Automatic Estimation of Epipolar Geometry

Problem Statement

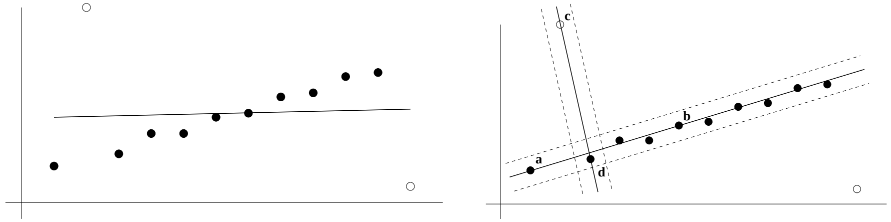
Given Image pair

Find The fundamental matrix $\mathtt{F}$ and correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$.

- Compute image points
- Compute correspondences
- Compute epipolar geometry

Robust line estimation

Fit a line to 2D data containing outliers



There are two problems:

(i) a line fit to the data $\min_{\mathbf{l}} \sum_i d_{\perp i}^2$; and,

(ii) a classification of the data into inliers (valid points) and outliers.

RANdom SAmple Consensus (RANSAC)

[Fischler and Bolles, 1981]

Objective Robust fit of a model to a data set S which contains outliers.

Algorithm

(i) Randomly select a sample of s data points from S and instantiate the model from this subset.

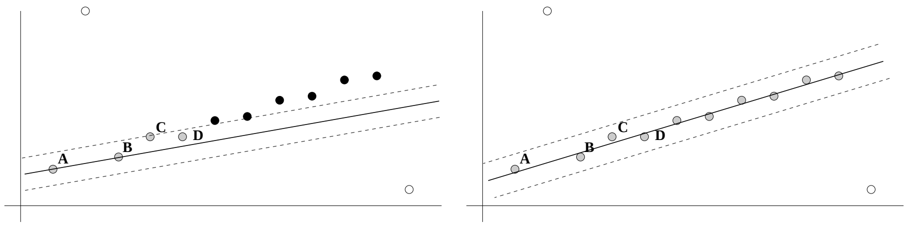
(ii) Determine the set of data points S_i which are within a distance threshold t of the model. The set S_i is the consensus set of the sample and defines the inliers of S.

(iii) If the size of S_i (the number of inliers) is greater than some threshold T, re-estimate the model using all the points in S_i and terminate.

(iv) If the size of S_i is less than T, select a new subset and repeat the above.

(v) After N trials the largest consensus set S_i is selected, and the model is re-estimated using all the points in the subset S_i.

Robust ML estimation



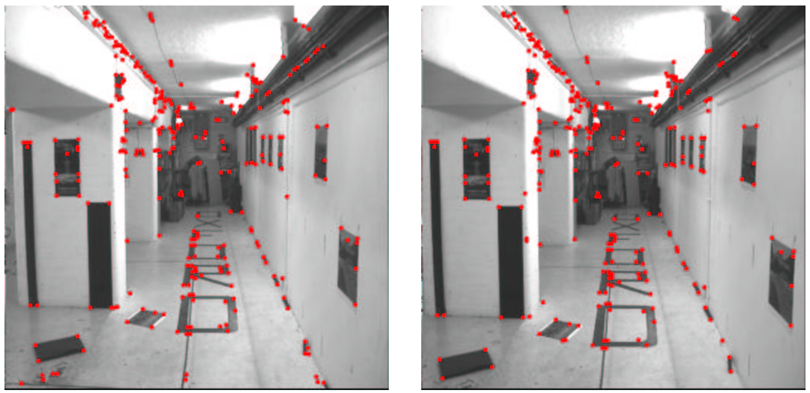
An improved fit by

- A better minimal set
- Robust MLE: instead of $\min_{\mathbf{l}} \sum_i d_{\perp i}^2$

$$\min_{\mathbf{l}} \sum_i \gamma\left(d_{\perp i}\right) \qquad \text{with} \quad \gamma(e) = \begin{cases} e^2 & e^2 < t^2 \text{ inlier} \\ t^2 & e^2 \geq t^2 \text{ outlier} \end{cases}$$

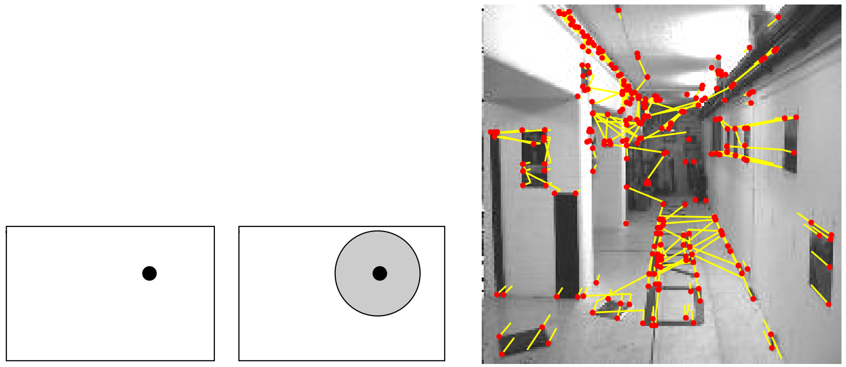
Feature extraction: "Corner detection"

Interest points [Harris]



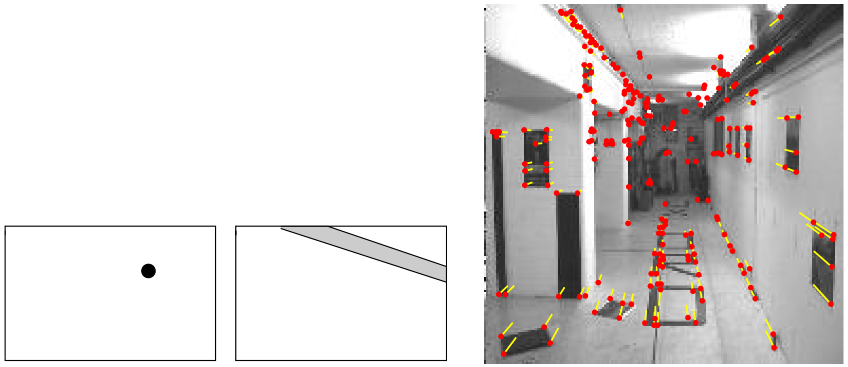
- 100s of points per image

Correlation matching



- Match each corner to most similar looking corner in the other image
- Many wrong matches (10-50%), but enough to compute the fundamental matrix.

Correspondences consistent with epipolar geometry



- Use RANSAC robust estimation algorithm
- Obtain correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ and $\mathtt{F}$
- Guided matching by epipolar line
- Typically: final number of matches is about 200-250, with distance error of $\sim$0.2 pixels.

Automatic Estimation of F and correspondences

Algorithm based on RANSAC [Torr]

(i) Interest points: Compute interest points in each image.

(ii) Putative correspondences: use cross-correlation and proximity.

(iii) RANSAC robust estimation:
Repeat
- (a) Select random sample of 7 correspondences
- (b) Compute F
- (c) Measure support (number of inliers)

Choose the F with the largest number of inliers.

(iv) MLE: re-estimate F from inlier correspondences.

(v) Guided matching: generate additional matches.

Adaptive RANSAC

- $N = \infty$, sample_count$= 0$.
- While $N >$ sample_count Repeat
  - Choose a sample and count the number of inliers.
  - Set $\epsilon = 1 - (\text{number of inliers})/(\text{total number of points})$
  - Set N from ϵ with $p = 0.99$.
  - Increment the sample_count by one.
- Terminate.

e.g. for a sample size of 4

Number of inliers	1 - ϵ	Adaptive N
6	2%	20028244
10	3%	2595658
44	16%	6922
58	21%	2291
73	26%	911
151	56%	43

How many samples?

