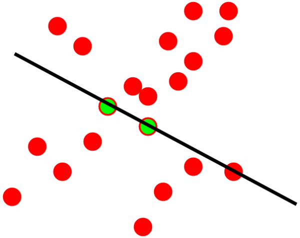
Details on RANSAC: Algorithm



- **Randomly sample the number of points required to fit the model**
- Solve for model parameters using sample
- Score the model by the ratio of *inliers*
  - $\rightarrow$ points that fit the model up to a certain threshold
- Repeat 1-3 until the best model is found with high confidence

These slides are from Gianni Franchi

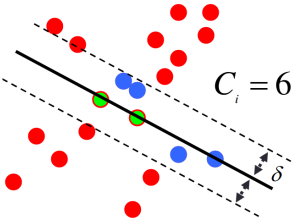
Details on RANSAC: Algorithm



- Randomly sample the number of points required to fit the model
- **Solve for model parameters using sample**
- Score the model by the ratio of *inliers*
  - $\rightarrow$ points that fit the model up to a certain threshold
- Repeat 1-3 until the best model is found with high confidence

These slides are from Gianni Franchi

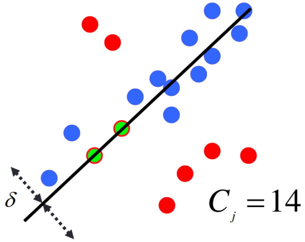
Details on RANSAC: Algorithm



- Randomly sample the number of points required to fit the model
- Solve for model parameters using sample
- **Score the model by the ratio of inliers**
  - **$\rightarrow$ points that fit the model up to a certain threshold**
- Repeat 1-3 until the best model is found with high confidence

These slides are from Gianni Franchi

Details on RANSAC: Algorithm



- Randomly sample the number of points required to fit the model
- Solve for model parameters using sample
- Score the model by the ratio of *inliers*
  - $\rightarrow$ points that fit the model up to a certain threshold
- **Repeat 1-3 until the best model is found with high confidence**

These slides are from Gianni Franchi

Details on RANSAC: Parameters

- The size of the sample set n (minimum size to define the model)
- Error tolerance threshold δ
- Minimum consensus (ratio of inliers w) for a model to be acceptable
- Number of iteration k

The proportion of inliers w has to be empirically defined as well as the tolerance threshold δ.

These slides are from Gianni Franchi

RANSAC: Setting the number of iterations k

Let us write $\mathbf{P(inlier)} = \mathbf{w}$ the probability of choosing an inlier.
Then for a sample set of size n: $\mathbf{P(a\ Subset\ with\ no\ outlier)} = \mathbf{w^n}$.
And then $\mathbf{P(a\ Subset\ with\ outlier(s)\ )} = \mathbf{1 - w^n}$.
So, the probability of choosing a subset with outliers in all k repetitions is :
$\mathbf{P(k\ Subset\ with\ outlier(s)\ )} = \mathbf{(1 - w^n)^k}$.
So the probability of successful run is $\mathbf{P(success)} = \mathbf{1 - (1 - w^n)^k}$.
Finally we get:

$$k = \frac{\log(1 - P(\text{success}))}{\log(1 - w^n)}$$

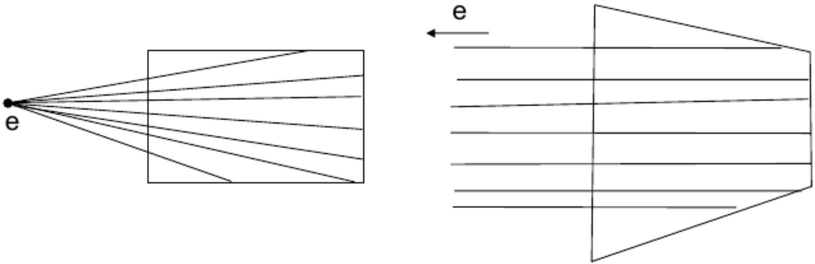
These slides are from Gianni Franchi

RANSAC: number of iteration k with $P(\text{success}) = 0.99$

Sample size n	Proportion of outliers						
	5%	10%	20%	25%	30%	40%	50%
2	**2**	**3**	**5**	**6**	**7**	**11**	**17**
3	**3**	**4**	**7**	**9**	**11**	**19**	**35**
4	**3**	**5**	**9**	**13**	**17**	**34**	**72**
5	**4**	**6**	**12**	**17**	**26**	**57**	**146**
6	**4**	**7**	**16**	**24**	**37**	**97**	**293**
7	**4**	**8**	**20**	**33**	**54**	**163**	**588**
8	**5**	**9**	**26**	**44**	**78**	**272**	**1177**

Presentation Outline

Stereo Rectification

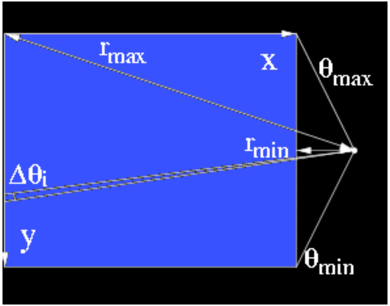


From [Pollefeys 2002]

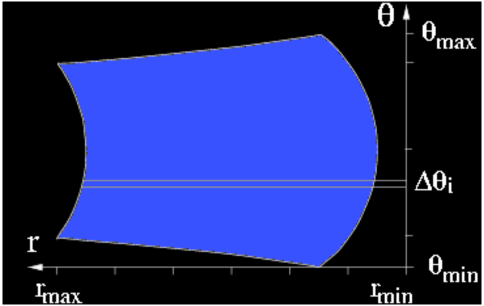
- Objective: come back to the ideal stereo case.
- Find the homography H that makes epipolar lines parallel.
- H transfers the epipole to infinity: $He = \begin{pmatrix} 1 & 0 & 0 \end{pmatrix}^t$.
- Numerical problems when e is close to (or inside!) the image.

