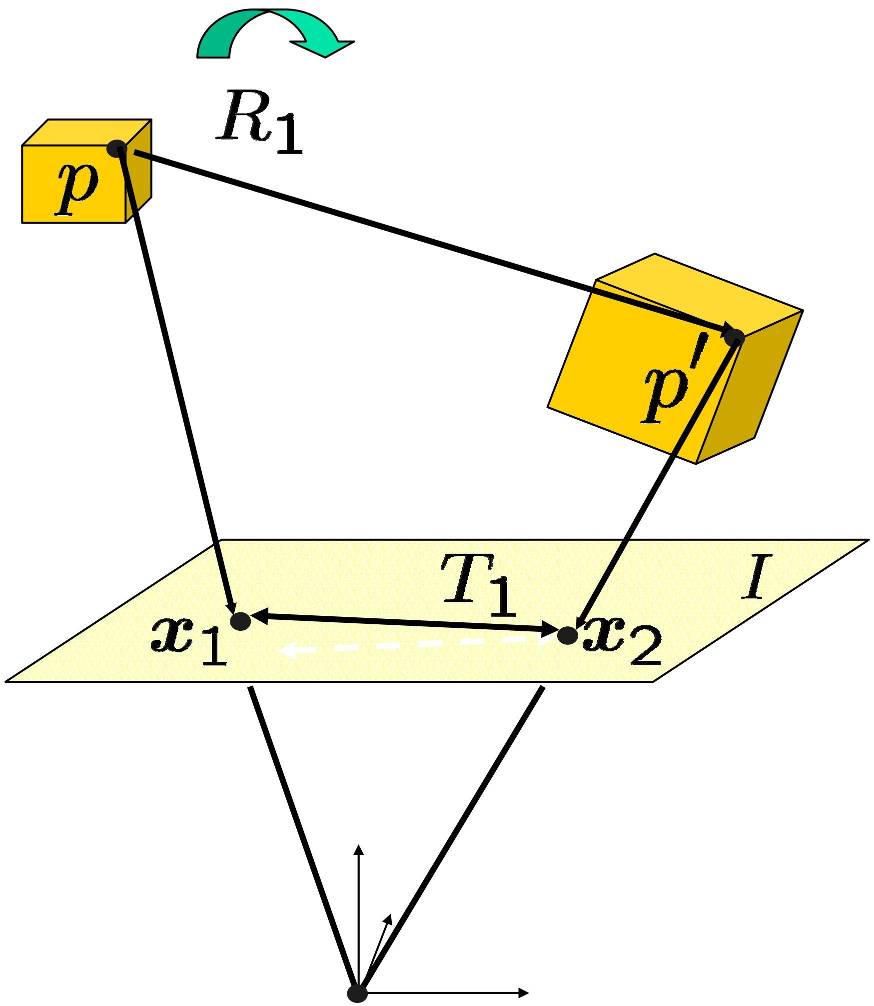
Reconstruction of Multiple Motions using Generalized Principal Component Analysis (GPCA)

René Vidal
Center for Imaging Science
Johns Hopkins University

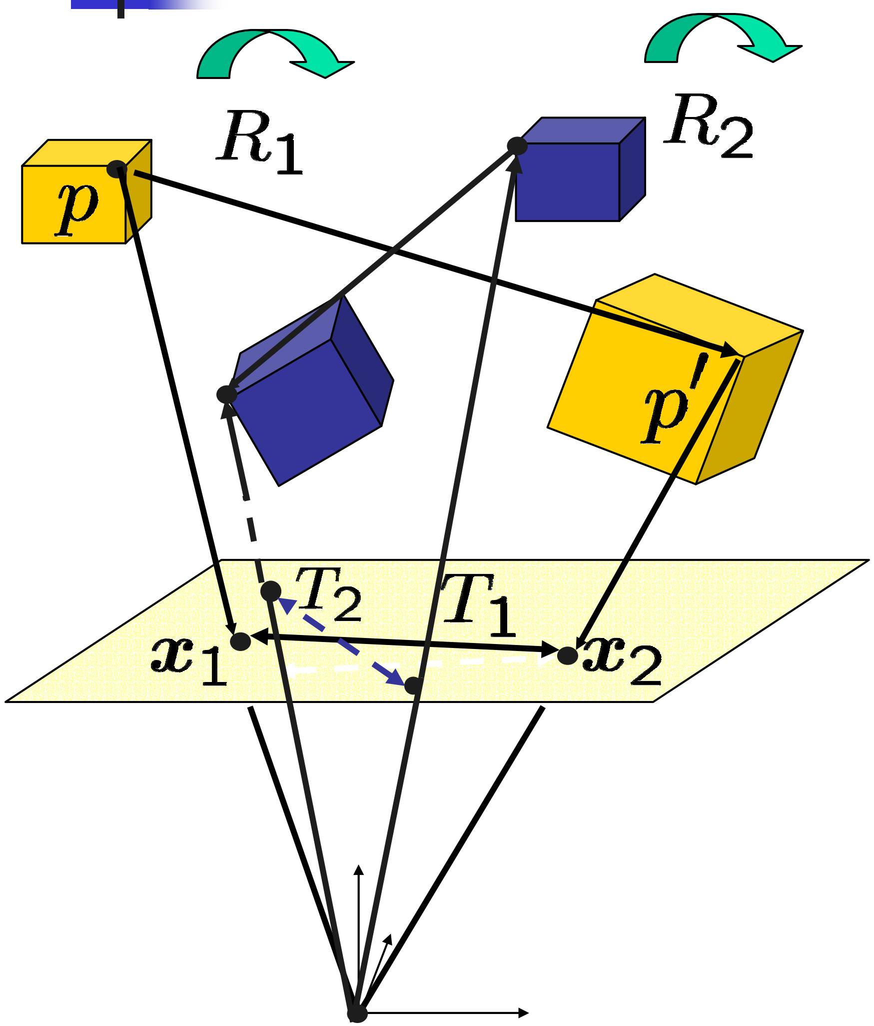
One-body two-views



- Image point: $\boldsymbol{x} = [x, y, z]^T \in \mathbb{R}^3$
- Camera motion
  - Rotation: $R_1 \in SO(3)$
  - Translation: $\widehat{T_1} \in so(3)$
- Epipolar constraint

$$\boldsymbol{x}_2^T \underbrace{\hat{T}_1 R_1}_{F_1 \in \mathbb{R}^{3\times 3}} \boldsymbol{x}_1 = 0$$

Multiple-bodies two-views



- Image point $\boldsymbol{x} = [x, y, z]^T \in \mathbb{R}^3$
- Camera motion
  - Rotation: $R_1 \in SO(3)$
  - Translation: $\widehat{T_1} \in so(3)$
- Epipolar constraint

$$\boldsymbol{x}_2^T \underbrace{\widehat{T}_1 R_1}_{F_1 \in \mathbb{R}^{3\times3}} \boldsymbol{x}_1 = 0$$

- Multiple motions

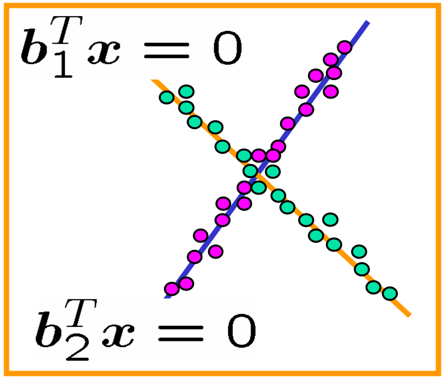
$$\{(R_i, T_i)\}_{i=1}^n \quad \{F_i \doteq \widehat{T_i} R_i\}_{i=1}^n$$

Motivation and problem statement

- A static scene: multiple 2D motion models
- A dynamic scene: multiple 3D motion models

- Given an image sequence, determine
  - Number of motion models (affine, Euclidean, etc.)
  - Motion model: affine (2D) or Euclidean (3D)
  - Segmentation: model to which each pixel belongs