For probability p of no outliers:

$$N = \log(1-p)/\log(1-(1-\epsilon)^s)$$

- N, number of samples
- s, size of sample set
- ϵ, proportion of outliers

e.g. for $p = 0.95$

Sample size	Proportion of outliers ϵ						
s	5%	10%	20%	25%	30%	40%	50%
2	2	2	3	4	5	7	11
3	2	3	5	6	8	13	23
4	2	3	6	8	11	22	47
5	3	4	8	12	17	38	95
6	3	4	10	16	24	63	191
7	3	5	13	21	35	106	382
8	3	6	17	29	51	177	766

This page left empty

Part 2 : Three-view and Multiple-view Geometry

Computing a Metric Reconstruction

Two View Reconstruction Ambiguity

Given: image point correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$,
compute a reconstruction:

$$\{\mathtt{P},\ \mathtt{P}',\ \mathbf{X}_i\} \quad \text{with} \quad \mathbf{x}_i = \mathtt{P}\mathbf{X}_i \quad \mathbf{x}'_i = \mathtt{P}'\mathbf{X}_i$$

Ambiguity

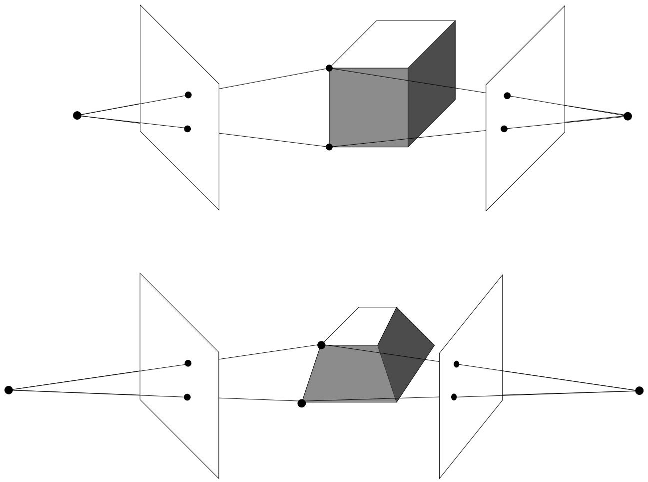
$$\mathbf{x}_i = \mathtt{P}\mathbf{X}_i = \mathtt{P}\ \mathtt{H}(\mathtt{H})^{-1}\ \mathbf{X}_i = \tilde{\mathtt{P}}\tilde{\mathbf{X}}_i$$

$$\mathbf{x}'_i = \mathtt{P}'\mathbf{X}_i = \mathtt{P}'\ \mathtt{H}(\mathtt{H})^{-1}\ \mathbf{X}_i = \tilde{\mathtt{P}}'\tilde{\mathbf{X}}_i$$

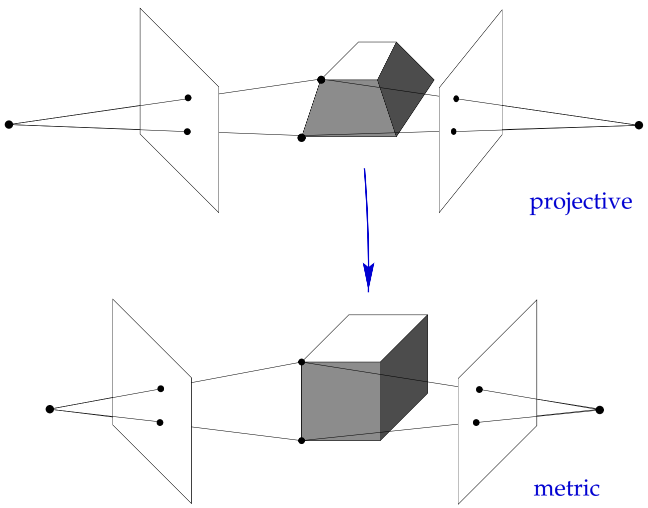
$\{\tilde{\mathtt{P}},\ \tilde{\mathtt{P}}',\ \tilde{\mathbf{X}}_i\}$ is an equivalent Projective Reconstruction.

Reconstruction from two views

Given only image points and their correspondence,
what can be determined?



Metric Reconstruction



Correct: angles, length ratios.

Algebraic Representation of Metric Reconstruction

Compute $\mathtt{H}$

$$\{\mathtt{P}^1,\ \mathtt{P}^2, ..., \mathtt{P}^m,\ \mathbf{X}_i\} \quad \underset{\mathtt{H}}{\rightarrow} \quad \{\mathtt{P}^1_{\mathrm{M}},\ \mathtt{P}^2_{\mathrm{M}}, ..., \mathtt{P}^m_{\mathrm{M}},\ \mathbf{X}^{\mathrm{M}}_i\}$$

Projective Reconstruction   Metric Reconstruction

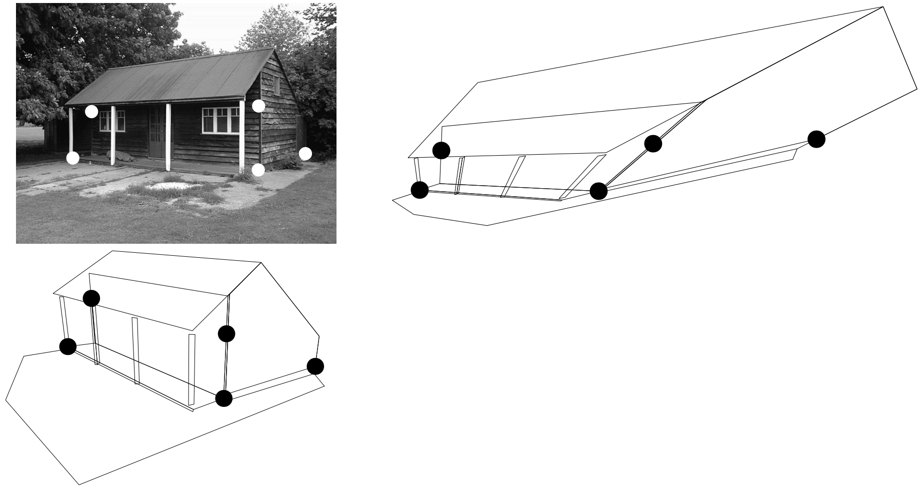
- Remaining ambiguity is rotation (3), translation (3) and scale (1).
- Only 8 parameters required to rectify entire sequence ($15-7=8$).

How?

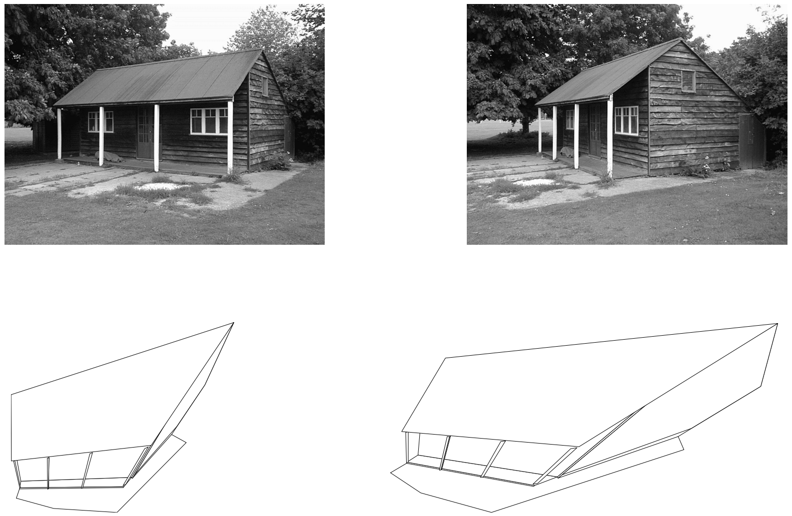
- Calibration points: position of 5 scene points.
- Scene geometry: e.g. parallel lines/planes, orthogonal lines/planes, length ratios.
- Auto-calibration: e.g. camera aspect ratio constant for sequence.

Direct Metric Reconstruction

Use 5 or more 3D points with known Euclidean coordinates to determine $\mathtt{H}$



Projective Reconstruction



Stratified Reconstruction

Given a projective reconstruction $\{\mathtt{P}^j, \mathbf{X}_i\}$, compute a metric reconstruction via an intermediate affine reconstruction.

(i) affine reconstruction: Determine the vector $\mathbf{p}$ which defines $\boldsymbol{\pi}_\infty$. An affine reconstruction is obtained as $\{\mathtt{P}^j\mathtt{H}_{\mathrm{P}}, \mathtt{H}_{\mathrm{P}}^{-1}\mathbf{X}_i\}$ with

$$\mathtt{H}_{\mathrm{P}} = \begin{bmatrix} \mathtt{I} & \mathbf{0} \\ -\mathbf{p}^\top & 1 \end{bmatrix}$$

(ii) Metric reconstruction: is obtained as $\{\mathtt{P}^j_{\mathrm{A}}\mathtt{H}_{\mathrm{A}}, \mathtt{H}_{\mathrm{A}}^{-1}\mathbf{X}_{\mathrm{A}_i}\}$ with

$$\mathtt{H}_{\mathrm{A}} = \begin{bmatrix} \mathtt{K} & \mathbf{0} \\ \mathbf{0}^\top & 1 \end{bmatrix}$$

Stratified Reconstruction

- Start with a projective reconstruction.
- Find transformation to upgrade to affine reconstruction.
  - Equivalent to finding the plane at infinity.
- Find transformation to upgrade to metric (Euclidean) reconstruction.
  - Equivalent to finding the "absolute conic"
- Equivalent to camera calibration
  - If camera calibration is known then metric reconstruction is possible.
  - Metric reconstruction implies knowledge of angles – camera calibration.

Stratified reconstruction ...