Polar Rectification (Pollefeys et al 1999)

Solution: Polar re-parameterization of the two images around their epipoles.



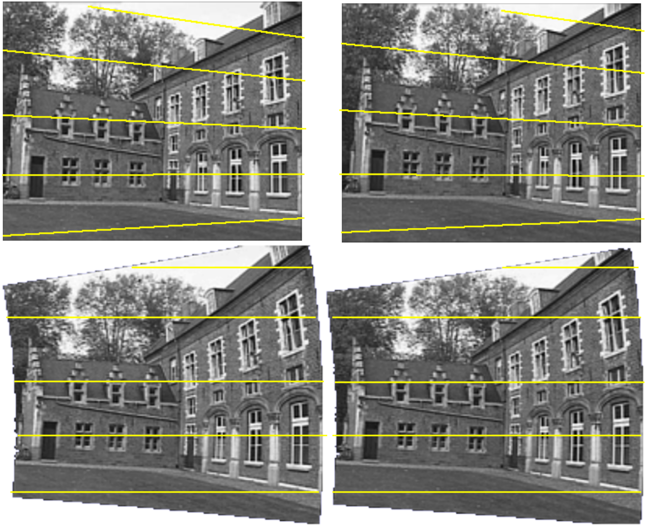
Original



Rectified

From **[Pollefeys 2002]**

Polar Rectification (Pollefeys et al 1999)



From **[Pollefeys 2002]**

Disparity and Depth estimation

- Rectify the two images.
- Compute the dense correspondence between the two images along each epipolar line.
- The horizontal shift between the two images is the disparity.
- The depth is inversely proportional to the disparity.

Left

Disparity
From [Pollefeys 2004]

Right

Epipolar Flow Estimation

Input: Image pair

Keypoint-based sparse flow estimation

- Detector: Blockwise FAST
- Descriptor: 11x11 pixel patch
- Error filtering based on local coherence

Fundamental matrix estimation

- With the 8-points algorithm + RANSAC

Dense optical flow estimation

- Search domain reduced to the epipolar lines
- Propagation of the seed flow vectors coming from the sparse flow estimation

Error filtering

- Make erroneous pixels diverge from epipolar lines
- Filter them according to the epipolar line distance and to local coherence

Small holes filling

- Simple linear interpolation of the disparity to fill small holes caused by error filtering

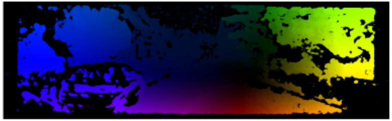
[Garrigues 17]

Epipolar Flow Estimation

[Garrigues 17]:

- Real-Time semi-dense optical flow and relative depth estimation.
- Was ranked #1 on Kitti 2012 Optical Flow dataset (on sparse optical flow category).

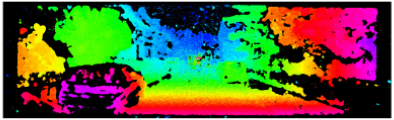
Output 1: optical flow



Output 2: disparity map

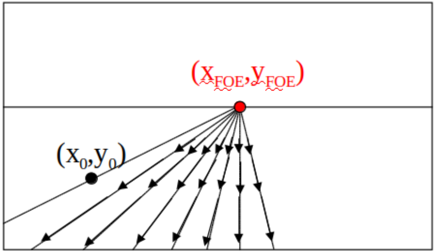
Output 3: relative depth map

(if the camera projection matrix is available)



Conclusion: Limitations of analytical methods

- Estimation strongly relies on local structure (texture), then depth estimation on textureless areas depends on complicated regularization methods.
- Depth calculation depends on the apparent displacement (speed) of a point with respect to the epipole (i.e. the Focus of Expansion FoE, that indicates the translation direction of the camera). Such calculation turns undetermined when the point gets close to the FoE.



DNN for 3d reconstruction

- Like Optical Flow, Depth can benefit from Deep Networks dense prediction capabilities.
- Training can be easily done on *synthetic* or *real RGB-d* data, and loss function is also relatively straightforward.
- One determining benefit of DNN is their ability to exploit potentially *all the depth indices:* parallax, perspective, size and texture gradients, shading,...

...See next lecture on Machine Learning based Depth Estimation!

References

- **[Hartley and Zisserman 2003]** R. Hartley and A. Zisserman
  Multiple View Geometry in Computer Vision
  Cambridge University Press, 2003

- **[Pollefeys 2002]** M. Pollefeys
  Visual 3D Modeling from Images (Tutorial and slides)
  https://www.cs.unc.edu/~marc/tutorial/, 2002

- **[Pollefeys 99]** M. Pollefeys, R. Koch and L. Van Gool
  A simple and efficient rectification method for general motion
  Proc. International Conference on Computer Vision, pp.496-501, 1999.

- **[Garrigues 17]** M. Garrigues and A. Manzanera
  Fast Semi Dense Epipolar Flow Estimation
  IEEE Winter Conf. on Applications of Computer Vision (WACV). Sta Rosa, CA, pp.1-8, 2017