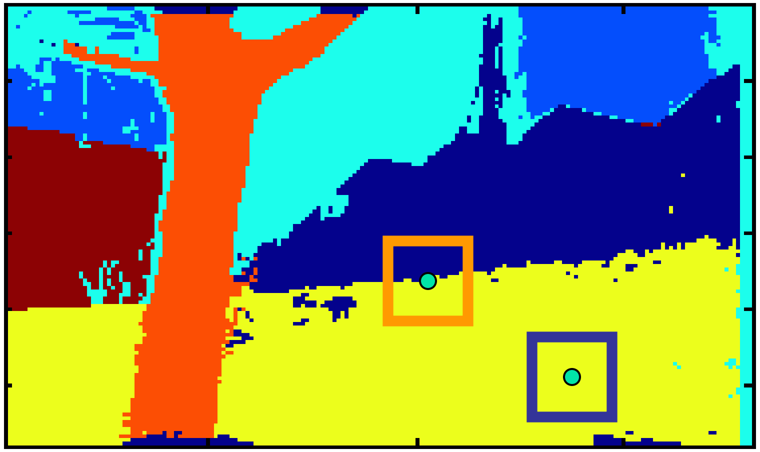
Prior work: chicken-and-egg problem



- Probabilistic techniques
  - Generative model
    - data membership + motion model
  - Expectation Maximization
    - E-step: Given motion models, segment image data
    - M-step: Given data segmentation, estimate motion models
- 2-D Motion Segmentation
  - Layered representation (Jepson-Black'93, Ayer-Sawhney '95, Darrel-Pentland'95, Weiss-Adelson'96, Weiss'97, Torr-Szeliski-Anandan '99)
- 3-D Motion Segmentation
  - EM+Reprojection Error: Feng-Perona'98
  - EM+Model Selection: Torr '98
- How to initialize iterative algorithms?

Prior work on 2-D motion segmentation

- Local methods (Wang-Adelson '93)
  - Estimate one model per pixel using data in a window
  - Cluster models with K-means
  - Iterate
  - Aperture problem
  - Motion across boundaries



- Global methods (Irani-Peleg '92)
  - Dominant motion: fit one motion model to all pixels
  - Look for misaligned pixels & fit a new model to them
  - Iterate

- Normalized cuts (Shi-Malik '98)
  - Similarity matrix based on motion profile
  - Segment pixels using eigenvector

Prior work on 3-D motion segmentation

- Factorization techniques, multiple views
  - Orthographic/discrete: Costeira-Kanade '98, Gear '98, Kanatani'01,'02,'03, Zelnik-Manor-Irani'03, Vidal-Hartley'04
  - Perspective/continuous: Vidal-Soatto-Sastry '02
  - Omnidirectional/continuous: Shakernia-Vidal-Sastry '03

- Special cases:
  - Points in a line (orth-discrete): Han and Kanade '00
  - Points in a line (persp.-continuous): Levin-Shashua '01
  - Points in a conic (perspective): Avidan-Shashua '01
  - Points in moving in planes: Sturm '02

- 2-body case: Wolf-Shashua '01

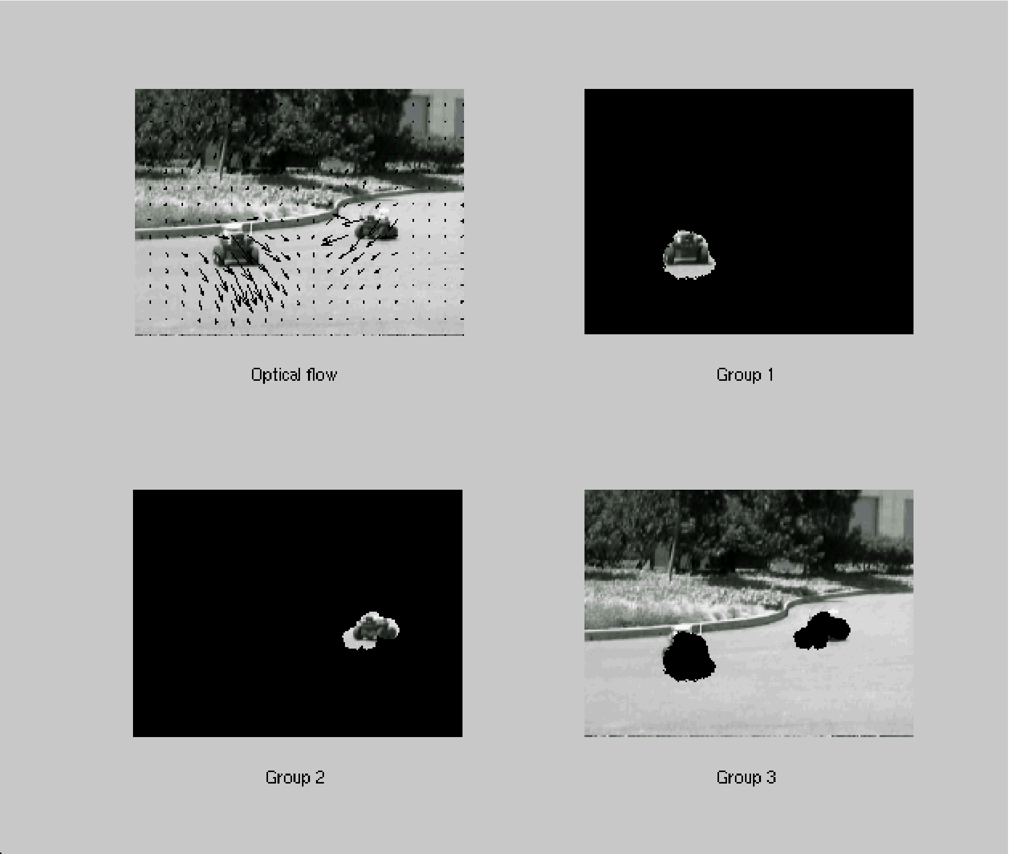
Our approach to motion segmentation

- Solve the initialization problem algebraically
  - Number of motions = degree of a polynomial
  - Motion parameters = factors of a polynomial
- Estimation of multiple motion models equivalent to estimation of one multibody motion model
  - Eliminate feature clustering
    - Find equation that does not depend on data clustering
  - Estimate multibody motion model to all image data
    - Fit a complex polynomial to data
  - Segment multibody motion model
    - Compute derivatives of the polynomial
- Applies to most motion models in vision
  - 2-D: translational, similarity and affine
  - 3-D: translational, fundamental matrix, homography, trifocal tensors, multiple affine cameras

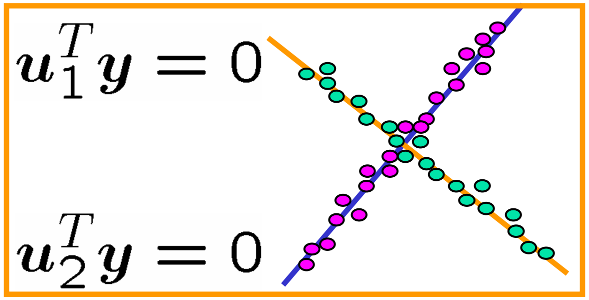
Segmentation of 2-D translational motions

- Scene having multiple optical flows $\{\boldsymbol{u}_i \in \mathbb{P}^2\}_{i=1}^n$
- Brightness constancy constraint (BCC) gives

$$\boldsymbol{y}^T\boldsymbol{u} = I_x u + I_y v + I_t = 0$$



- Multiple BCCs



- Multibody brightness constancy constraint

$$p_2(\boldsymbol{y}) = (\boldsymbol{u}_1^T\boldsymbol{y})(\boldsymbol{u}_2^T\boldsymbol{y}) = 0$$

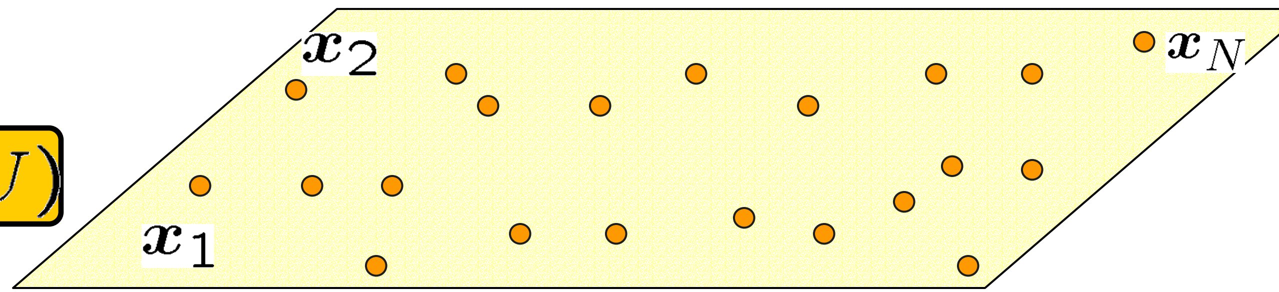
How to segment motions in general?