(i) Apply the transformations one after the other :

- Projective transformation – reduce to affine ambiguity

$$\begin{bmatrix} \mathtt{I} & \\ \mathbf{v}^\top & 1 \end{bmatrix}$$

- Affine transformation – reduce to metric ambiguity

$$\begin{bmatrix} \mathtt{K} & \\ & 1 \end{bmatrix}$$

- Metric ambiguity of scene remains

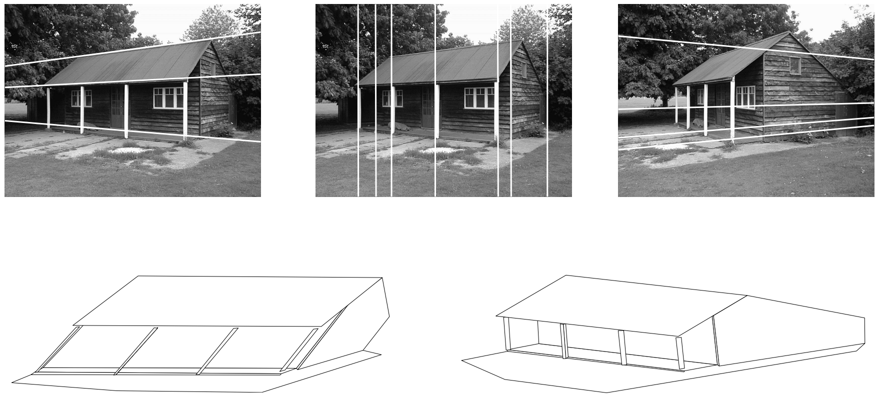
Anatomy of a 3D projective transformation

- General 3D projective transformation represented by a 4×4 matrix.

$$\mathtt{H} = \begin{bmatrix} s\mathtt{RK} & \mathbf{t} \\ \mathbf{v}^\top & 1 \end{bmatrix} = \begin{bmatrix} s\mathtt{R} & \mathbf{t} \\ & 1 \end{bmatrix} \begin{bmatrix} \mathtt{K} & \\ & 1 \end{bmatrix} \begin{bmatrix} \mathtt{I} & \\ \mathbf{v}^\top & 1 \end{bmatrix}$$

$$= \text{metric} \times \text{affine} \times \text{projective}$$

Reduction to affine



Affine reduction using scene constraints - parallel lines

Reduction to affine

Other scene constraints are possible :

- Ratios of distances of points on line (e.g. equally spaced points).
- Ratios of distances on parallel lines.

Points lie in front of the viewing camera.

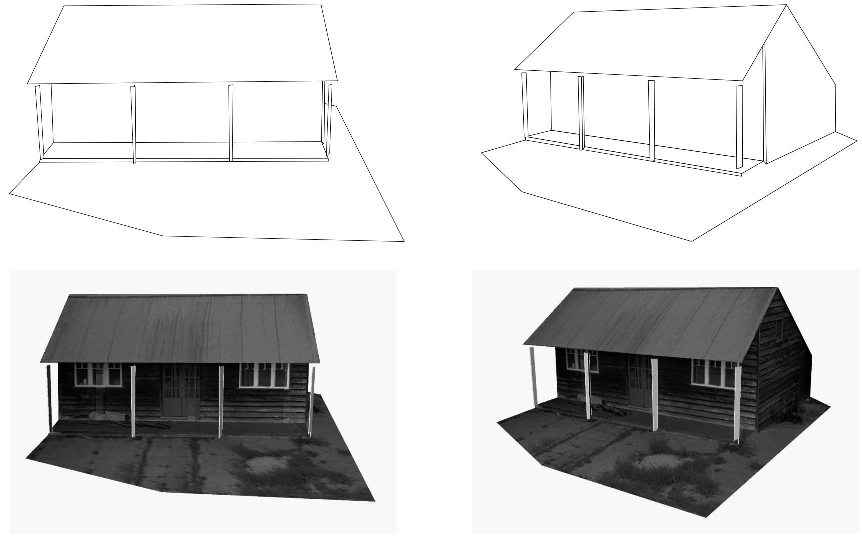
- Constrains the position of the plane at infinity.
- Linear-programming problem can be used to set bounds on the plane at infinity.
- Gives so-called “quasi-affine” reconstruction.
- Reference : Hartley-Azores.

Reduction to affine . . .

Common calibration of cameras.

- With 3 or more views, one can find (in principle) the position of the plane at infinity.
- Iteration over the entries of projective transform : $\begin{bmatrix} \mathtt{I} & \\ \mathbf{v}^\top & 1 \end{bmatrix}$.
- Not always reliable.
- Generally reduction to affine is difficult.

Metric Reconstruction



Metric Reconstruction . . .

Assume plane at infinity is known.

- Wish to make the step to metric reconstruction.
- Apply a transformation of the form $\begin{bmatrix} \mathtt{K} & \\ & 1 \end{bmatrix}$
- Linear solution exists in many cases.

The Absolute Conic

- Absolute conic is an imaginary conic lying on the plane at infinity.
- Defined by

$$\Omega \ : \ \mathrm{X}^2 + \mathrm{Y}^2 + \mathrm{Z}^2 = 0 \ ; \ \mathrm{T} = 0$$

- Contains only imaginary points.
- Determines the Euclidean geometry of the space.
- Represented by matrix $\Omega = \mathrm{diag}(1, 1, 1, 0)$.
- Image of the absolute conic (IAC) under camera $\mathtt{P} = \mathtt{K}[\mathtt{R} \mid \mathbf{t}]$ is given by $\boldsymbol{\omega} = (\mathtt{K}\mathtt{K}^\top)^{-1}$.
- Basic fact :

$\boldsymbol{\omega}$ is unchanged under camera motion.

Example of calibration

Images taken with a non-translating camera:

Using the infinite homography

(i) When a camera moves, the image of a plane undergoes a projective transformation.

(ii) If we have affine reconstruction, we can compute the transformation $\mathtt{H}$ of the plane at infinity between two images.

(iii) Absolute conic lies on the plane at infinity, but is unchanged by this image transformation :

(iv) Transformation rule for dual conic $\boldsymbol{\omega}^* = \boldsymbol{\omega}^{-1}$.

$$\boldsymbol{\omega}^* = \mathtt{H}_j \boldsymbol{\omega}^* \mathtt{H}_j{}^\top$$

(v) Linear equations on the entries of $\boldsymbol{\omega}^*$.

(vi) Given three images, solve for the entries of $\boldsymbol{\omega}^*$.

(vii) Compute $\mathtt{K}$ by Choleski factorization of $\boldsymbol{\omega}^* = \mathtt{K}\mathtt{K}^\top$.

Mosaiced image showing projective transformations

Computation of $\mathtt{K}$

Calibration matrix of camera is found as follows :

- Compute the homographies (2D projective transformations) between images.
- Form equations
$$\boldsymbol{\omega}^* = \mathtt{H}_{ij}\boldsymbol{\omega}^*\mathtt{H}_{ij}{}^\top$$
- Solve for the entries of $\boldsymbol{\omega}^*$
- Choleski factorization of $\boldsymbol{\omega}^* = \mathtt{K}\mathtt{K}^\top$ gives $\mathtt{K}$.

Affine to metric upgrade

Principal is the same for non-stationary cameras once principal plane is known.

- $\mathtt{H}_{ij}$ is the "infinite homography" (i.e. via the plane at infinity) between images i and j.
- May be computed directly from affinely-correct camera matrices.
- Given camera matrices
$$\mathtt{P}_i = [\mathtt{M}_i | \mathbf{t}_i] \quad ; \quad \mathtt{P}_j = [\mathtt{M}_j | \mathbf{t}_j]$$
- Infinite homography is given by
$$\mathtt{H}_{ij} = \mathtt{M}_i\mathtt{M}_j^{-1}$$
- Algorithm proceeds as for fixed cameras.

Changing internal camera parameters

The previous calibration procedure (affine-to-metric) may be generalized to case of changing internal parameters.

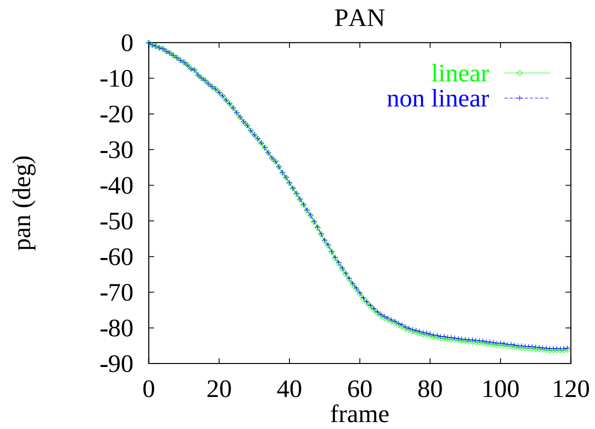
See paper tomorrow given by Agapito.

