Projective Geometry for Computer Vision

Raquel A. Romano

MIT Artificial Intelligence Laboratory

romano@ai.mit.edu

3D Computer Vision

Classical Problem:

Given a collection of 2D images, build a model of the 3D world.

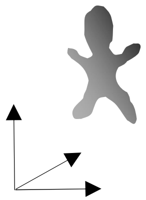
Example Applications:

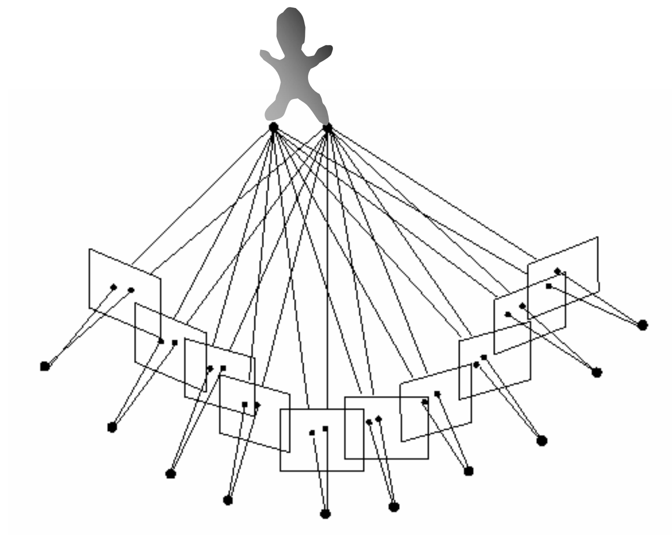
- virtual/immersive environments
- robotics & autonomous vehicles
- minimally invasive surgery

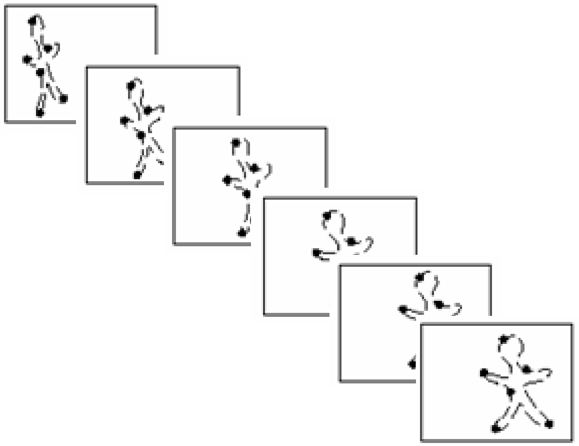
Outline

1. Projective Geometry Overview
2. Minimal Projective Parameters
3. Projective Parameter Estimation
4. Motion Boundary Detection
5. Conclusion

Image Formation

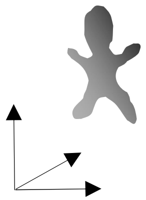


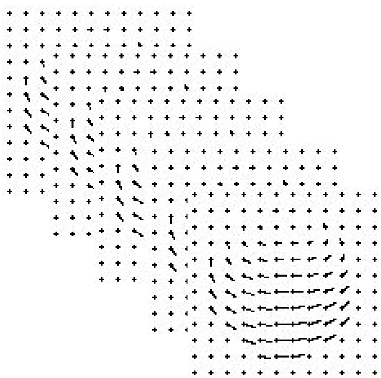


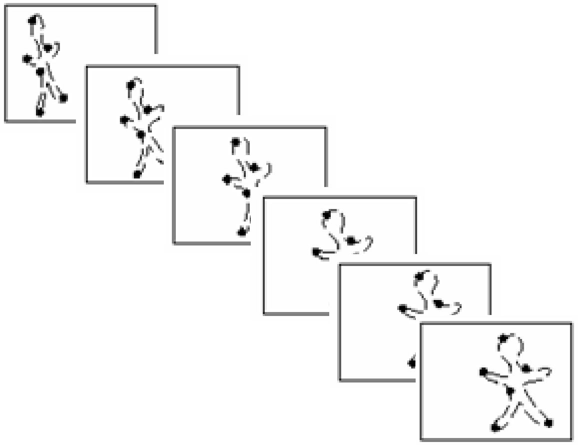


3D scene → imaging → 2D images

Computer Vision







3D scene model

data

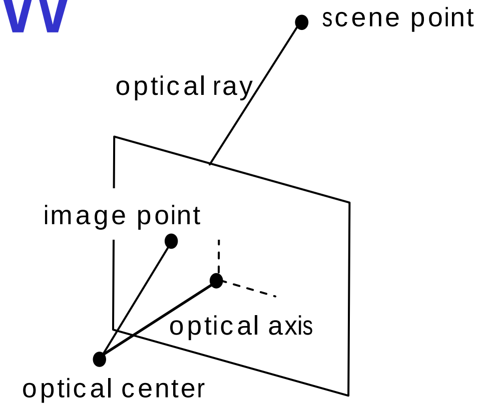
2D images

analysis

measurement

Camera Geometry: Single View

pinhole model of perspective projection



unknown depth at each point

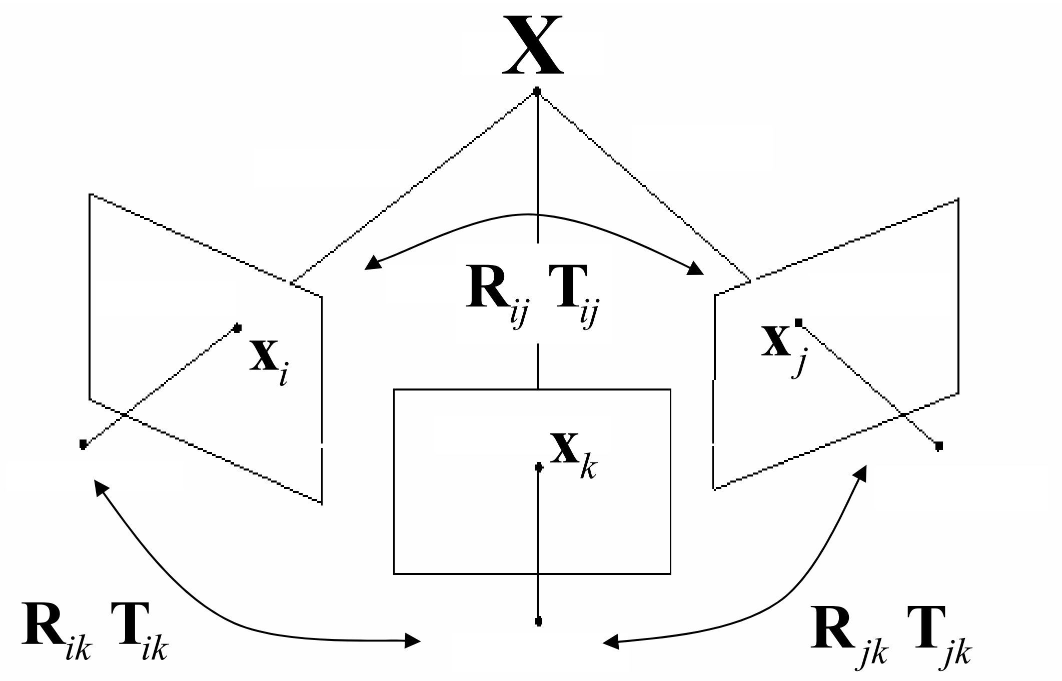
$$x = \frac{X}{Z} \qquad y = \frac{Y}{Z}$$

unknown internal camera parameters

$$\begin{bmatrix} x \\ y \end{bmatrix} \rightarrow \begin{bmatrix} f_x & s \\ 1 & f_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_x \\ c_y \end{bmatrix}$$