- One motion – one subspace:
  Principal Component Analysis (PCA)

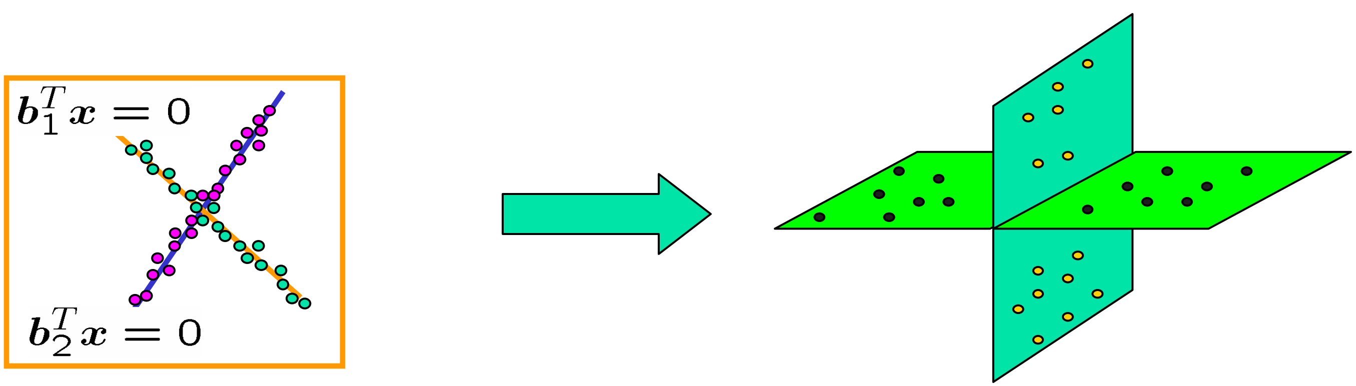
$$U \Sigma V^T = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_N] \in \mathbb{R}^{K \times N}$$

Basis for S

$\dim(S) = \text{rank}(U)$

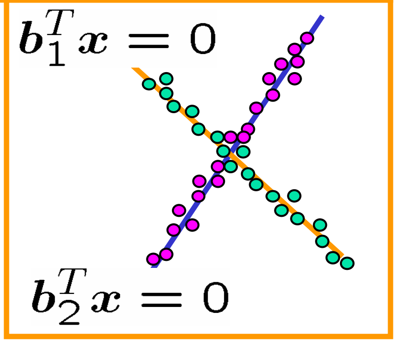


- Multiple motions – multiple subspaces
  Generalized Principal Component Analysis (PCA)



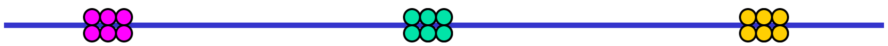
Generalized Principal Component Analysis

- Given points on multiple subspaces, identify
  - The number of subspaces and their dimension
  - A basis for each subspace
  - The segmentation of the data points
- "Chicken-and-egg" problem
  - Given segmentation, estimate subspaces
  - Given subspaces, segment the data



- Prior work
  - Geometric approaches: 2 planes in R^3 (Shizawa-Maze '91)
  - Factorization approaches: (Boult-Brown '91, Costeira-Kanade '98, Kanatani '01) cluster the data+ apply standard PCA to each cluster
  - Iterative algorithms: e.g. K-plane clustering (Bradley'00)
  - Probabilistic approaches (Tipping-Bishop '99): learn the parameters of a mixture model using e.g. EM
- Initialization?

Motivating example: algebraic clustering in 1D



$$x = b_1 \text{ or } x = b_2 \cdots x = b_n$$

$$p_n(x) = (x - b_1) \cdots (x - b_n) = 0$$

$$p_n(x) = x^n + c_1 x^{n-1} + \cdots c_n = 0$$

$$p_n(x) = \begin{bmatrix} x^n & \cdots & x & 1 \end{bmatrix} \boldsymbol{c} = 0$$

$$P_n \boldsymbol{c} = \underbrace{\begin{bmatrix} x_1^n & \cdots & x_1 & 1 \\ x_2^n & \cdots & x_2 & 1 \\ \vdots & & \vdots & \vdots \\ x_N^n & \cdots & x_N & 1 \end{bmatrix}}_{P_n \in \mathbb{R}^{N \times (n+1}} \boldsymbol{c} = 0$$

How to compute n, c, b's?

- Number of clusters

$$n \doteq \min\{i : rank(P_i) = i\}$$

- Cluster centers

  Roots of $p_n(x)$

- Solution is unique if

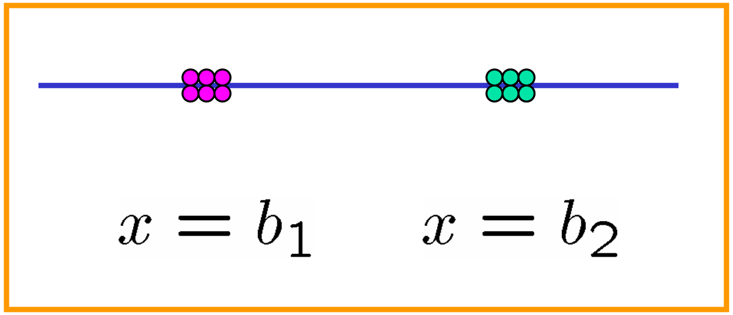
$$N_{points} \geq n_{groups}$$

- Solution is closed form if

$$n_{groups} \leq 4$$

GPCA: n hyperplanes = 1 polynomial of deg n

- 1-dimensional case

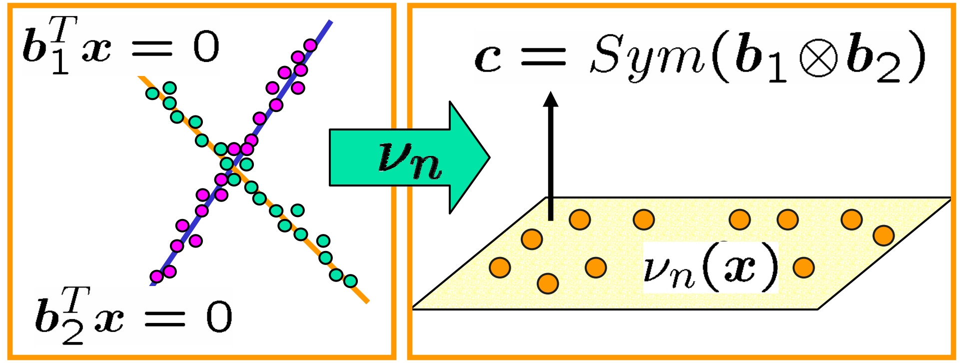


$$(x - b_1)\cdots(x - b_n) = 0$$

$$\sum_{i=0}^{n} c_i x^i = 0$$

$$\begin{bmatrix} x_1^n & \cdots & x_1 & 1 \\ \vdots & & \vdots & \vdots \\ x_N^n & \cdots & x_N & 1 \end{bmatrix} \boldsymbol{c} = 0$$

- K-dimensional case



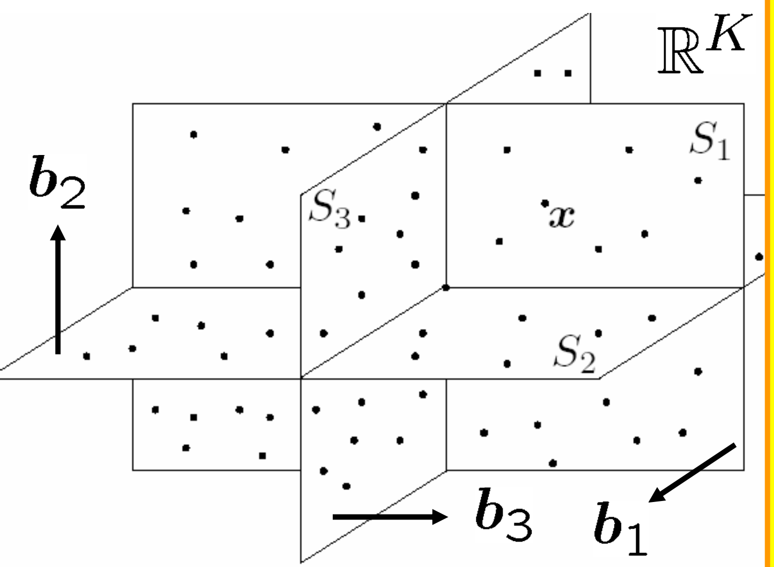
$$p_n(\boldsymbol{x}) = (\boldsymbol{b}_1^T \boldsymbol{x})\cdots(\boldsymbol{b}_n^T \boldsymbol{x}) = 0$$

$$\sum_I c_I \boldsymbol{x}^I = \nu_n(\boldsymbol{x})^T \boldsymbol{c} = 0$$

$$\begin{bmatrix} \nu_n(\boldsymbol{x}_1)^T \\ \vdots \\ \nu_n(\boldsymbol{x}_N)^T \end{bmatrix} \boldsymbol{c} = 0$$