This page left empty

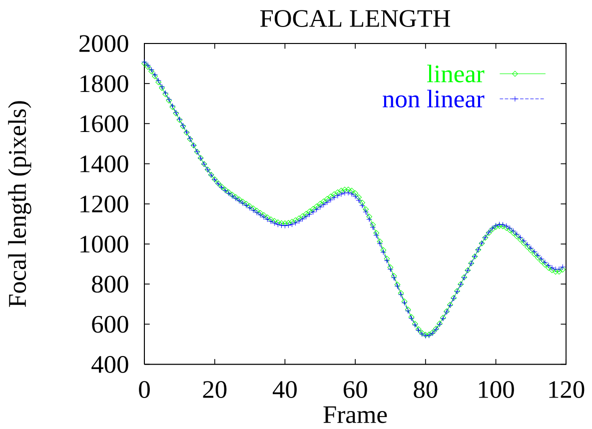
Nice Video

< Video Here >

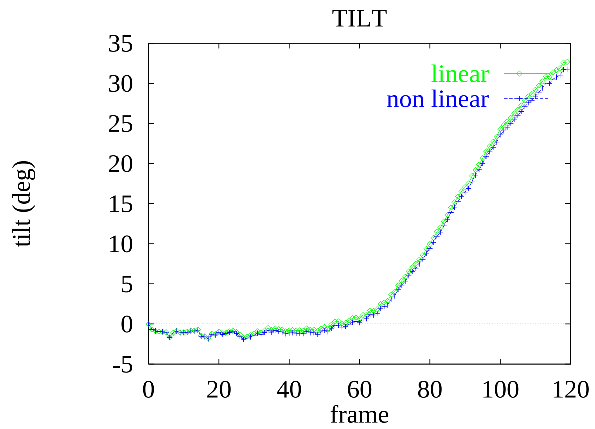
Nice Calibration Results – Pan angle



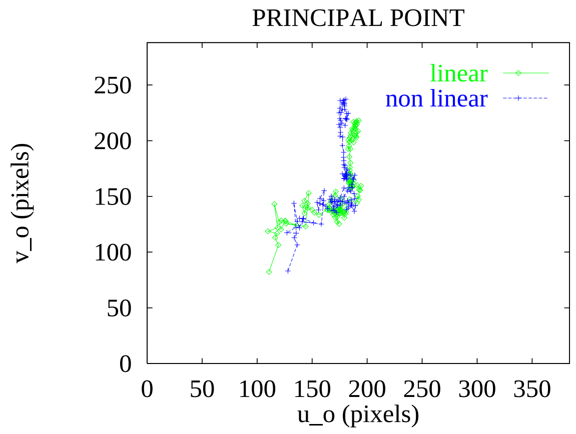
Nice Calibration Results – Focal Length



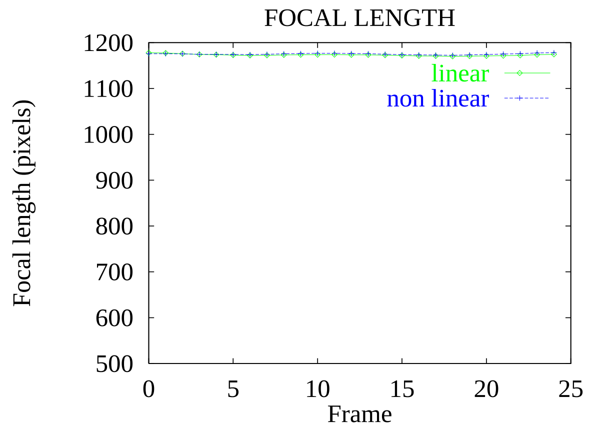
Nice Calibration Results – Tilt angle



Nice Calibration Results – Focal Length



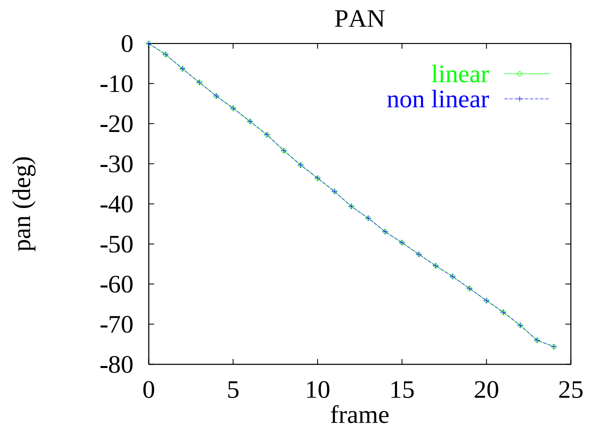
Keble Calibration Results – Focal Length



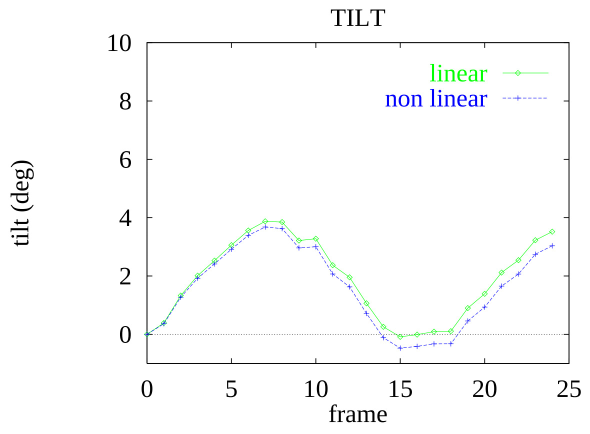
Keble College Video

< Video Here >

Keble Calibration Results – Pan angle

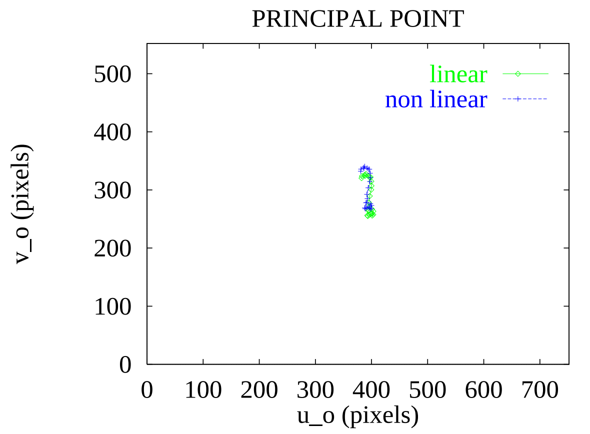


Keble Calibration Results – Tilt angle



This page left empty

Keble Calibration Results – Focal Length



This page left empty

The Trifocal Tensor

The Trifocal Tensor

(i) Defined for three views.

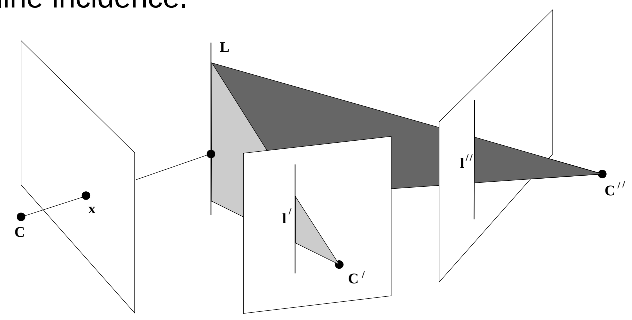
(ii) Plays a similar rôle to Fundmental matrix for two views.

(iii) Unlike fundamental matrix, trifocal tensor also relates lines in three views.

(iv) Mixed combinations of lines and points are also related.

Geometry of three views

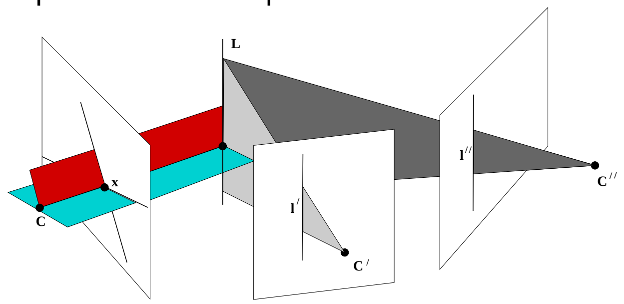
Point-line-line incidence.



- Correspondence $\mathbf{x} \leftrightarrow \mathbf{l}' \leftrightarrow \mathbf{l}''$

Geometry of three views . . .

- Let $\mathbf{l}^{(1)}$ and $\mathbf{l}^{(2)}$ be two lines that meet in $\mathbf{x}$.
- General line $\mathbf{l}$ back-projects to a plane $\mathbf{l}^\top \mathtt{P}$.
- Four plane are $\mathbf{l}^{(1)\top}\mathtt{P}$, $\mathbf{l}^{(2)\top}\mathtt{P}'$, $\mathbf{l}'^\top\mathtt{P}'$ and $\mathbf{l}''^\top\mathtt{P}''$
- The four planes meet in a point.



The trifocal relationship

Four planes meet in a point means determinant is zero.

$$\det \begin{bmatrix} \mathbf{l}^{(1)\top}\mathtt{P} \\ \mathbf{l}^{(2)\top}\mathtt{P} \\ \mathbf{l}'^{\top}\mathtt{P}' \\ \mathbf{l}''^{\top}\mathtt{P}'' \end{bmatrix} = 0$$

- This is a linear relationship in the line coordinates.
- Also (less obviously) linear in the entries of the point $\mathbf{x} = \mathbf{l}^{(1)} \times \mathbf{l}^{(2)}$.

This is the trifocal tensor relationship.

Tensor Notation . . .