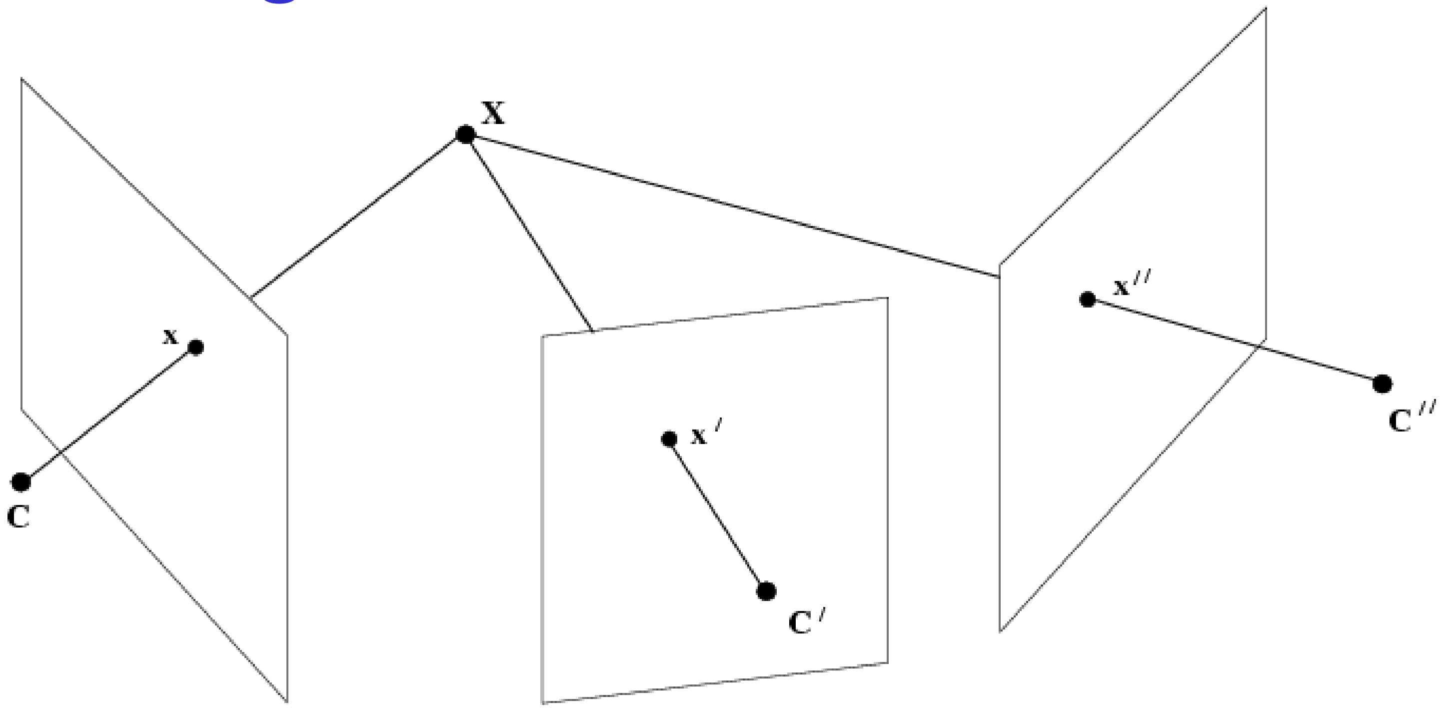
Camera Geometry: Multiple Views

unknown rotations and translations



$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \rightarrow \mathbf{R} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \mathbf{T}$$

Measured Data: Image Points and Lines



geometric constraint: optical rays intersect in 3D

projective geometry: express constraint in terms of measured 2D image features

Projective Camera Model

- linear model of image formation
- depth-independent expression for optical ray intersections
- multilinear relations among point and line matches

Bilinear Constraints

(Longuet-Higgins ,1981, Faugeras, 1992; Hartley, 1992)

$$\mathbf{X} = \lambda_i \mathbf{x}_i$$

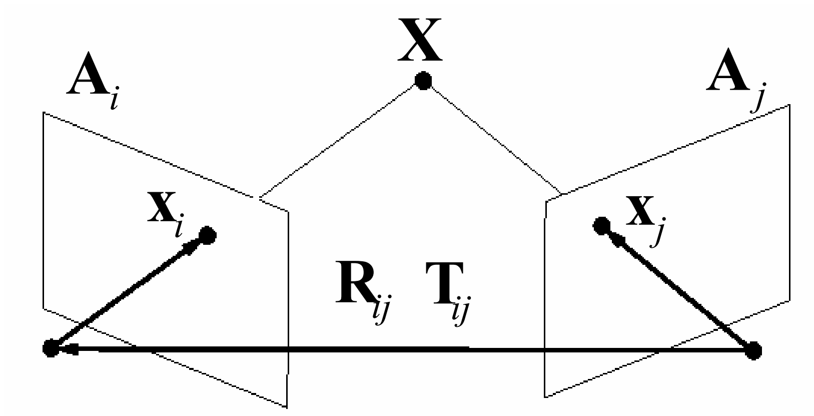
$$\lambda_j \mathbf{x}_j = \lambda_i \mathbf{R}_{ij} \mathbf{x}_i + \mathbf{T}_{ij}$$

$$\mathbf{x}_j^T [\mathbf{T}_{ij}]_\times \mathbf{R}_{ij} \mathbf{x}_i = 0$$

$$\mathbf{x}_i \rightarrow \mathbf{A}_i^{-1} \mathbf{x}_i$$

$$\mathbf{x}_j \rightarrow \mathbf{A}_j^{-1} \mathbf{x}_j$$

$$\mathbf{x}_j^T \underbrace{\mathbf{A}_j^{-T} [\mathbf{T}_{ij}]_\times \mathbf{R}_{ij} \mathbf{A}_i^{-1}}_{\mathbf{F}_{ij}} \mathbf{x}_i = 0$$

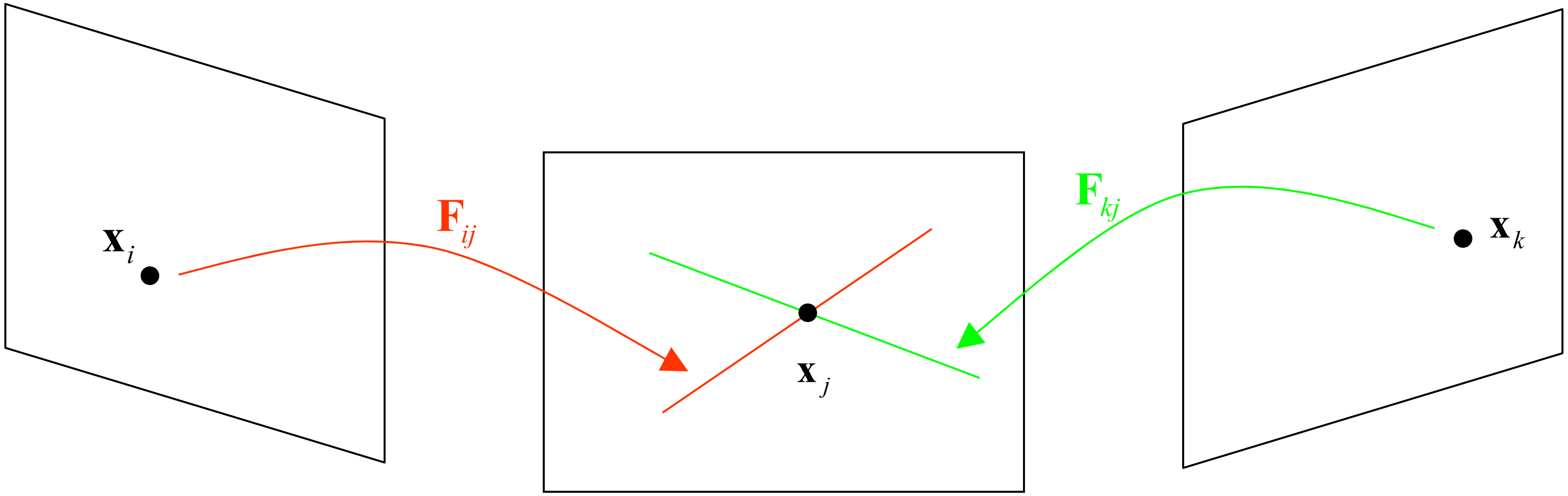


$$\mathbf{x}_j^T \mathbf{F}_{ij} \mathbf{x}_i = 0$$

fundamental matrix

Fundamental Matrix

Maps a point in one image to a line in the other image that contains its match



Given matching points in two views, predict the matching point in a third image.

Projective Models in Practice

- View synthesis and interpolation: point transfer function for dense point correspondences
- Self-calibration: automatic recovery of internal camera parameters from fundamental matrices
- Bundle adjustment initialization: initial rotation and translation for nonlinear Euclidean optimization

Outline

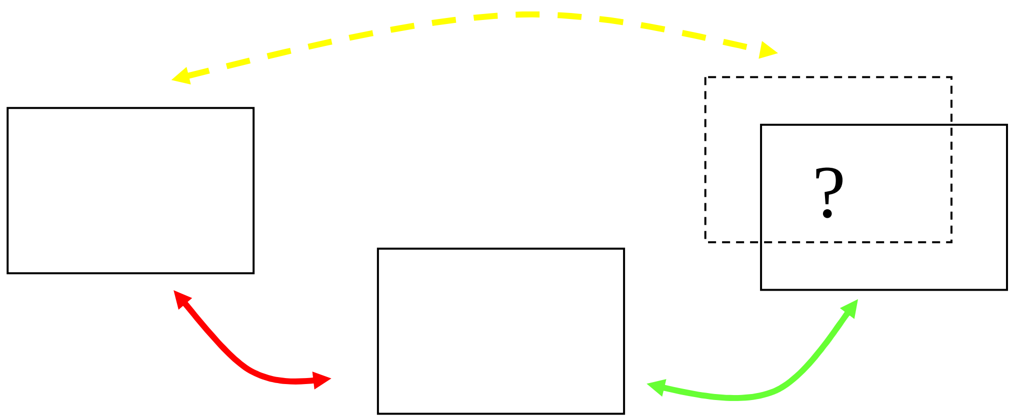
1. Projective Geometry Overview
2. Minimal Projective Parameters
3. Projective Parameter Estimation
4. Motion Boundary Detection
5. Conclusion

Practical Problem

- Few point matches between some views.
- Unstable for estimating geometric relationships.