GPCA: the case of hyperplanes

- Identify n $(K-1)$-dimensional subspaces of $\mathbb{R}^K$
  - K: dimension of ambient space (known)
  - n : number of subspaces (unknown)
  - $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$: normal to each subspace (unknown)



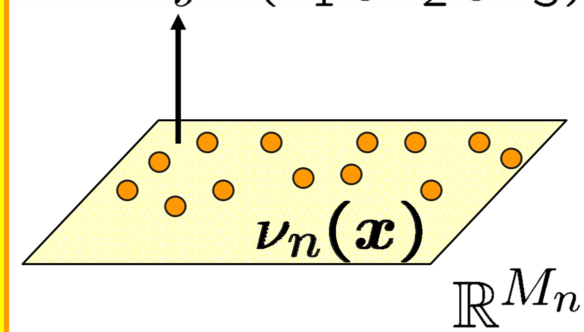
Veronese map

$$\nu_n : \mathbb{R}^K \mapsto \mathbb{R}^{M_n}$$

$$\nu_2 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x^2 \\ xy \\ y^2 \end{pmatrix}$$

$$M_n = \binom{n+K-1}{n}$$

$$\boldsymbol{c} = Sym(\boldsymbol{b}_1 \otimes \boldsymbol{b}_2 \otimes \boldsymbol{b}_3)$$



Solution to motion segmentation by GPCA

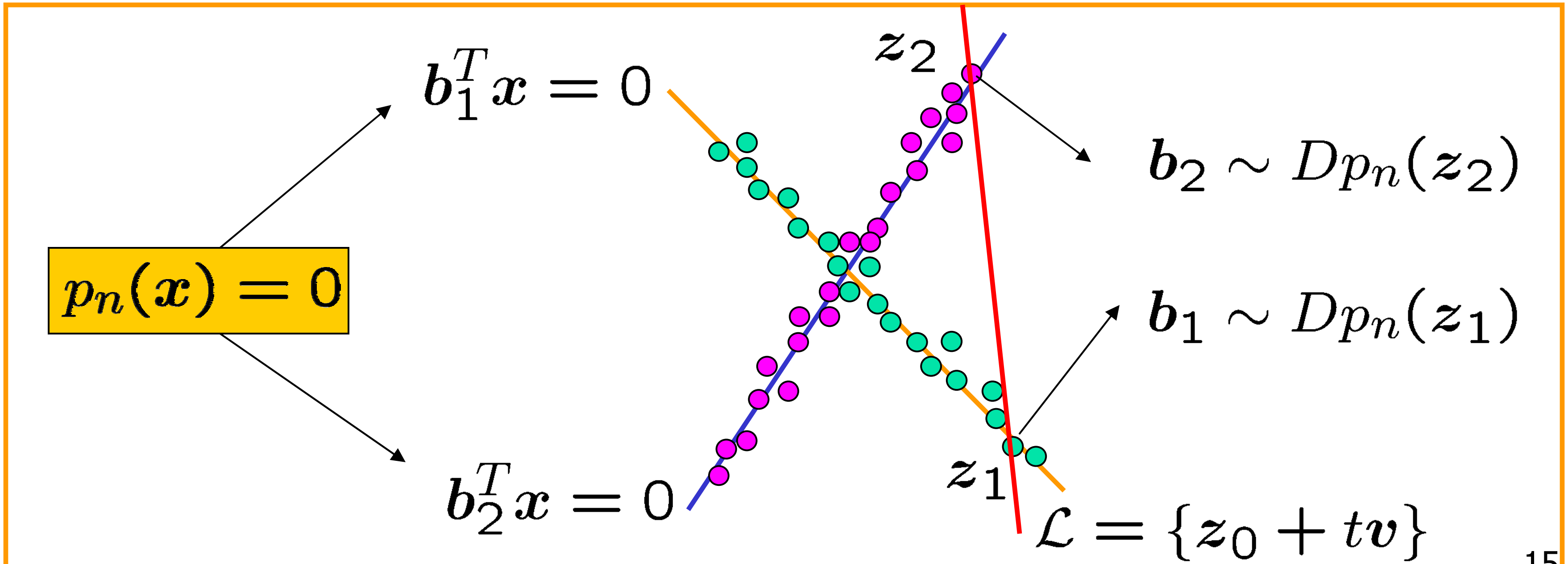
Theorem: Hyperplane clustering using GPCA

- Estimate multibody motion model: fit polynomial

$$p_n(\boldsymbol{x}) = (\boldsymbol{b}_1^T \boldsymbol{x}) \cdots (\boldsymbol{b}_n^T \boldsymbol{x}) = 0$$

- Estimate motion models: differentiate the polynomial

$$\boldsymbol{b}_i = Dp_n(\boldsymbol{x})|_{\boldsymbol{x}=\boldsymbol{z}_i} \qquad \begin{aligned} \boldsymbol{z}_i &= \boldsymbol{z}_0 + t_i \boldsymbol{v} \\ t_i &= \text{Root}\left[p_n(\boldsymbol{z}_0 + t\boldsymbol{v})\right] \end{aligned}$$



2-D motion segmentation results

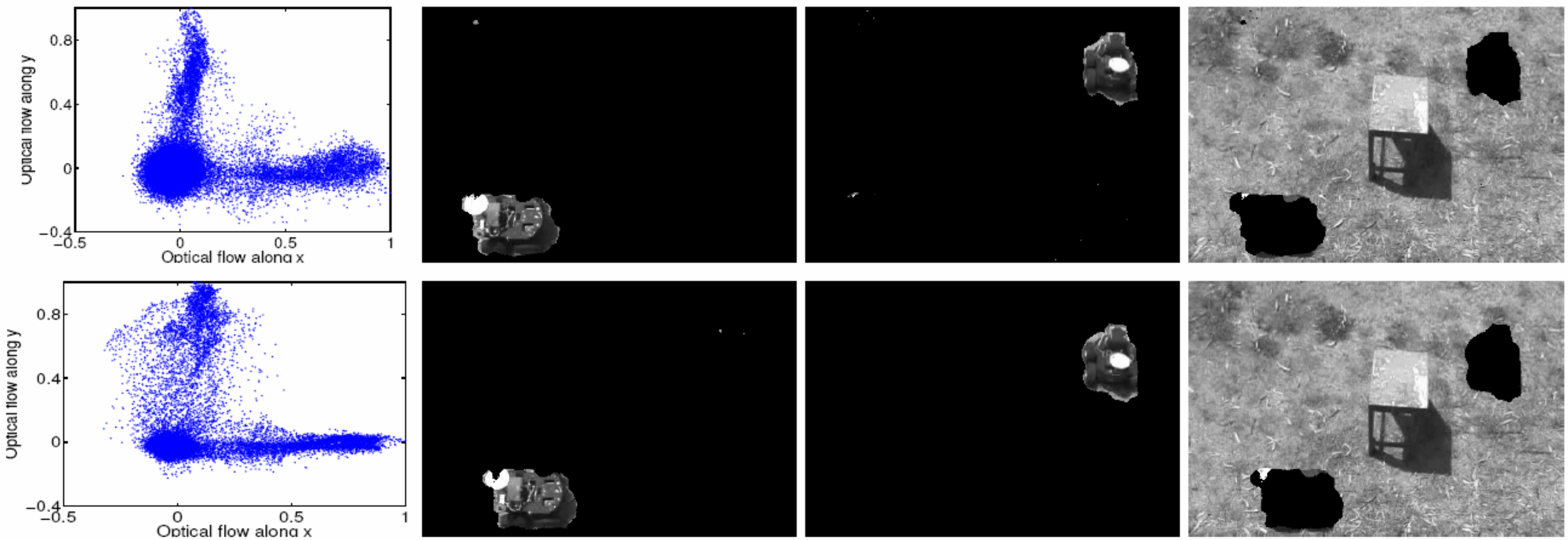


Fig. 1. Segmenting the optical flow of the two-robot sequence by clustering lines in $\mathbb{C}^2$.