Line coordinates

- Line is represented by a vector $\mathbf{l} = (l_1, l_2, l_3)$
- In new coordinate system $\hat{\mathbf{e}}_j$, line has coordinate vector $\hat{\mathbf{l}}$,

$$\hat{\mathbf{l}}^\top = \mathbf{l}^\top \mathtt{H}$$

- Line coordinates transform according to $\mathtt{H}$.
- Preserves incidence relationship. Point lies on line if :

$$\hat{\mathbf{l}}^\top \hat{\mathbf{x}} = (\mathbf{l}^\top \mathtt{H})(\mathtt{H}^{-1}\mathbf{x}) = \mathbf{l}^\top \mathbf{x} = 0$$

- **Terminology :** l_j transforms covariantly.
- Use **lower indices** for *covariant* quantities.

Tensor Notation

Point coordinates.

- Consider basis set $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)$.
- Point is represented by a vector $\mathbf{x} = (x^1, x^2, x^3)^\top$.
- New basis : $\hat{\mathbf{e}}_j = \sum_i \mathtt{H}^i_j \mathbf{e}_i$.
- With respect to new basis $\mathbf{x}$ represented by

$$\hat{\mathbf{x}} = (\hat{x}^1, \hat{x}^2, \hat{x}^3) \text{ where } \hat{\mathbf{x}} = \mathtt{H}^{-1}\mathbf{x}$$

- If basis is transformed according to $\mathtt{H}$, then point coordinates transform according to $\mathtt{H}^{-1}$.
- **Terminology :** x^i transforms contravariantly.
- Use **upper indices** for *contravariant* quantities.

Summation notation

- Repeated index in upper and lower positions implies summation.

Example

Incidence relation is written $l_i x^i = 0$.

Transformation of covariant and contravariant indices

Contravariant transformation

$$\hat{x}^j = (\mathtt{H}^{-1})^j_i x^i$$

Covariant transformation

$$\hat{l}_j = \mathtt{H}^i_j l_i$$

More transformation examples

Camera mapping has one covariant and one contravariant index : P^i_j.
Transformation rule $\hat{\mathtt{P}} = \mathtt{G}^{-1}\mathtt{PF}$ is

$$\hat{P}^j_i = (G^{-1})^j_s \, P^s_r \, F^r_i$$

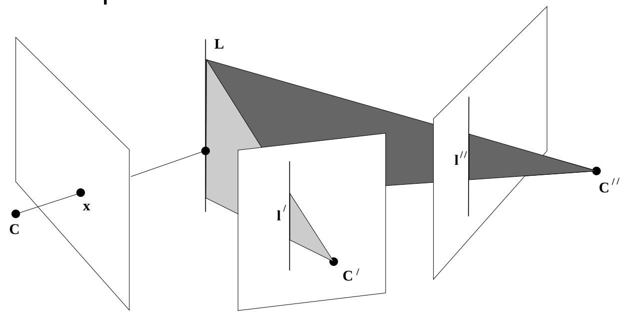
Trifocal tensor T^{jk}_i has one covariant and two contravariant indices.

Transformation rule :

$$\hat{T}^{jk}_i = F^r_i \, (G^{-1})^j_s \, (H^{-1})^k_t T^{st}_r$$

Basic Trifocal constraint

Basic relation is a point-line-line incidence.



- Point x^i in image 1
- lines l'_j and l''_k in images 2 and 3.

The ϵ tensor

Tensor ϵ_{rst} :

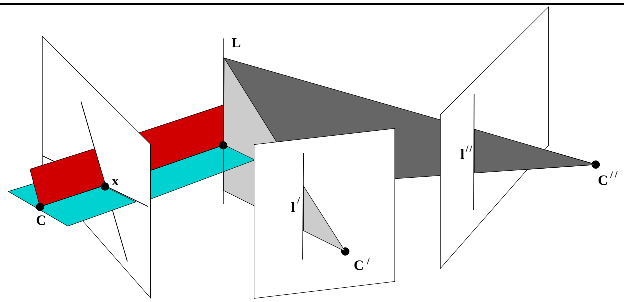
- Defined for $r, s, t = 1, \ldots, 3$

$$\begin{aligned} \epsilon_{rst} &= 0 \quad \text{unless } r,\ s \text{ and } t \text{ are distinct} \\ &= +1 \quad \text{if } rst \text{ is an even permutation of } 123 \\ &= -1 \quad \text{if } rst \text{ is an odd permutation of } 123 \end{aligned}$$

- Related to the cross-product :

$$\mathbf{c} = \mathbf{a} \times \mathbf{b} \quad \Longleftrightarrow c_i = \epsilon_{ijk} a^j b^k \ .$$

Basic Trifocal constraint . . .



- Let $l^{(1)}_r$ and $l^{(2)}_s$ be two lines that meet in point x^i.
- The four lines $l^{(1)}_r$, $l^{(2)}_s$, l'_j and l''_k back-project to planes in space.
- The four planes meet in a point.

Derivation of the basic three-view relationship

- Line l_i back-projects to plane $l_i \mathbf{P}^i$ where $\mathbf{P}^i$ is i-th row.
- Four planes are coincident if

$$(l_r^{(1)} \mathbf{P}^r) \wedge (l_s^{(2)} \mathbf{P}^s) \wedge (l'_j \mathbf{P}'^j) \wedge (l''_k \mathbf{P}''^k) = 0$$

where 4-way wedge $\wedge$ means determinant.

- Thus

$$l_r^{(1)} l_s^{(2)} l'_j l''_k \quad \mathbf{P}^r \wedge \mathbf{P}^s \wedge \mathbf{P}'^j \wedge \mathbf{P}''^k = 0$$

- Multiply by constant $\epsilon^{rsi} \epsilon_{rsi}$ gives

$$\epsilon^{rsi} l_r^{(1)} l_s^{(2)} l'_j l''_k \quad \epsilon_{rsi} \mathbf{P}^r \wedge \mathbf{P}^s \wedge \mathbf{P}'^j \wedge \mathbf{P}''^k = 0$$

- Intersection (cross-product) of $l_r^{(1)}$ and $l_s^{(2)}$ is the point x^i :

$$l_r^{(1)} l_s^{(2)} \epsilon^{rsi} = x^i$$

Definition of the trifocal tensor

Basic relationship is

$$x^i \, l'_j \, l''_k \quad \epsilon_{rsi} \mathbf{P}^r \wedge \mathbf{P}^s \wedge \mathbf{P}'^j \wedge \mathbf{P}''^k = 0$$

Define

$$\epsilon_{rsi} \mathbf{P}^r \wedge \mathbf{P}^s \wedge \mathbf{P}'^j \wedge \mathbf{P}''^k = T_i^{jk}$$

Point-line-line relation is

$$x^i \, l'_j \, l''_k T_i^{jk} = 0$$

T_i^{jk} is covariant in one index (i), contravariant in the other two.

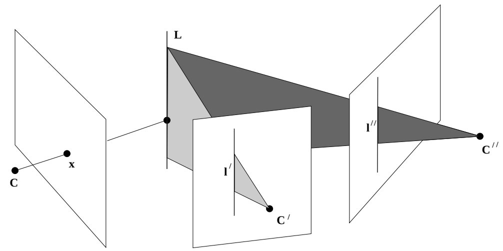
Line Transfer

Basic relation is

$$x^i \, l'_j \, l''_k \, T_i^{jk} = 0$$

Interpretation :

- Back projected ray from $\mathbf{x}$ meets intersection of back-projected planes from $\mathbf{l}'$ and $\mathbf{l}''$.
- Line in space projects to lines $\mathbf{l}'$ and $\mathbf{l}''$ and to a line passing through $\mathbf{x}$.



Line transfer

- Denote $l_i = l'_j \, l''_k T_i^{jk}$
- See that $x^i \, l_i = 0$ when x^i lies on the projection of the intersection of the planes.
- Thus l_i represents the transferred line corresponding to l'_j and l''_k.
- We write

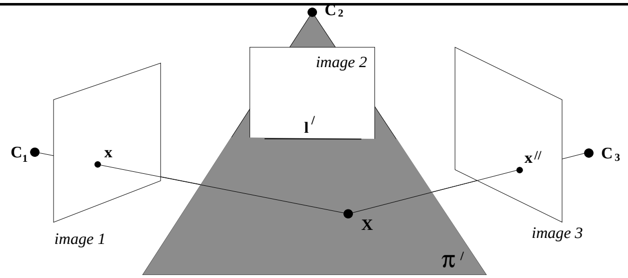
$$l_i \approx l'_j \, l''_k T_i^{jk}$$

- Cross-product of the two sides are equal :

$$l_r \epsilon^{irs} l'_j \, l''_k T_i^{jk} = 0^s$$

- Derived from basic relation $x^i l'_j l''_k T_i^{jk}$ by replacing x^i by $l_r \epsilon^{irs}$.