Geometric Consistency

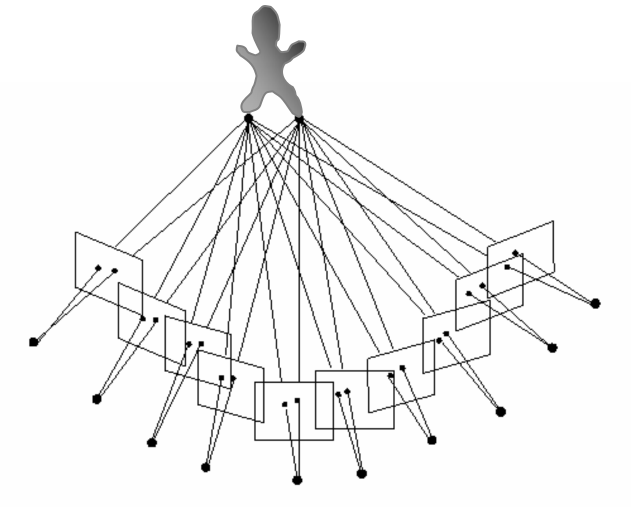
Pairwise geometric relations may be inconsistent.



Goals

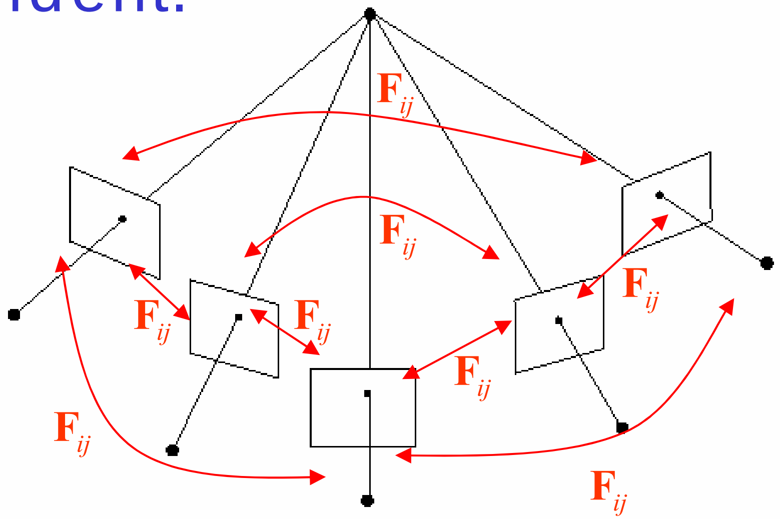
- Impose algebraic geometric constraints on stationary points seen in arbitrarily many views.

- Avoid estimating too many parameters: depths, rotations, translations



Geometric Dependencies

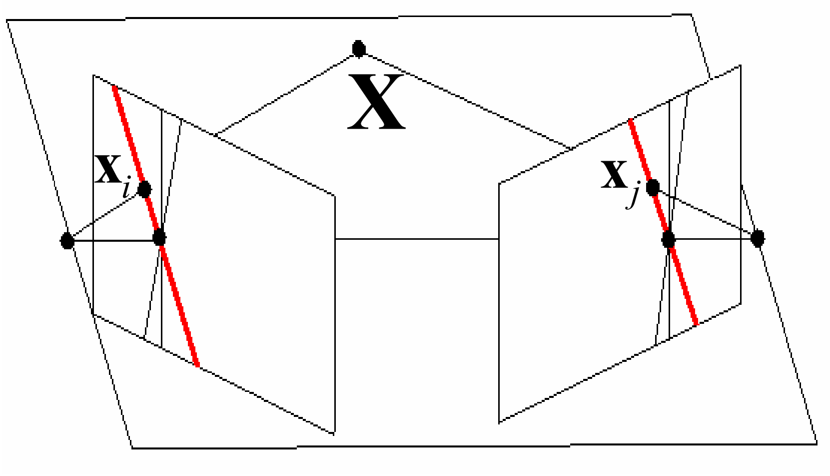
- Pairwise projective geometric relations are interdependent.



- Approach: define projective dependencies and restrict solutions to be globally consistent

Projective Bilinear Parameters

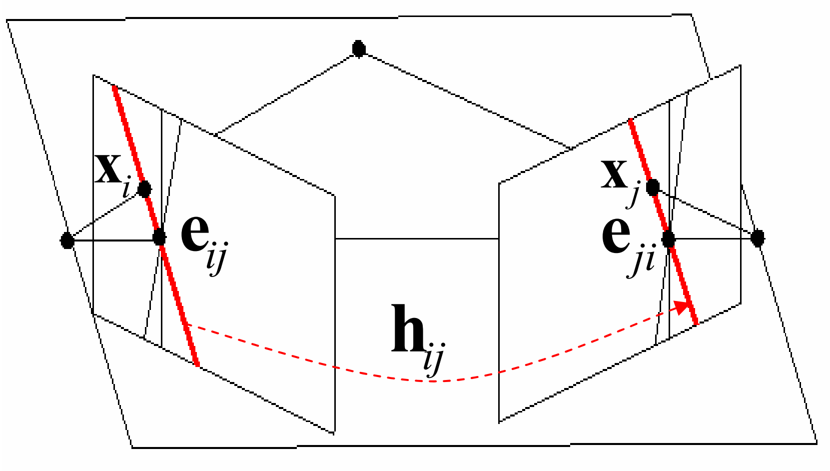
$$\mathbf{x}_j^T \mathbf{F}_{ij} \mathbf{x}_i = 0$$



$$\mathbf{F}_{ij} = \mathbf{A}_j^{-T} \left[\mathbf{T}_{ij}\right]_{\times} \mathbf{R}_{ij} \mathbf{A}_i^{-1}$$

Projective Bilinear Parameters

$$\mathbf{x}_j^T \mathbf{F}_{ij} \mathbf{x}_i = 0$$



epipoles

$\mathbf{e}_{ij}$ $\mathbf{e}_{ji}$

epipolar collineation

$\mathbf{h}_{ij}$

$$\mathbf{F}_{ij} \cong [\mathbf{e}_{ji}]_\times [\mathbf{p}_j \quad \mathbf{q}_j] \mathbf{h}_{ij} \begin{bmatrix} \mathbf{q}_i^T \\ -\mathbf{p}_i^T \end{bmatrix} [\mathbf{e}_{ij}]_\times$$

imaged 3D translation & rotation

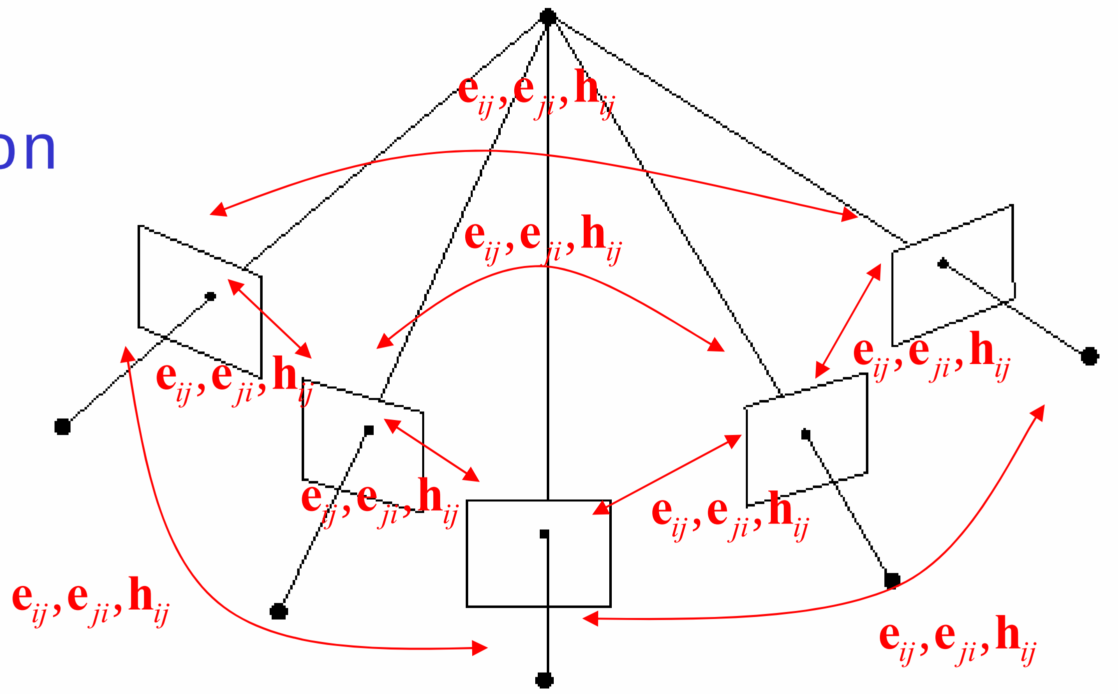
(Csurka, et.al., 1997)