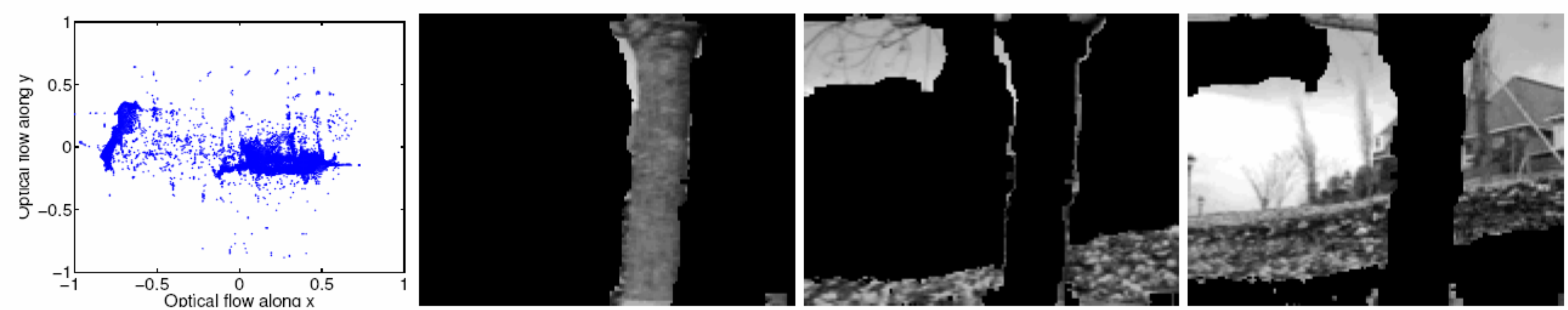
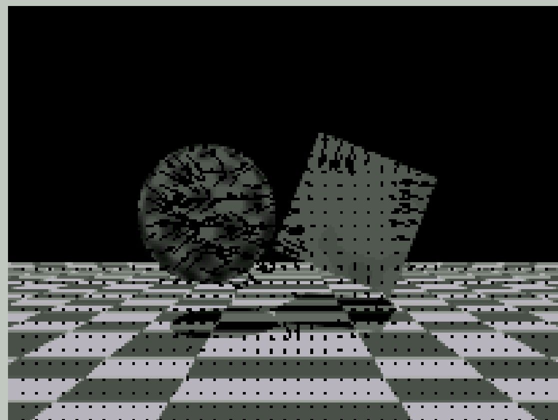
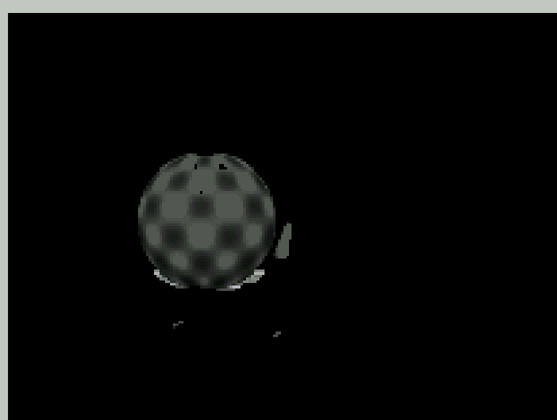


Fig. 2. Segmenting the optical flow of the flower-garden sequence by clustering lines in $\mathbb{C}^2$.

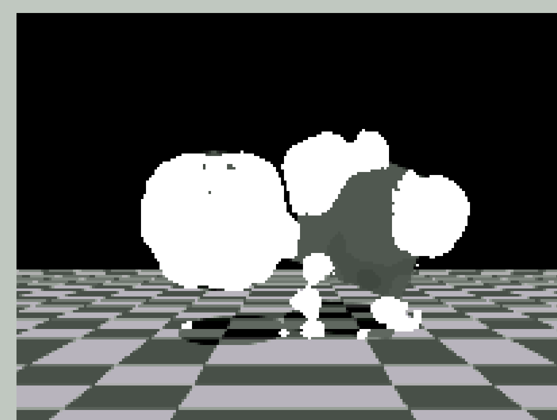


Optical flow

Group 1



Group 2



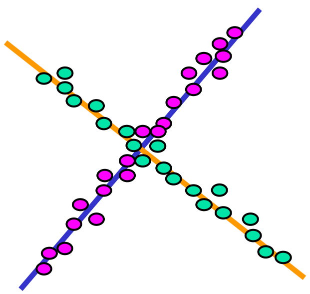
Group 3

Segmentation of 3-D translational motions

- Multiple objects translating in 3D $\{e_i \in \mathbb{R}^3\}_{i=1}^n$
- Epipolar constraint gives GPCA problem with K=3

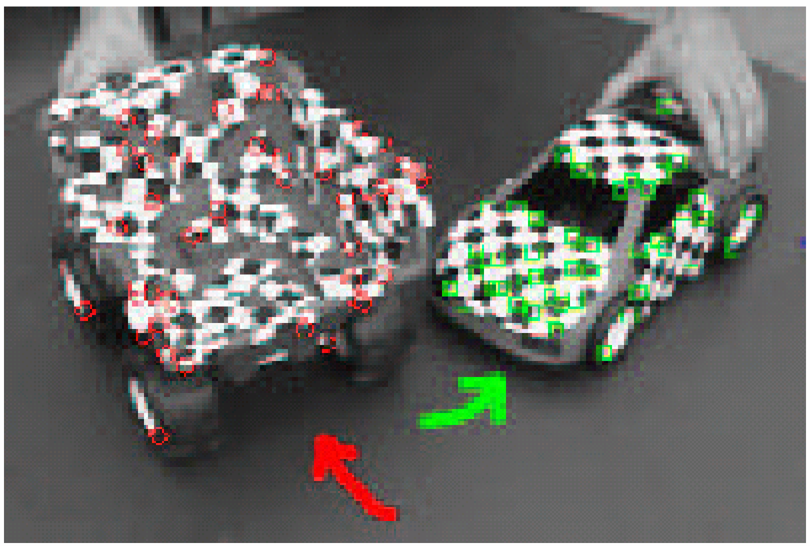
$$e_1^T(x_1 \times x_2) = 0$$

$$e_2^T(x_1 \times x_2) = 0$$

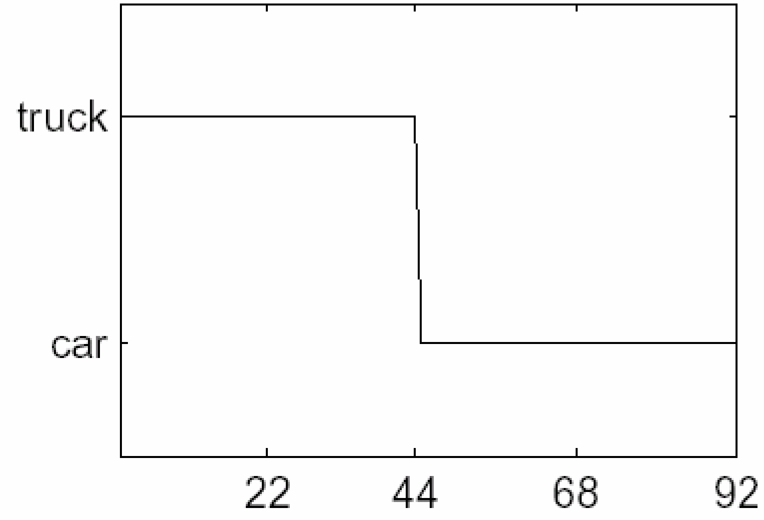


Multibody epipolar const.