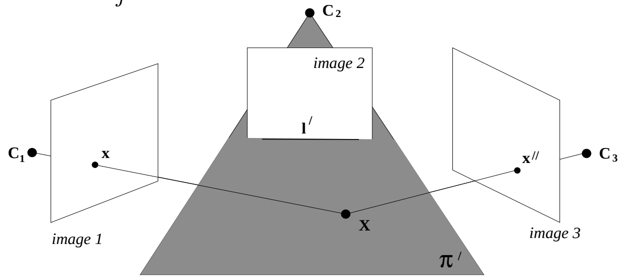
Point transfer via a plane



- Line $\mathbf{l}'$ back-projects to a plane $\boldsymbol{\pi}'$.
- Ray from $\mathbf{x}$ meets $\boldsymbol{\pi}'$ in a point $\mathbf{X}$.
- This point projects to point $\mathbf{x}''$ in the third image.
- For fixed $\mathbf{l}'$, mapping $\mathbf{x} \mapsto \mathbf{x}''$ is a homography.

Contraction of trifocal tensor with a line

- Write $H_i^k = l'_j T_i^{jk}$
- Then $x''^k = H_i^k x^i$.
- H_i^k represents the homography from image 1 to image 3 via the plane of the line l'_j.



Point-transfer and the trifocal tensor

- If $\mathbf{l}''$ is **any** line through $\mathbf{x}''$, then trifocal condition holds.

$$l''_k (x^i \, l'_j \, T_i^{jk}) = 0$$

- $x^i \, l'_j \, T_i^{jk}$ must represent the point x''^k.

$$x''^k \approx x^i \, l'_j \, T_i^{jk}$$

- Alternatively (cross-product of the two sides)

$$x^i \, l'_j \, \epsilon_{krs} x''^r \, T_i^{jk} = 0_s$$

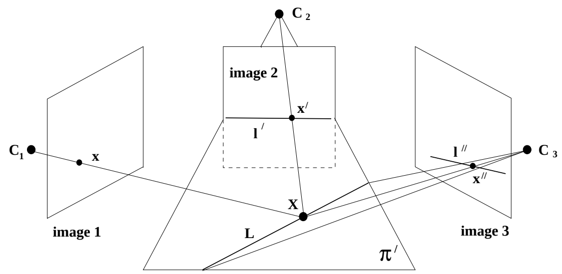
- Derived from basic relation $x^i l'_j l''_k T_i^{jk}$ by replacing l''_k by $x''^r \epsilon_{krs}$.

Three-point correspondence

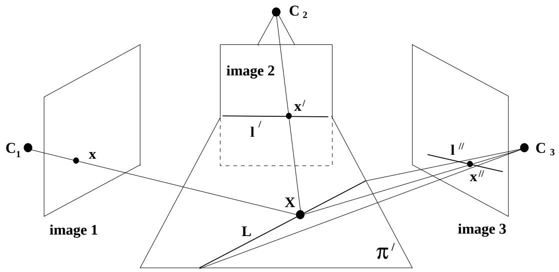
Given a triple correspondence $\mathbf{x} \leftrightarrow \mathbf{x}' \leftrightarrow \mathbf{x}''$

- Choose any lines $\mathbf{l}'$ and $\mathbf{l}''$ passing through $\mathbf{x}'$ and $\mathbf{x}''$
- Trifocal condition holds

$$x^i \, l'_j \, l'_k \, T_i^{jk} = 0$$



Geometry of the three-point correspondence



- 4 choices of lines ⇒ 4 equations.
- May also be written as

$$x^i x'^r \epsilon_{rsi} x''^t \epsilon_{tuj} T_i^{jk} = 0$$

- Gives 9 equations, only 4 linearly independent.

Summary of incidence relations

(i) Point in first view, lines in second and third

$$l'_j \, l''_k \, T_i^{jk} x^i = 0$$

(ii) Point in first view, point in second and line in third

$$x^i x'^j l''_k \, \epsilon_{jpr} \mathcal{T}_i^{pk} = 0_r$$

(iii) Point in first view, line in second and point in third

$$x^i \, l'_j \, x''^k \, \epsilon_{kqs} \mathcal{T}_i^{jq} = 0_s$$

(iv) Point in three views

$$x^i x'^j x''^k \, \epsilon_{jpr} \epsilon_{kqs} \mathcal{T}_i^{pq} = 0_{rs}$$

Summary of transfer formulas

(i) Point transfer from first to third view via a plane in the second.

$$x''^k = x^i \, l'_j \, T_i^{jk}$$

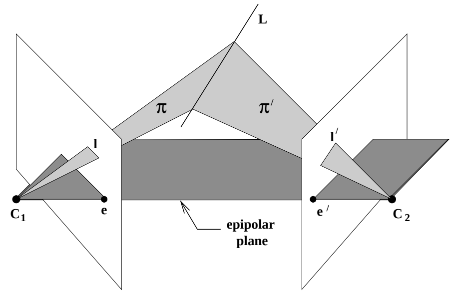
(ii) Point transfer from first to second view via a plane in the third.

$$x'^j = x^i \, l''_k \, T_i^{jk}$$

(iii) Line transfer from third to first view via a plane in the second; or, from second to first view via a plane in the third.

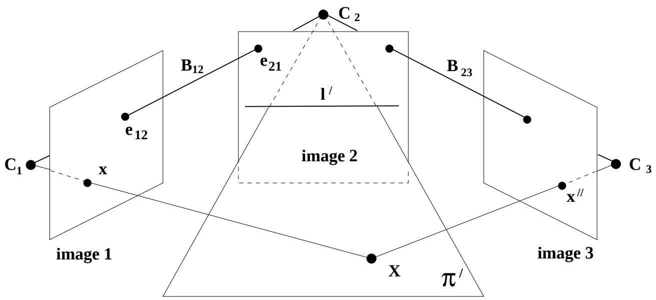
$$l_i = l'_j \, l''_k \, T_i^{jk}$$

Degeneracies of line transfer



- Degeneracy of line transfer for corresponding epipolar lines.
- When the line lies in an epipolar plane, its position can not be inferred from two views.
- Hence it can not be transferred to a third view.

Degeneracy of point transfer



Transferring points from images 1 and 2 to image 3 :

Only points that can not be transferred are those points on the baseline between centres of cameras 1 and 2.

For transfer with fundamental matrix, points in the trifocal plane can not be transferred.

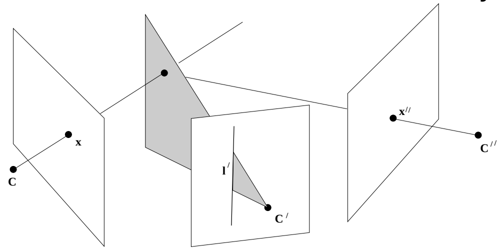
Contraction on a point

$$\text{In } x''_k \approx x^i l'_j T_i^{jk} \text{ write } x^i T_i^{jk} = G^{jk}$$

- Represents a mapping from line l''_k to the point x'^j :

$$x''^k \approx G^{jk} l'_j = (x^i T_i^{jk}) l'_j$$

- As l'_j varies x''^k traces out the projection of the ray through x^i.
- Epipolar line in third image.
- Epipole is the intersection of these lines for varying x^i.



Finding epipolar lines

To find the epipolar line corresponding to a point x^i :

- Transfer to third image via plane back-projected from l'_j

$$x''_k = x^i T_i^{jk} l'_j$$

- Epipolar line satisfies $l''_k x''^k = 0$ for each such x''_k.
- For *all* l'_j

$$x^i \, l'_j \, l''_k \, T_i^{jk} = 0$$

- Epipolar line corresponding to x^i found by solving

$$l''_k \, (x^i T_i^{jk}) = 0^j$$

Result : Epipole is the common perpendicular to the null-space of all $x^i T_i^{jk}$.

This page left empty

Extraction of camera matrices from trifocal tensor

Formula for Trifocal tensor

- Trifocal tensor is independent of projective transformation.
- May assume that first camera is $[\mathtt{I} \mid \mathbf{0}]$
- Other cameras are $[\mathtt{A}|\mathbf{a}_4]$ and $[\mathtt{B}|\mathbf{b}_4]$
- Formula
$$T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$$
- Note : $\mathbf{a}_4$ and $\mathbf{b}_4$ represent the epipoles :
  - Centre of the first camera is $(0, 0, 0, 1)$.
  - Epipole is image of camera centre.

$$\mathbf{a}_4 = [\mathtt{A}|\mathbf{a}_4] \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Where does this formula come from ?

Formula for trifocal tensor

$$\begin{aligned} T_i^{jk} &= \epsilon_{rsi} \mathbf{P}^r \wedge \mathbf{P}^s \wedge \mathbf{P}'^j \wedge \mathbf{P}''^k \\ &= 2 \det \begin{bmatrix} \sim \mathbf{P}^i \\ \mathbf{P}'^j \\ \mathbf{P}''^k \end{bmatrix} \end{aligned}$$

Notation : $\sim \mathbf{P}^i$ means omit row i.