Projective Parameters

- provide a complete projective model of camera configuration

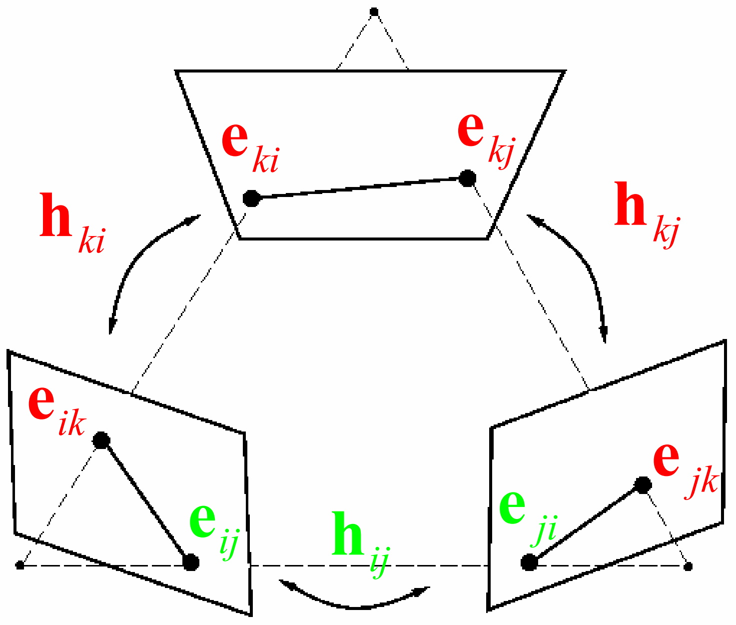
But...

- set of all pairwise parameters are still redundant
- not all images have sufficient overlap



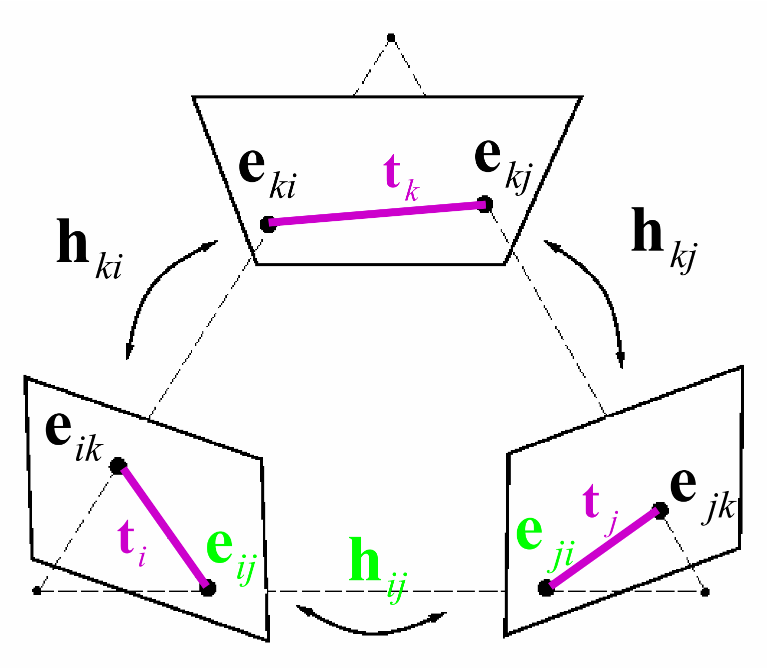
Trifocal Dependencies

- derive dependencies among three fundamental matrices
- correctly models degrees of freedom in camera configuration
- geometrically consistent parameterized model of view triplets



Trifocal Dependencies

- derive dependencies among three fundamental matrices
- correctly models degrees of freedom in camera configuration
- geometrically consistent parameterized model of view triplets



trifocal lines available from two fundamental matrices

Outline

1. Projective Geometry Overview
2. Minimal Projective Parameters
3. Projective Parameter Estimation
4. Motion Boundary Detection
5. Conclusion

Recovering Camera Geometry

view i

view k

view j

← few correspondences →

Linear Initialization

8-point Algorithm

(Hartley, 1995)

Minimize $\sum_{\{(\mathbf{x}_i, \mathbf{x}_j)\}} \left(\mathbf{x}_j^T \mathbf{F}_{ij} \mathbf{x}_i \right)$ over all matching point pairs.

Rewrite bilinear constraints as

$$\begin{bmatrix} x_i x_j & y_i x_j & x_j & x_i y_j & y_i y_j & y_j & x_j & y_j & 1 \end{bmatrix} \mathbf{f}_{ij} = \mathbf{0}$$

where

$$\mathbf{f}_{ij} = \begin{bmatrix} f_{11} & f_{12} & f_{13} & f_{21} & f_{22} & f_{23} & f_{31} & f_{32} & f_{33} \end{bmatrix}^T$$

and solve linear system

$$\mathbf{A}\mathbf{f}_{ij} = \mathbf{0}$$

Projection to Parameter Space

Map linear estimate of fundamental matrix
to projective parameter space:

$$\mathbf{F}_{ij} \rightarrow \mathbf{p}_7^{ij} = \{\mathbf{e}_{ij}, \mathbf{e}_{ji}, \mathbf{h}_{ij}\} \rightarrow \mathbf{p}_4^{ij} = \{\boldsymbol{\gamma}_i, \boldsymbol{\gamma}_j, \mathbf{h}_{ij}\}$$

- parameterization requires choice of projective basis
- basis affects shape of error surface for nonlinear optimization

Geometric Objective Function

point-to-epipolar-line distance ~ image reprojection error

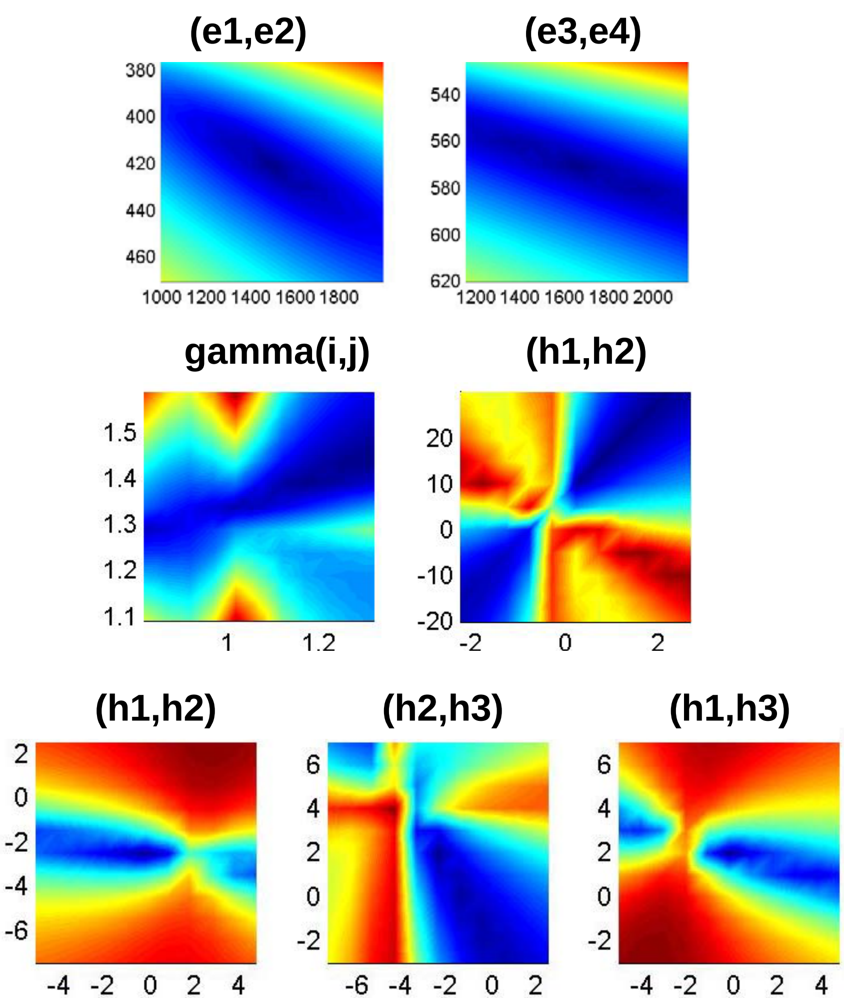
weighted residual of bilinear constraint

$$E(\mathbf{x}_i, \mathbf{x}_j; \mathbf{p}_7^{ij}) = w_{ij}\ \mathbf{x}_j^T \mathbf{F}_{\mathbf{p}_7^{ij}} \mathbf{x}_i$$