$$p_n(y) = (e_1^T y) \cdots (e_n^T y)$$

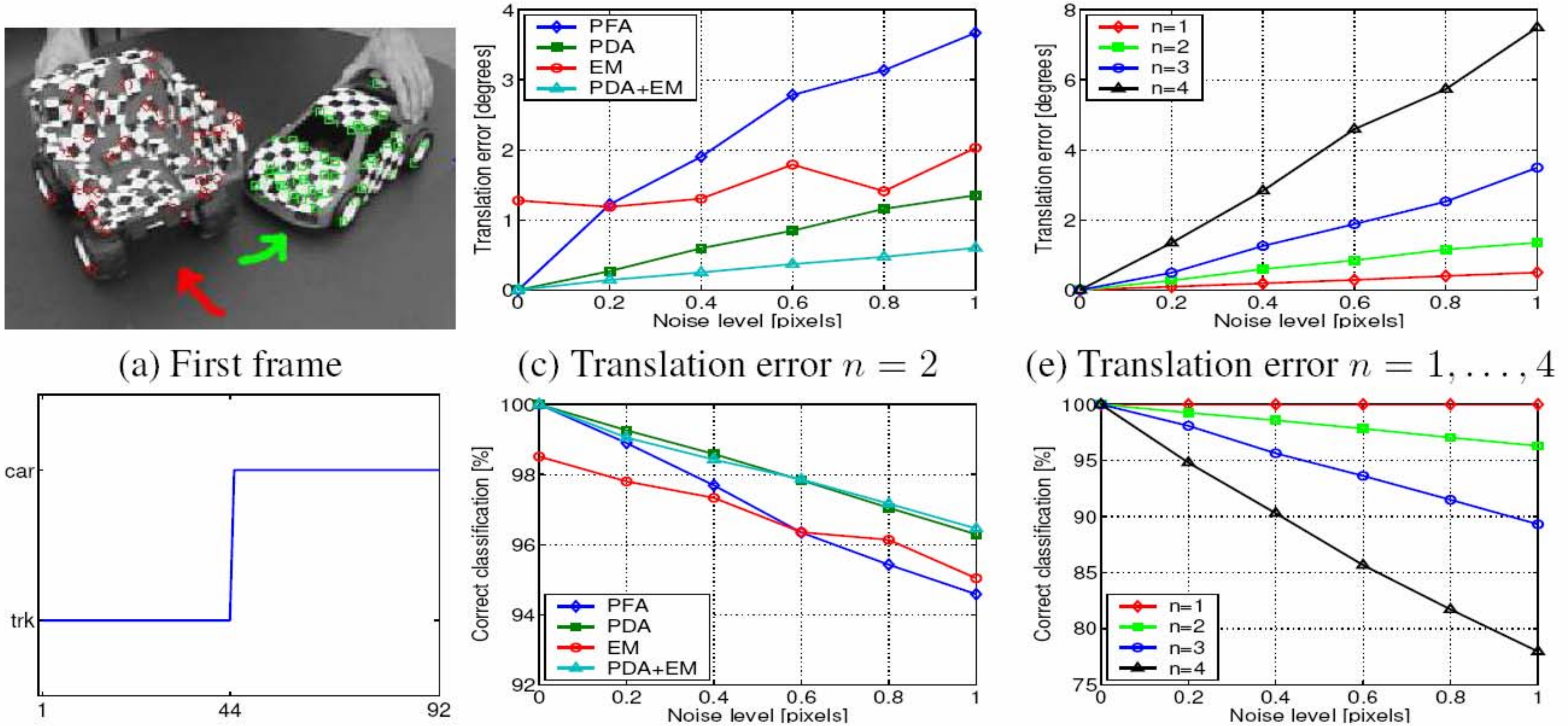


(a) First frame



(b) Feature segmentation

Segmentation of 3-D translational motions

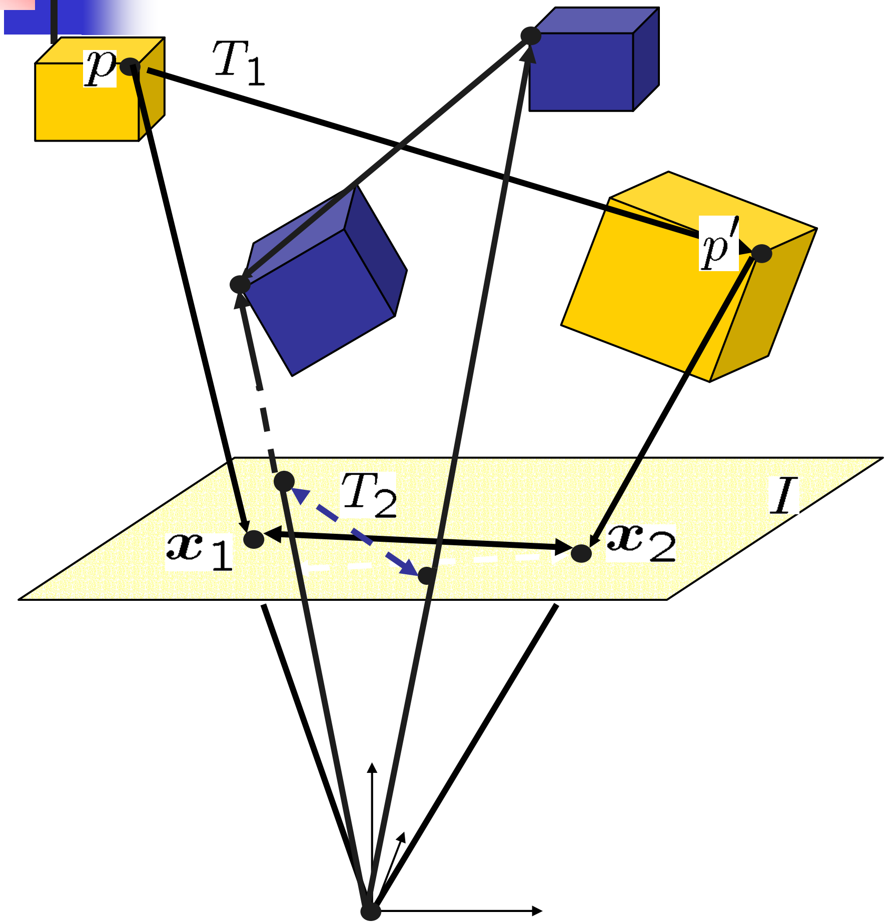


(a) First frame (c) Translation error $n = 2$ (e) Translation error $n = 1, \ldots, 4$

(b) Feature segmentation (d) % of correct classif. $n = 2$ (f) % of correct classif. $n = 1, \ldots, 4$

Fig. 3. Segmenting 3-D translational motions by clustering planes in $\mathbb{R}^3$. Left: segmenting a real sequence with 2 moving objects. Center: comparing our algorithm with PFA and EM as a function of noise in the image features. Right: performance of PFA as a function of the number of motions.

Segmentation of rigid motions: 2 views



- Rotation: $R_1 \in SO(3)$
- Translation: $\widehat{T_1} \in so(3)$
- Epipolar constraint

$$x_2^T \underbrace{\widehat{T}_1 R_1}_{F_1 \in \mathbb{R}^{3\times 3}} x_1 = 0$$

- Multiple motions

$$\{(R_i, T_i)\}_{i=1}^n \quad \{F_i \doteq \widehat{T_i} R_i\}_{i=1}^n$$

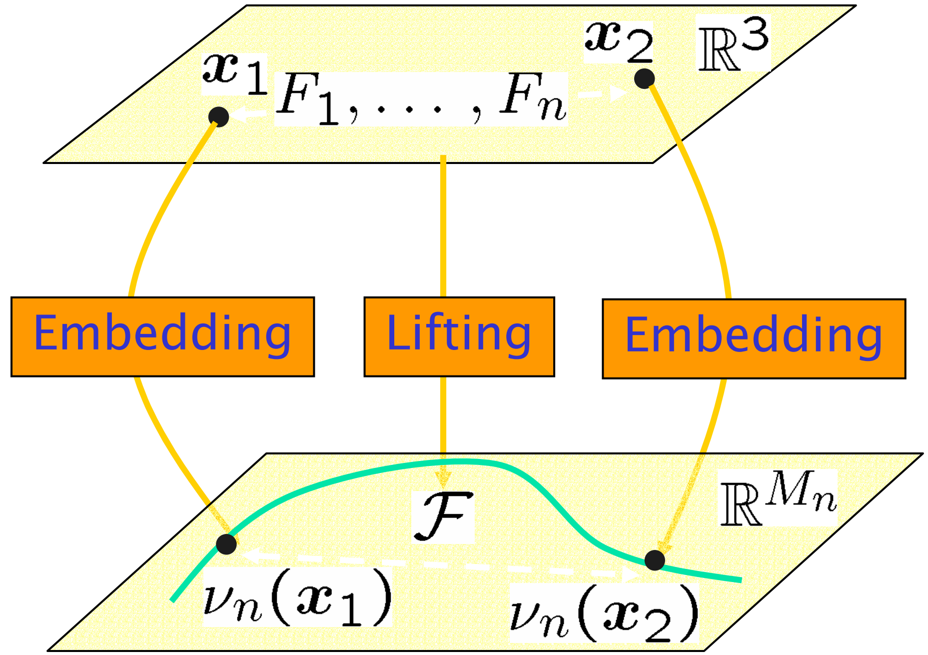
Multibody epipolar constraint

$$\prod_{i=1}^{n} (x_2^T F_i x_1) = 0$$

- Satisfied by ALL points regardless of segmentation
- Segmentation is algebraically eliminated!!!