Example, when $i = 1$

$$\begin{aligned} T_1^{jk} &= \det \begin{bmatrix} & 1 & & \\ & & 1 & \\ a_1^j & a_2^j & a_3^j & a_4^j \\ b_1^k & b_2^k & b_3^k & b_4^k \end{bmatrix} \\ &= a_1^j b_4^k - a_4^j b_1^k \end{aligned}$$

Extraction of the camera matrices.

Basic formula
$$T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$$

- Entries of T_i^{jk} are quadratic in the entries of camera matrices.
- **But** if epipoles a_4^j and b_4^k are known, entries are linear.

Strategy :

- Estimate the epipoles.
- Solve linearly for the remaining entries of A and B.
- 27 equations in 18 unknowns.

Exact formulae are possible, but not required for practical computation.

Matrix formulas involving trifocal tensor

Given the trifocal tensor written in matrix notation as $[\mathtt{T}_1, \mathtt{T}_2, \mathtt{T}_3]$.

(i) **Retrieve the epipoles** $\mathbf{e}_{21}, \mathbf{e}_{31}$

Let $\mathbf{u}_i$ and $\mathbf{v}_i$ be the left and right null vectors respectively of $\mathtt{T}_i$, i.e. $\mathtt{T}_i{}^\top \mathbf{u}_i = \mathbf{0}$, $\mathtt{T}_i \mathbf{v}_i = \mathbf{0}$. Then the epipoles are obtained as the null-vectors to the following 3×3 matrices

$$[\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3]\mathbf{e}_{21} = \mathbf{0} \quad [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]\mathbf{e}_{31} = \mathbf{0}$$

(ii) **Retrieve the fundamental matrices** $\mathtt{F}_{12}, \mathtt{F}_{13}$

$$\mathtt{F}_{12} = [\mathbf{e}_{21}]_\times [\mathtt{T}_1, \mathtt{T}_2, \mathtt{T}_3]\mathbf{e}_{31} \quad \mathtt{F}_{13} = [\mathbf{e}_{31}]_\times [\mathtt{T}_1{}^\top, \mathtt{T}_2{}^\top, \mathtt{T}_3{}^\top]\mathbf{e}_{21}$$

(iii) **Retrieve the camera matrices** $\mathtt{P}', \mathtt{P}''$ **(with** $\mathtt{P} = [\mathtt{I} \mid \mathbf{0}]$**)**

Normalize the epipoles to unit norm. Then

$$\mathtt{P}' = [(\mathtt{I} - \mathbf{e}_{21}\mathbf{e}_{21}{}^\top)[\mathtt{T}_1, \mathtt{T}_2, \mathtt{T}_3]\mathbf{e}_{31} \mid \mathbf{e}_{21}] \quad \mathtt{P}'' = [-[\mathtt{T}_1{}^\top, \mathtt{T}_2{}^\top, \mathtt{T}_3{}^\top]\mathbf{e}_{21} \mid \mathbf{e}_{31}]$$

This page left empty

This page left empty

This page left empty

Computation of the trifocal tensor

Summary of relations

Other point or line correspondences also yield constraints on T.

Correspondence	Relation	number of equations
three points	$x^i\, x'^j\, x''^k \epsilon_{jqu}\epsilon_{krv} T_i^{qr} = 0_{uv}$	4
two points, one line	$x^i\, x'^j\, l''_r \epsilon_{jqu} T_i^{qr} = 0_u$	2
one point, two lines	$x^i\, l'_q\, l''_r T_i^{qr} = 0$	1
three lines	$l_p\, l'_q\, l''_r \epsilon^{piw} T_i^{qr} = 0^w$	2

Trilinear Relations

Linear equations for the trifocal tensor

Given a 3-point correspondence

$$\mathbf{x} \leftrightarrow \mathbf{x}' \leftrightarrow \mathbf{x}''$$

The trifocal tensor relationship is

$$x^i\, x'^j\, x''^k \epsilon_{jqu}\epsilon_{krv} T_i^{qr} = 0_{uv}$$

- Relationship is linear in the entries of T.
- each correspondence gives 9 equations, 4 linearly independent.
- T has 27 entries – defined up to scale.
- 7 point correspondences give 28 equations.
- Linear or least-squares solution for the entries of T.

Solving the equations

Given 26 equations we can solve for the 27 entries of T.

- Need 7 point correspondences
- or 13 line correspondences
- or some mixture.

Total set of equations has the form

$$\mathtt{E}\mathbf{t} = \mathbf{0}$$

- With 26 equations find an exact solution.
- With more equations, least-squares solution.

Solving the equations ...

- Solution :
  - Take the SVD : $\mathtt{E} = \mathtt{UDV}^\top$.
  - Solution is the last column of $\mathtt{V}$ corresponding to smallest singular value).
  - Minimizes $||\mathtt{Et}||$ subject to $||\mathtt{t}|| = 1$.
- Normalization of data is *essential*.

What are the constraints

Some of the constraints are easy to find.

(i) Each $T_{\cdot}^{jk}$ must have rank 2.
(ii) Their null spaces must lie in a plane.
(iii) This gives 4 constraints in all.
(iv) 4 other constraints are not so easily formulated.

Constraints

- T has 27 entries, defined only up to scale.
- Geometry only has 18 degrees of freedom.
  - 3 camera matrices account for $3 \times 11 = 33$ dof.
  - Invariant under 3D projective transformation (15 dof).
  - Total of 18 dof.
- T must satisfy several constraints to be a geometrically valid trifocal tensor.
- To get good results, one must take account of these constraints (cf Fundamental matrix case).

Constraints through parametrization.

- Define T in terms of a set of parameters.
- Only valid Ts may be generated from parameters.

Recall formula for T:

$$T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$$

- Only valid trifocal tensors are generated by this formula.
- Parameters are the entries a_i^j and b_i^k.
- Over-parametrized : 24 parameters in all.

Algebraic Estimation of T

Similar to the algebraic method of estimation of $\mathtt{F}$.

Minimize the algebraic error $||\mathtt{E}\mathbf{t}||$ subject to

(i) $||\mathbf{t}|| = 1$
(ii) $\mathbf{t}$ is the vector of entries of T.
(iii) T is of the form $T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$.

Difficulty is this constraint is a quadratic constraint in terms of the parameters.

Minimization knowing the epipoles

Camera matrices $[\mathtt{I} \mid \mathbf{0}]$, $[\mathtt{A}|\mathbf{a}_4]$ and $[\mathtt{B}|\mathbf{b}_4]$.

$$T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$$

As with fundamental matrix, $\mathbf{a}_4$ and $\mathbf{b}_4$ are the epipoles of the first image.

If $\mathbf{a}_4$ and $\mathbf{b}_4$ are known, then T is linear in terms of the other parameters. We may write

$$\mathbf{t} = \mathtt{G}\mathbf{p}$$

- $\mathbf{p}$ is the matrix of 18 remaining entries of camera matrices $\mathtt{A}$ and $\mathtt{B}$.
- $\mathbf{t}$ is the 27-vector of entries of T.
- $\mathtt{G}$ is a 27×18 matrix.

Minimization knowing the epipoles ...

Minimization problem

$$\text{Minimize } ||\mathtt{E}\mathbf{t}|| \text{ subject to } ||\mathbf{t}|| = 1 \ .$$

becomes

$$\text{Minimize } ||\mathtt{E}\mathtt{G}\mathbf{p}|| \text{ subject to } ||\mathtt{G}\mathbf{p}|| = 1 \ .$$

- Exactly the same problem as with the fundamental matrix.
- Linear solution using the SVD.
- Reference : Hartley – Royal Society paper.

Algebraic estimation of T

Complete algebraic estimation algorithm is

(i) Find a solution for T using the normalized linear (7-point) method
(ii) Estimated T will not satisfy constraints.
(iii) Compute the two epipoles $\mathbf{a}_4$ and $\mathbf{b}_4$.
  (a) Find the left (*respectively* right) null spaces of each $T_{\cdot}^{jk}$.
  (b) Epipole is the common perpendicular to the null spaces.
(iv) Reestimate T by algebraic method assuming values for the epipoles.

Iterative Algebraic Method

Find the trifocal tensor T that minimizes $||\mathtt{E}\mathbf{t}||$ subject to $||\mathbf{t}|| = 1$ and $T_i^{jk} = a_i^j b_4^k - a_4^j b_i^k$.

- **Concept :** Vary epipoles $\mathbf{a}_4$ and $\mathbf{b}_4$ to minimize the algebraic error $||\mathtt{E}\mathbf{t}'|| = ||\mathtt{E}\mathtt{G}\mathbf{p}||$.
- **Remark :** Each choice of epipoles $\mathbf{a}_4$ and $\mathbf{b}_4$ defines a minimimum error vector $\mathtt{E}\mathtt{G}\mathbf{p}$ as above.
- Use Levenberg-Marquardt method to minimize this error.
- Simple 6×27 minimization problem.
  - 6 inputs – the entries of the two epipoles
  - 27 outputs – the algebraic error vector $\mathtt{E}\mathbf{t}' = \mathtt{E}\mathtt{G}\mathbf{p}$.
- Each step requires estimation of $\mathbf{p}$ using algebraic method.