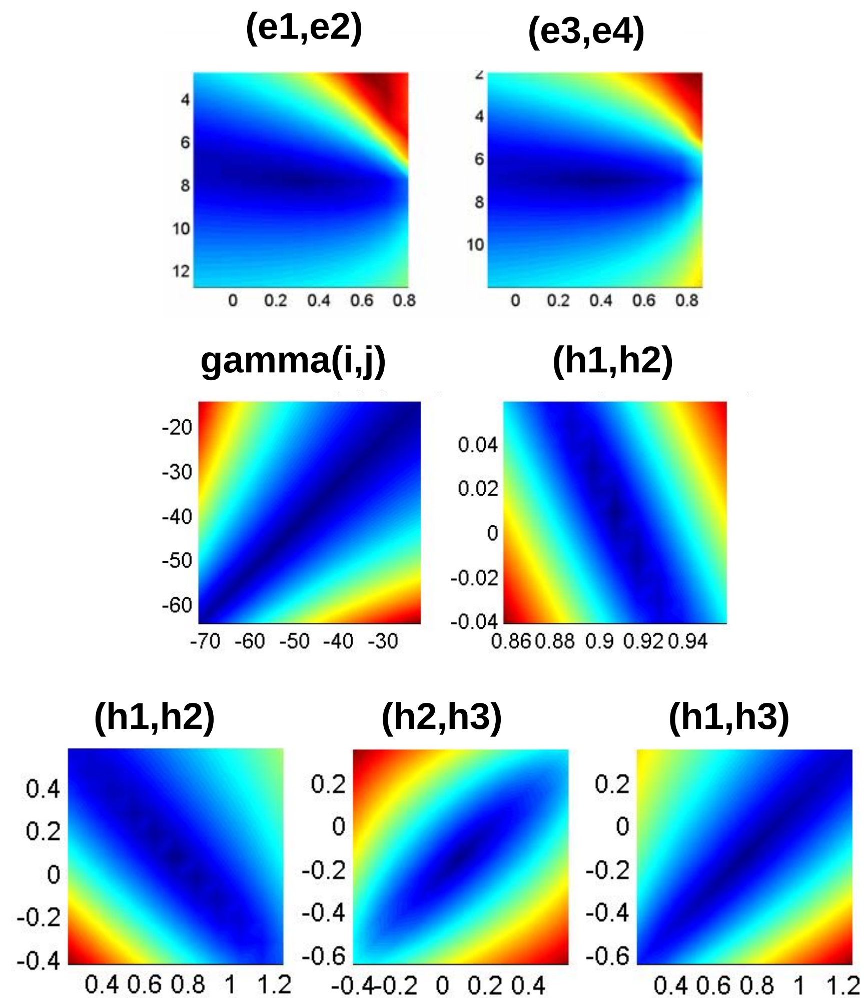
$$w_{ij} = \frac{1}{(\mathbf{F}_{ij}\mathbf{x}_i)_1^2 + (\mathbf{F}_{ij}\mathbf{x}_i)_2^2} + \frac{1}{(\mathbf{F}_{ij}^T\mathbf{x}_j)_1^2 + (\mathbf{F}_{ij}^T\mathbf{x}_j)_2^2}$$

Error Surface Depends on Basis

canonical basis



geometrically defined basis

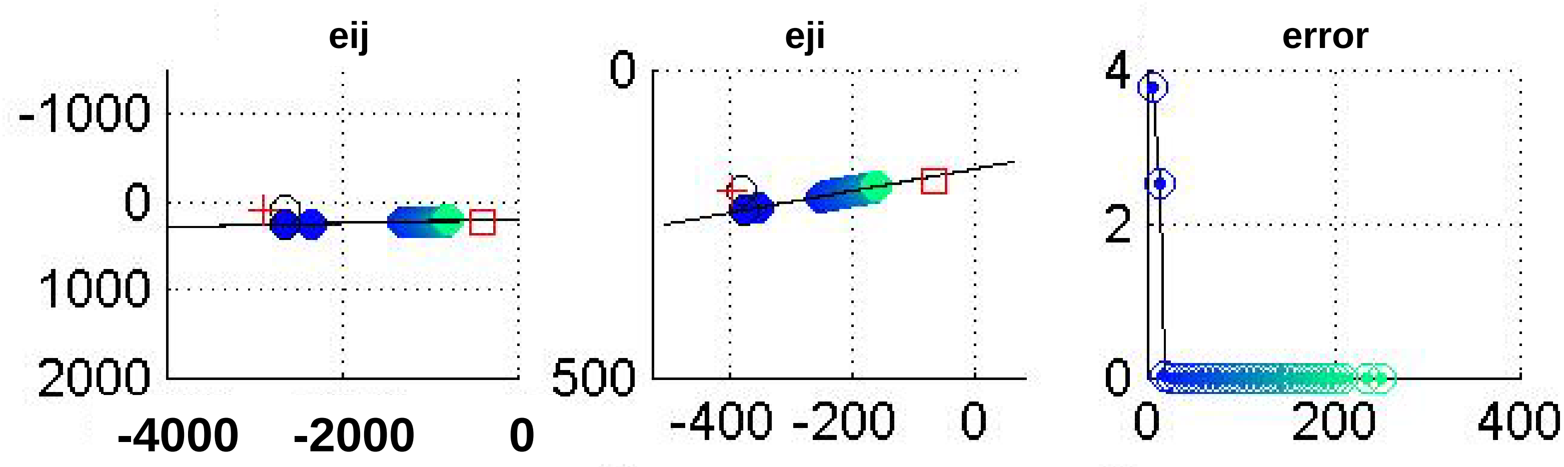


Nonlinear Trifocal Estimation

i	k	j

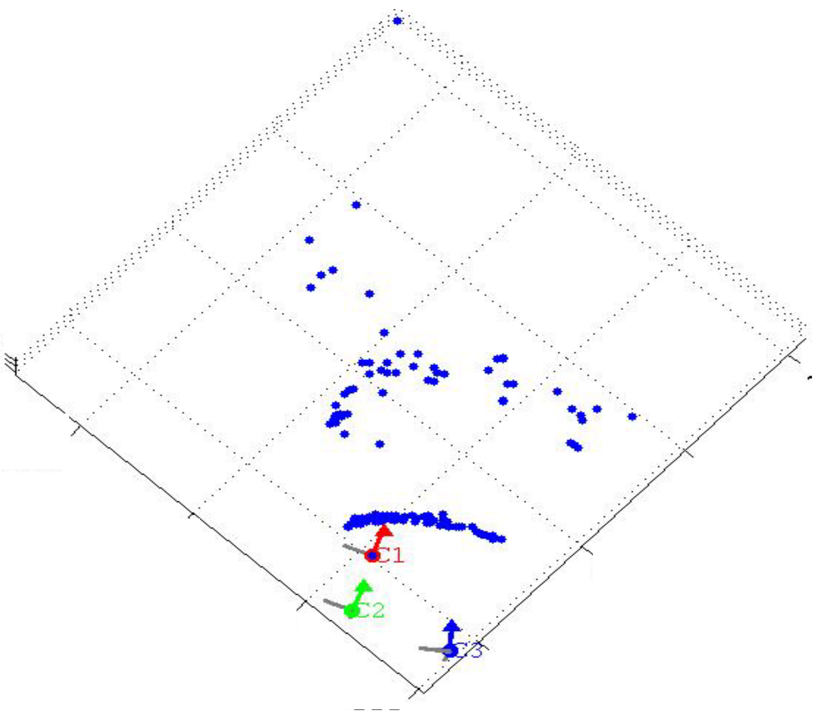
1. Initialize epipolar geometry
   - 8-point algorithm: linear solution to fundamental matrix for all view pairs
   - extract epipoles and epipolar collineations
2. 7D nonlinear minimization: bifocal parameters for view pairs (i,k) (j,k)
3. Trifocally constrained estimation for view pair (i,j)
   - compute trifocal lines
   - project parameters to trifocally constrained space
   - 4D nonlinear minimization for bifocal parameters