The multibody fundamental matrix

$$\prod_{i=1}^{n} (\boldsymbol{x}_2^T F_i \boldsymbol{x}_1) = 0$$



$$\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1) = 0$$

Bilinear on embedded data!

- Veronese map (polynomial embedding)

$$\nu_n : [x, y, z]^T \mapsto [x^n, x^{n-1}y, x^{n-1}z, \ldots, z^n]^T \in \mathbb{R}^{M_n} \quad (\mathbb{R}^{\frac{(n+1)(n+2)}{2}})$$

- Multibody fundamental matrix $\mathcal{F} \doteq \sum_{\sigma \in \mathfrak{S}_n} F_{\sigma(1)} \otimes \cdots \otimes F_{\sigma(n)}$

$$\mathcal{F} \in \mathbb{R}^{M_n \times M_n} : \quad 3 \times 3 \quad 6 \times 6 \quad 10 \times 10 \quad \cdots$$

Estimation of multibody fundamental matrix

1-body motion

$$x_2^T \underbrace{F}_{3\times 3} x_1 = 0$$

$$\underbrace{A_1\left(\{x_1^j, x_2^j\}_{j=1}^N\right)}_{\in\mathbb{R}^{N\times 9}} f = 0$$

$$\boxed{\text{rank}(A_1) = 8}$$

n-body motion

$$\nu_n(x_2)^T \underbrace{\mathcal{F}}_{M_n\times M_n} \nu_n(x_1) = 0$$

$$\underbrace{A_n\left(\nu_n(x_1), \nu_n(x_2)\right)}_{A_n\in\mathbb{R}^{N\times M_n^2}} f = 0$$

$$\boxed{\text{rank}(A_n) = M_n^2 - 1}$$

Estimation of the number of motions

- **Theorem:** Given $N \geq M_n^2 - 1$ image points corresponding to n motions, if at least 8 points correspond to each object, then

$$\mathrm{rank}(A_i) \begin{cases} > M_i^2 - 1, & \text{if } i < n, \\ = M_i^2 - 1, & \text{if } i = n, \\ < M_i^2 - 1, & \text{if } i > n. \end{cases}$$

$$n = \min\{i : \mathrm{rank}(A_i) = M_i^2 - 1\}$$

Minimum number of points

n	1	2	3	4
N	8	35	99	225

Segmentation of fundamental matrices

Given $\{(\boldsymbol{x}_1^j, \boldsymbol{x}_2^j)\}_{j=1}^N$

rank condition for n_{motions}

$$n = \min\{i : \text{rank}(A_i) = M_i^2 - 1\}$$

linear system F

$$A_n \boldsymbol{f} = 0$$

$$\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1) = \prod_{i=1}^{n} (\boldsymbol{x}_2^T F_i \boldsymbol{x}_1) = 0$$

$$\mathcal{F} \in \mathbb{R}^{M_n \times M_n}$$

$$F_1 \quad F_2 \quad \ldots \quad F_n \in \mathbb{R}^{3\times 3}$$

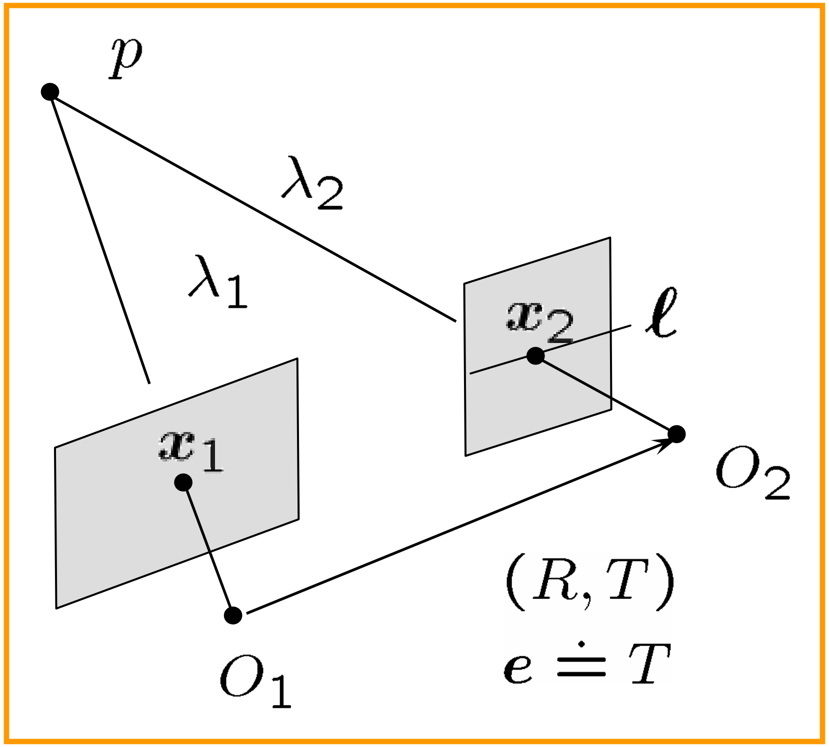
$\mathcal{F}$, $\boldsymbol{x}_1$ → $\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots, \boldsymbol{\ell}_n$ → $\tilde{\boldsymbol{e}}$ → $\boldsymbol{e}_1, \boldsymbol{e}_2, \ldots, \boldsymbol{e}_n$ → $F_1, F_2, \ldots, F_n$

Multibody epipolar transfer

Multibody epipole

Fundamental matrices

Multibody epipolar transfer



$$\boldsymbol{\ell}_i \doteq F_i \boldsymbol{x}_1 \in \mathbb{R}^3$$

Lifting

$$\tilde{\boldsymbol{\ell}} \doteq \mathcal{F} \nu_n(\boldsymbol{x}_1) \in \mathbb{R}^{M_n}$$

Multibody epipolar line

$$\tilde{\boldsymbol{\ell}} \doteq \sum_{\sigma \in \mathfrak{S}_n} \boldsymbol{\ell}_{\sigma(1)} \otimes \cdots \otimes \boldsymbol{\ell}_{\sigma(n)}$$

$$\boldsymbol{\ell}_1, \quad \boldsymbol{\ell}_2, \quad \dots, \quad \boldsymbol{\ell}_n$$

Epipolar lines are the derivatives of the multibody epipolar constraint at an image pair

Polynomial differentiation

$$\boldsymbol{\ell} = \frac{\partial\left(\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1)\right)}{\partial \boldsymbol{x}_2}$$