This page left empty

This page left empty

This page left empty

Automatic Estimation of a Projective Reconstruction for a Sequence

Outline

first frame of video

(i) Projective reconstruction: 2-views, 3-views, N-views

(ii) Obtaining correspondences over N-views

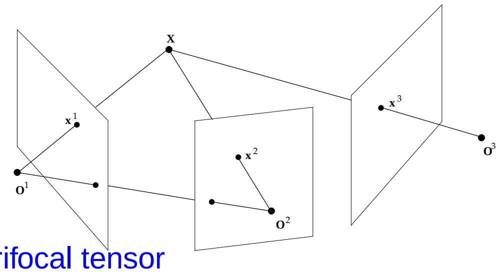
Reconstruction from three views

Given: image point correspondences $\mathbf{x}_i^1 \leftrightarrow \mathbf{x}_i^2 \leftrightarrow \mathbf{x}_i^3$,
compute a projective reconstruction:

$$\{\mathtt{P}^1,\ \mathtt{P}^2,\ \mathtt{P}^3,\ \mathbf{X}_i\} \quad \text{with} \quad \mathbf{x}_i^j = \mathtt{P}^j\,\mathbf{X}_i$$

What is new?

verify correspondences



- 3-view tensor: the trifocal tensor
- Compute from 6 image point correspondences.
- Automatic algorithm similar to $\mathtt{F}$. [Torr & Zisserman]

Automatic Estimation of the trifocal tensor and correspondences

(i) Pairwise matches: Compute point matches between view pairs using robust $\mathtt{F}$ estimation.

(ii) Putative correspondences: over three views from two view matches.

(iii) RANSAC robust estimation:

Repeat

(*a*) Select random sample of 6 correspondences

(*b*) Compute T (1 or 3 solutions)

(*c*) Measure support (number of inliers)

Choose the T with the largest number of inliers.

(iv) MLE: re-estimate T from inlier correspondences.

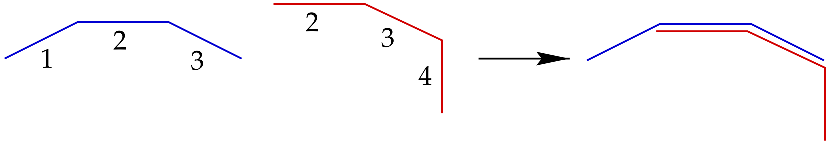
(v) Guided matching: generate additional matches.

Projective Reconstruction for a Sequence

(i) Compute all 2-view reconstructions for consecutive frames.

(ii) Compute all 3-view reconstructions for consecutive frames.

(iii) Extend to sequence by hierarchical merging:



(iv) Bundle-adjustment: minimize reprojection error

$$\min_{\mathtt{P}^j\,\mathbf{X}_i} \sum_{i \in \text{points}} \sum_{j \in \text{frames}} d\left(\mathbf{x}_i^j,\ \mathtt{P}^j \mathbf{X}_i\right)^2$$

(v) Automatic algorithm [Fitzgibbon & Zisserman]

Cameras and Scene Geometry for an Image Sequence

Given video

first frame of video

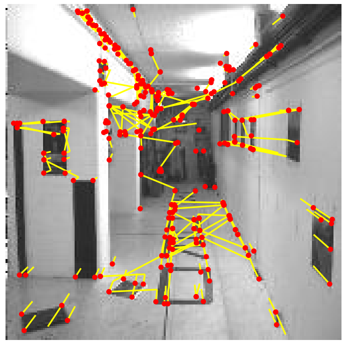
- Point correspondence (tracking).
- Projective Reconstruction.

Interest points computed for each frame

first frame of video

- About 500 points per frame

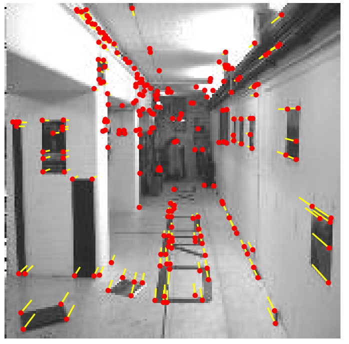
Point tracking: Correlation matching



first frame of video

- 10-50% wrong matches

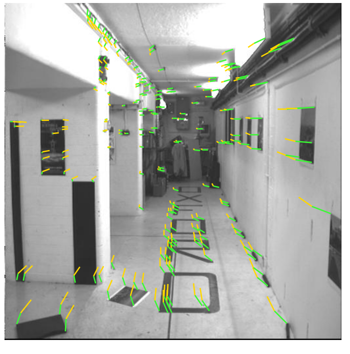
Point tracking: Epipolar-geometry guided matching



first frame of video

- Compute F so that matches consistent with epipolar geometry.
- Many fewer false matches, but still a loose constraint.

Point tracking: Trifocal tensor guided matching



first frame of video

- Compute trifocal tensor so that matches consistent with 3-views.
- Tighter constraint, so even fewer false matches.
- Three views is the last significant improvement.

Reconstruction from Point Tracks

Compute 3D points and cameras from point tracks

a frame of the video

- Hierarchical merging of sub-sequences.
- Bundle adjustment.

Application I: Graphical Models

Compute VRML piecewise planar model

Example II: Extended Sequence

140 frames of a 340 frame sequence

a frame of the video

Application II: Augmented Reality

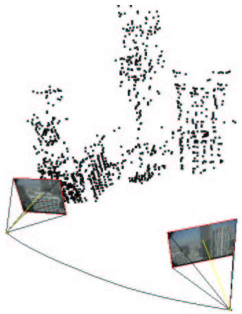
Using computed cameras and scene geometry,
insert virtual objects into the sequence.

a frame of the video

330 frames

Metric Reconstruction

140 frames of a 340 frame sequence



a frame of the video

3D Insertion

a frame of the video

Further Reading

Some of these papers are available from http://www.robots.ox.ac.uk/∼vgg

- Beardsley, P., Torr, P. H. S., and Zisserman, A. 3D model acquisition from extended image sequences. In B. Buxton and Cipolla R., editors, *Proc. 4th European Conference on Computer Vision, LNCS 1065, Cambridge*, pages 683–695. Springer–Verlag, 1996.
- Deriche R., Zhang Z., Luong Q. T., and Faugeras O., Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In J. O. Eckland, editor, *Proc. 3rd European Conference on Computer Vision, LNCS 800/801, Stockholm*, pages 567–576. Springer-Verlag, 1994.
- Faugeras, O. "What can be seen in three dimensions with an uncalibrated stereo rig?", *Proc. ECCV*, LNCS 588. Springer-Verlag, 1992.
- Faugeras, O., "Three-Dimensional Computer Vision", MIT Press, 1993.
- Faugeras, O. "Stratification of three-dimensional vision: projective, affine, and metric representation", *J. Opt. Soc. Am.*, A12:465–484, 1995.
- Fischler, M. A. and Bolles, R. C., "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography", *CommACM*, 24, 6, 381–395, 1981.
- Fitzgibbon, A and Zisserman, A, Automatic camera recovery for closed or open image sequences. In *Proc. ECCV*, pages 311–326. Springer-Verlag, June 1998.
- Hartley, R., Gupta, R., and Chang, T. "Stereo from uncalibrated cameras", *Proc. CVPR*, 1992.
- Hartley, R. I., In defence of the 8-point algorithm. In *ICCV5*, 1995.
- Hartley, R. I., A linear method for reconstruction from lines and points. In *Proc. ICCV*, pages 882–887, 1995.
- Hartley, R. I. and Zisserman, A., *Multiple View Geometry in Computer Vision*, Cambridge University Press, to appear.
- Kanatani, K., *Geometric Computation for Machine Vision*. Oxford University Press, Oxford, 1992.
- Longuet-Higgins H. C., A computer algorithm for reconstructing a scene from two projections. *Nature*, vol.293:133–135, 1981.
- Luong, Q. T. and Vieville, T., Canonical representations for the geometries of multiple projective views. *CVIU*, 64(2):193–229, September 1996.
- Mundy, J. and Zisserman, A., "Geometric invariance in computer vision", MIT Press, 1992. Introduction and Chapter 23 (projective geometry).
- Semple, J. and Kneebone, G. *Algebraic Projective Geometry*. Oxford University Press, 1979.
- Shashua, A. Trilinearity in visual recognition by alignment. In *Proc. ECCV*, volume 1, pages 479–484, May 1994.
- Spetsakis, M. E. and Aloimonos, J, Structure from motion using line correspondences. *IJCV*, 4(3):171–183, 1990.
- Torr P. H. S., *Outlier Detection and Motion Segmentation*. PhD thesis, University of Oxford, Engineering Dept., 1995.
- Torr, P. H. S. and Zisserman, A, Robust parameterization and computation of the trifocal tensor. *Image and Vision Computing*, 15:591–605, 1997.

Acknowledgements

The images in this document were created by Paul Beardsley, Antonio Criminisi, Andrew Fitzgibbon, David Liebowitz, Luc Robert, and Phil Torr.