Convergence

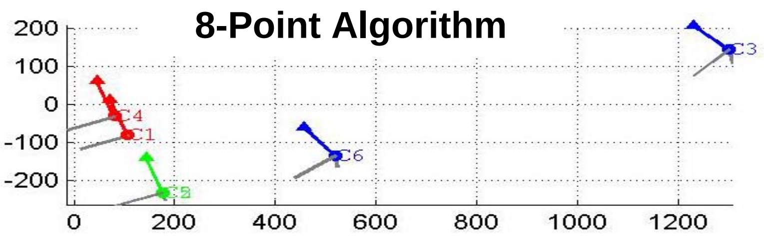


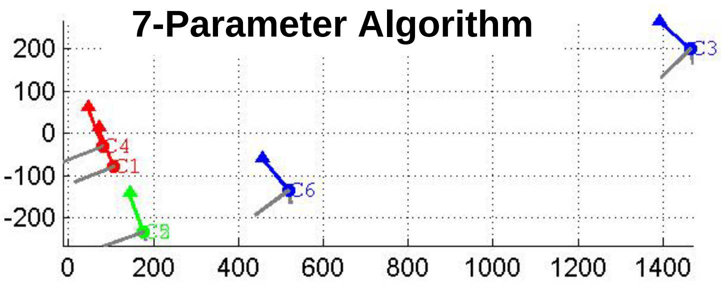
Ground Truth
8-point Algorithm
7-Parameter Search
Trifocal Projection
4-Parameter Search

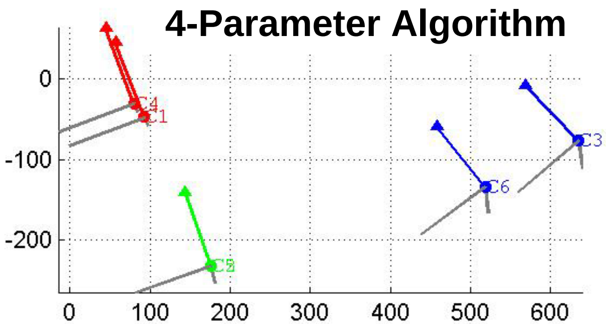
Ground Truth



Results





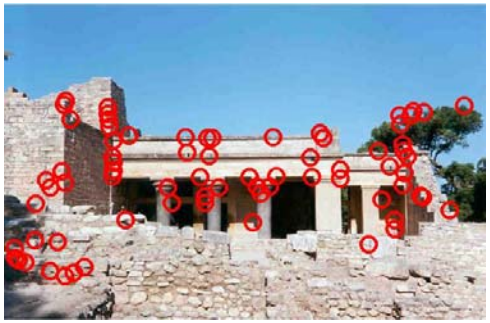


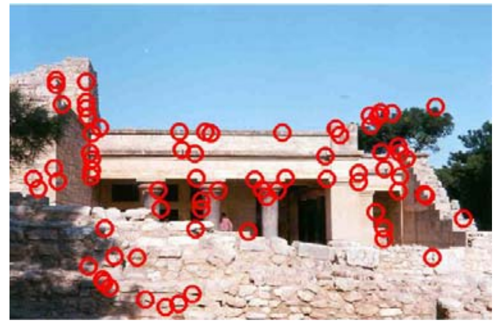
knossos sequence

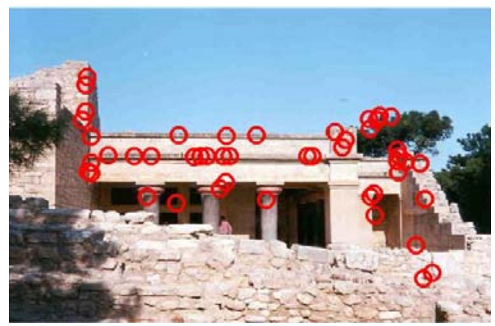
view i

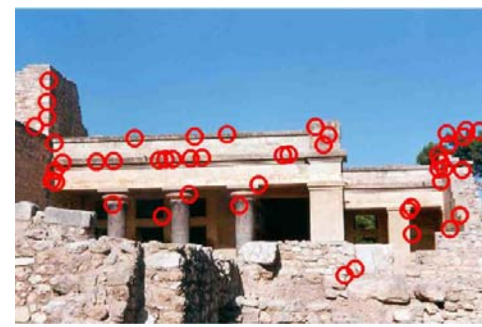
view k

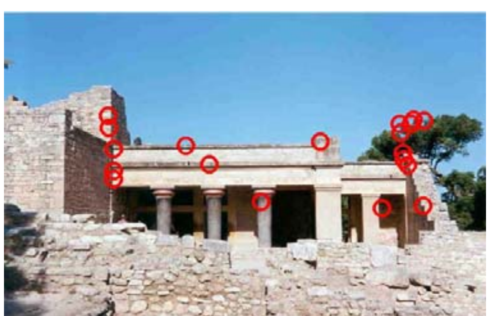
view j



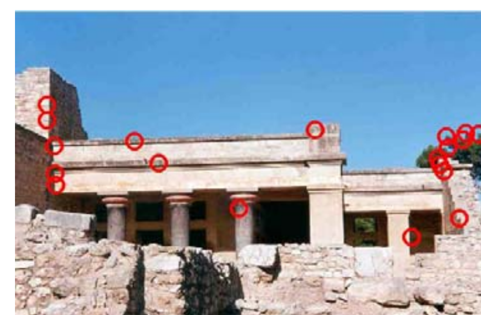




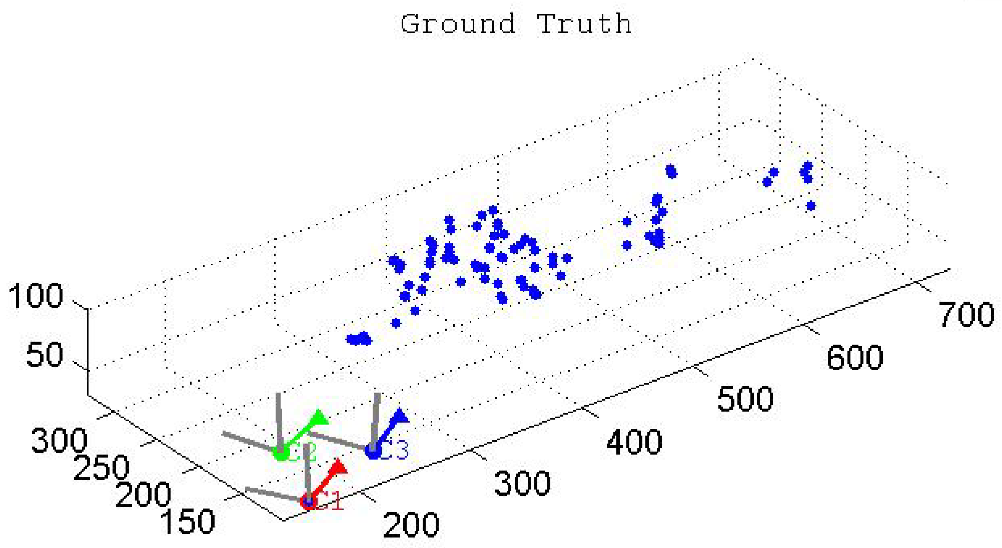




few
correspondences

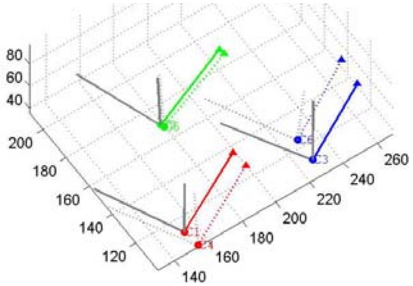


Ground Truth

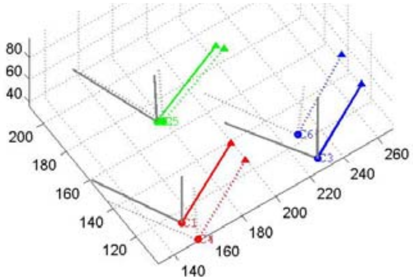


Results

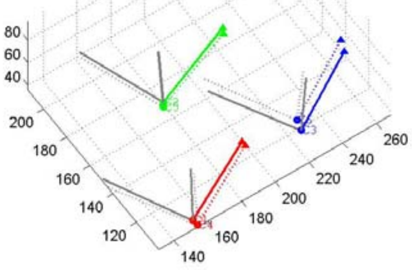
8-Point Algorithm



7-Parameter Algorithm



4-Parameter Algorithm



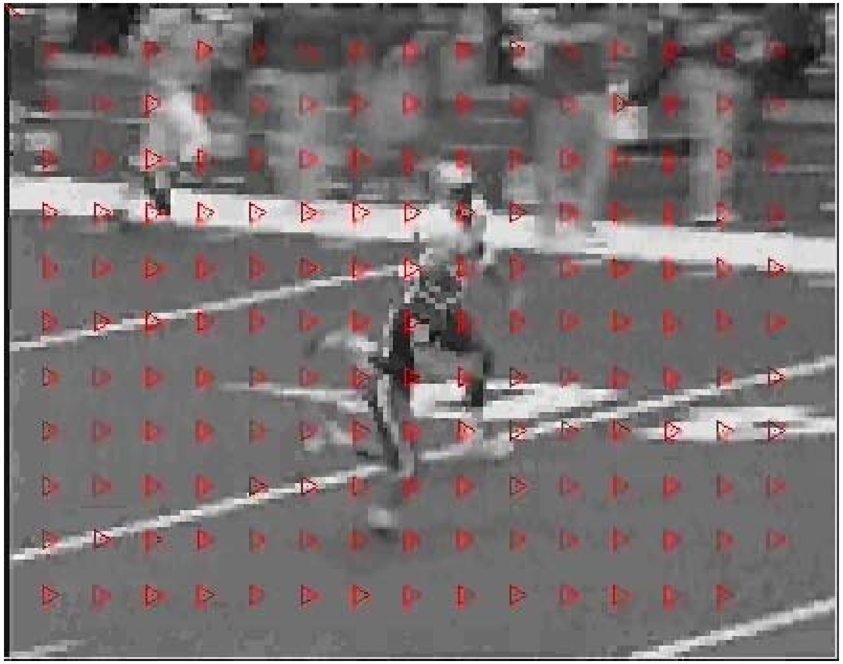
Summary

- Imposing projective constraints on camera geometry corrects the estimation of epipolar geometry
- Resulting camera configuration for multiple cameras is globally consistent

Outline

1. Projective Geometry Overview
2. Minimal Projective Parameters
3. Projective Parameter Estimation
4. Motion Boundary Detection
5. Conclusion

Camera and Scene Motion

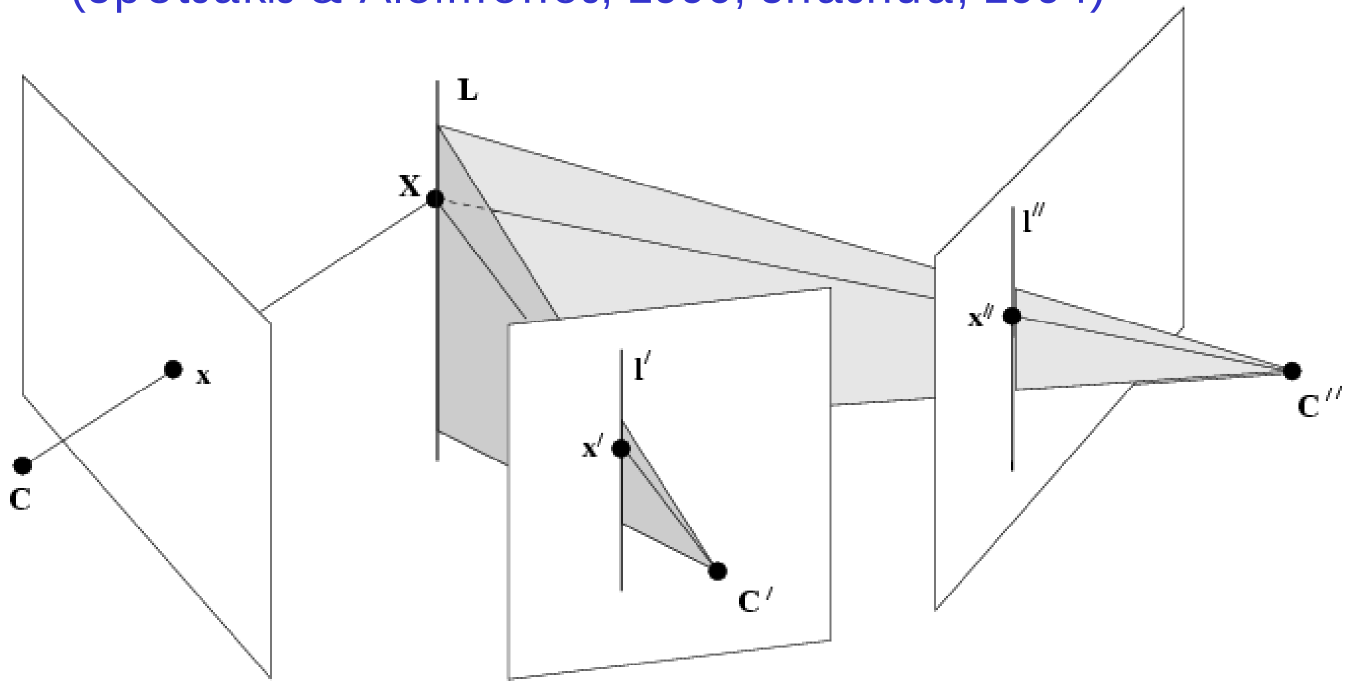


Combining Intensity and Geometry

trifocal tensor

projective linear form relating a point-line-line

(Spetsakis & Aloimonos, 1990; Shashua, 1994)



$$T(\mathbf{x}_i, \mathbf{l}_j, \mathbf{l}_k) = 0$$

Tensor Brightness Constraint

(Shashua & Hannah, 1995; Shashua & Stein, 1997)

$$u\,I_x + v\,I_y + I_t = 0$$

$$u = x - x_0 \qquad v = y - y_0$$

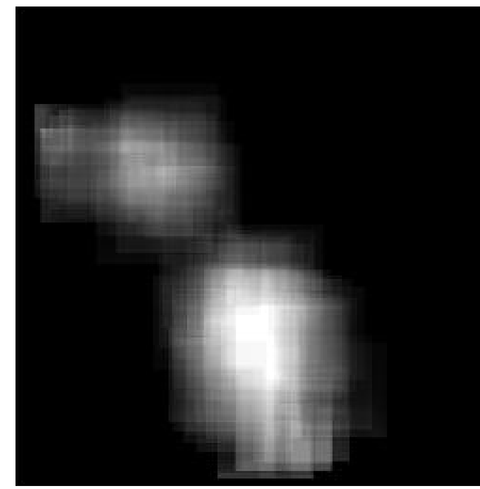
$$ax + by + c = 0$$

$$(a,b,c)^T \cong \begin{bmatrix} I_x \\ I_y \\ I_t - x_0 I_x - y_0 I_y \end{bmatrix}$$

- Horn-Schunk brightness constraint is linear in point coordinates
- Defines line in each image containing matching point
- Spatiotemporal gradient at every pixel provides test of rigid motion

Motion Boundary Detection

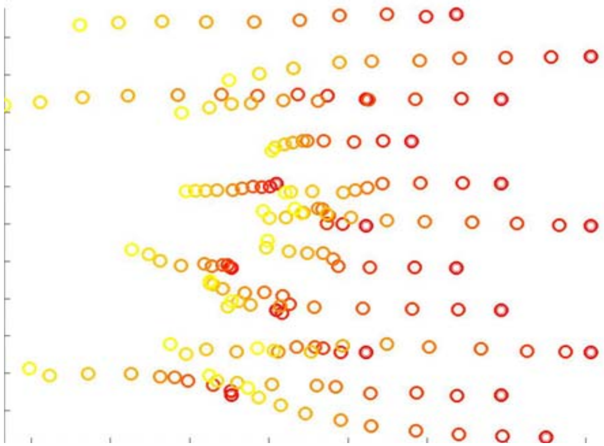
- Partition image into windows and solve for trifocal tensor coefficients.
- Only regions with rigid 3D motion have a good fit.
- Sum residual error of tensor solution.



- High residuals indicate regions that cross a motion boundary.

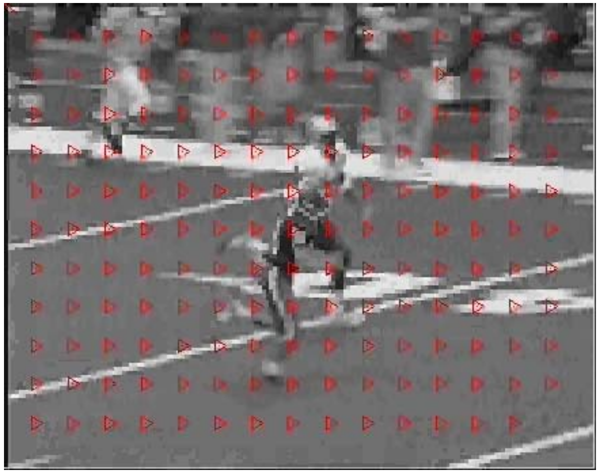
Multiple Frame Flow

- Track points over many frames



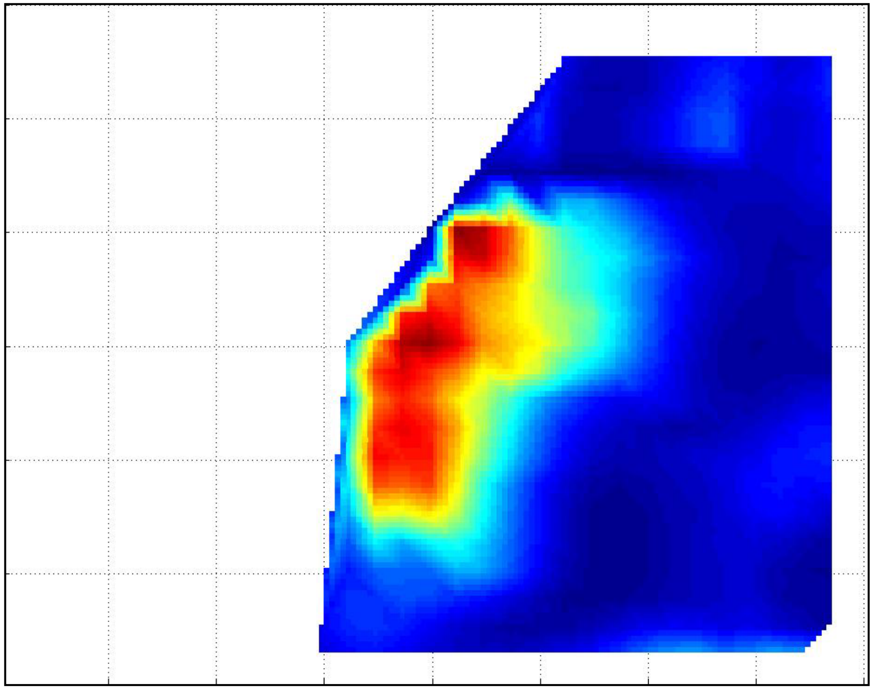
- Multi-frame tracks fall into separable classes

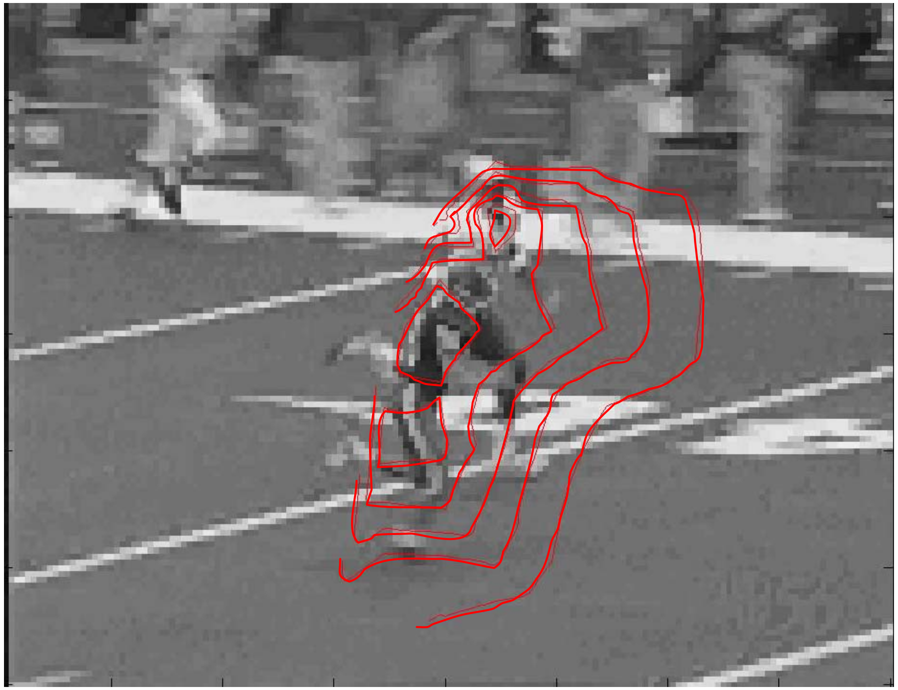
- Robustly fit tracks to linear approximation of instantaneous planar motion



$$\mathbf{x}(t) = \mathbf{x}_0 + t\,[\mathbf{A}\mathbf{x}_0 + \mathbf{b}]$$

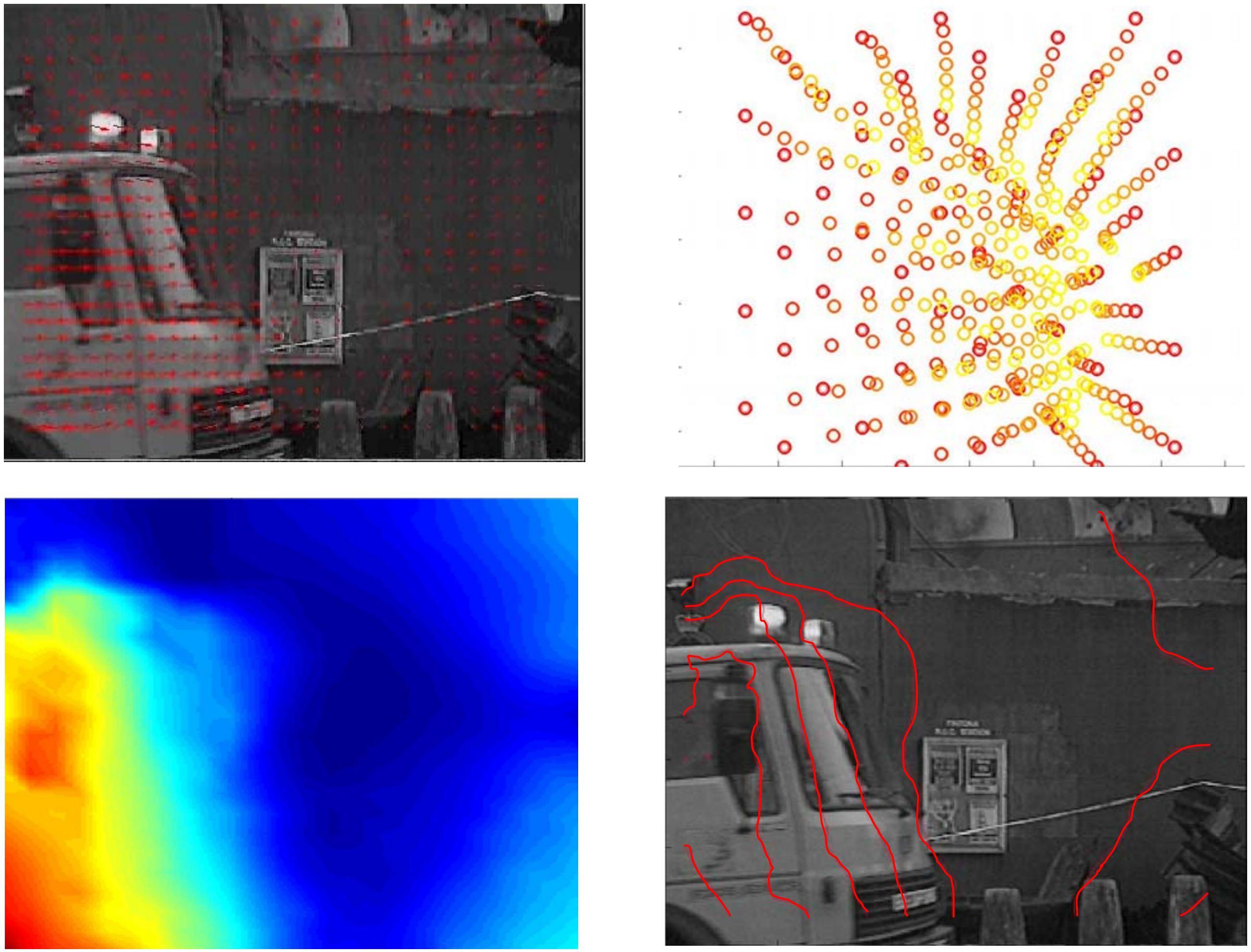
Detecting Independent Motions





Residual error of estimated motion model on all point tracks

Complexity of Motion Model



Conclusions

When possible, use domain and task knowledge to choose model:

- What type of information is needed
- What aspects of the imaging conditions are known or controlled
- What types of uncertainty can be modeled and compensated for

Future Needs

Role of learning in motion analysis:

- Supervised learning of geometric motion classes
- Data-driven model selection by flow classification
- Robust estimation of appropriate motion model
- Adaptive, time-varying estimation

END