Multibody epipole

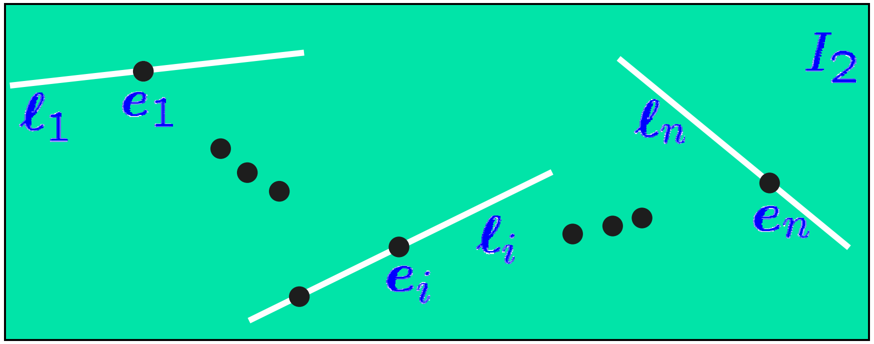
$\mathbb{R}^3$ $\quad e_i^T F_i = 0$

$e_i^T \ell = 0$

Lifting →

$\nu_n(e_i)^T F = 0 \quad \mathbb{R}^{M_n}$

$\tilde{e}^T \nu_n(\ell) = 0 \quad \tilde{e} \in \mathbb{R}^{M_n}$



All epipolar lines must pass through n epipoles

$$p_n(\ell) \doteq (e_1^T \ell)(e_2^T \ell) \cdots (e_n^T \ell) = \tilde{e}^T \nu_n(\ell) = 0$$

- The multibody epipole solution of linear system
- Epipoles are derivatives of $p_n(\ell)$ at epipolar lines

$$B_n \tilde{e} \doteq \begin{bmatrix} \nu_n(\ell^1)^T \\ \nu_n(\ell^2)^T \\ \vdots \\ \nu_n(\ell^m)^T \end{bmatrix} \tilde{e} = 0$$

$$e_i = \left. \frac{\partial \big(p_n(\ell)\big)}{\partial \ell} \right|_{\ell = \ell_i}$$

Individual fundamental matrices

$$\mathcal{F} \in \mathbb{R}^{M_n \times M_n}$$

$$F_1 \quad F_2 \quad \dots \quad F_n \in \mathbb{R}^{3\times 3}$$

- Fundamental matrices from second-order derivatives of multibody epipolar constraint at the epipoles

$$F_i = \left.\frac{\partial\Big(\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1)\Big)}{\partial \boldsymbol{x}_1 \boldsymbol{x}_2}\right|_{\substack{\boldsymbol{x}_1 = \boldsymbol{e}'_i \\ \boldsymbol{x}_2 = \boldsymbol{e}''_i}}$$

The multibody 8-point algorithm

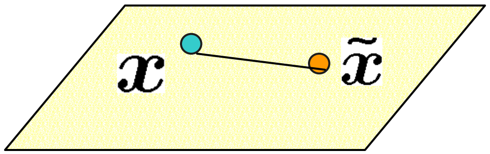
Image point	$\boldsymbol{x}_1 \in \mathbb{R}^3$	
	↓	Veronese map
Embedded image point	$\nu_n(\boldsymbol{x}_1) \in \mathbb{R}^{M_n}$	
	↓	Multibody epipolar transfer
Multibody epipolar line	$\tilde{\boldsymbol{\ell}} = F\nu_n(\boldsymbol{x}_1)$	
	↓	Polynomial Factorization
Epipolar lines	$\{\boldsymbol{\ell}_i \in \mathbb{R}^3\}_{i=1}^n$	
	↓	Linear system
Multibody epipole	$\tilde{\boldsymbol{e}}$	
	↓	Polynomial Factorization
Epipoles	$\{\boldsymbol{e}_i \in \mathbb{R}^3\}_{i=1}^n$	
	↓	Linear system
Fundamental matrix	$\{F_i \in \mathbb{R}^{3\times 3}\}_{i=1}^n$	

Remarks about the algorithm

- Algebraically equivalent to polynomial factorization
- Requires solving for roots of polynomial of degree **n** in one variable
- There is a closed form solution if **$n<5$**
- The algorithm is probably polynomial time
- It requires **$O(n^4)$** image points
- It neglects internal structure of the multibody fundamental matrix

Optimal 3D motion segmentation

$\boldsymbol{x} = \tilde{\boldsymbol{x}}$+noise



- Zero-mean Gaussian noise
- Constrained optimization problem on $Sym(SE(3) \otimes \cdots \otimes SE(3))$

$$\min \sum_{j=1}^{N} \|\tilde{\boldsymbol{x}}_1^j - \boldsymbol{x}_1^j\|^2 + \|\tilde{\boldsymbol{x}}_2^j - \boldsymbol{x}_2^j\|^2$$

$$\text{s.t. } (\tilde{\boldsymbol{x}}_2^{jT} F_1 \tilde{\boldsymbol{x}}_1^j) \cdots (\tilde{\boldsymbol{x}}_2^{jT} F_n \tilde{\boldsymbol{x}}_1^j) = 0$$

- Optimal function for 1 motion

$$\mathcal{J}(F_i) = \frac{(\boldsymbol{x}_2^T F_i \boldsymbol{x}_1)^2}{\|\widehat{e_3} F_i \boldsymbol{x}_1\|^2 + \|\widehat{e_3} F_i^T \boldsymbol{x}_2\|^2}$$

- Optimal function for n motions

$$\mathcal{J}(F_1, \ldots, F_n) = \frac{(\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1))^2}{\|\widehat{e_3} \mathcal{F} D\nu_n(\boldsymbol{x}_1)\|^2 + \|\widehat{e_3} \mathcal{F}^T D\nu_n(\boldsymbol{x}_2)\|^2}$$

- Solved using Riemanian Gradient Descent

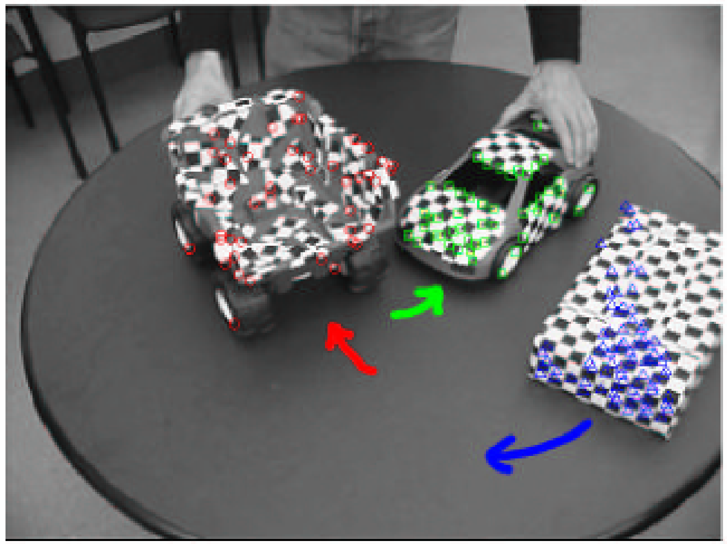
Comparison of 1 body and n bodies

Comparison of	2 views of 1 body	2 views of n bodies
An image pair	$\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^3$	$\nu_n(\boldsymbol{x}_1), \nu_n(\boldsymbol{x}_2) \in \mathbb{R}^{M_n}$
Epipolar constraint	$\boldsymbol{x}_2^T F \boldsymbol{x}_1 = 0$	$\nu_n(\boldsymbol{x}_2)^T F \nu_n(\boldsymbol{x}_1) = 0$
Fundamental matrix	$F \in \mathbb{R}^{3\times 3}$	$F \in \mathbb{R}^{M_n \times M_n}$
Linear estimation from N image pairs	$\begin{bmatrix} [\boldsymbol{x}_2^1 \otimes \boldsymbol{x}_1^1]^T \\ [\boldsymbol{x}_2^2 \otimes \boldsymbol{x}_1^2]^T \\ \vdots \\ [\boldsymbol{x}_2^N \otimes \boldsymbol{x}_1^N]^T \end{bmatrix} \boldsymbol{f} = 0$	$\begin{bmatrix} [\nu_n(\boldsymbol{x}_2^1) \otimes \nu_n(\boldsymbol{x}_1^1)]^T \\ [\nu_n(\boldsymbol{x}_2^2) \otimes \nu_n(\boldsymbol{x}_1^2)]^T \\ \vdots \\ [\nu_n(\boldsymbol{x}_2^N) \otimes \nu_n(\boldsymbol{x}_1^N)]^T \end{bmatrix} \boldsymbol{f} = 0$
Epipole	$\boldsymbol{e}^T F = 0$	$\nu_n(\boldsymbol{e})^T F = 0$
Epipolar lines	$\boldsymbol{\ell} = F \boldsymbol{x}_1 \in \mathbb{R}^3$	$\tilde{\boldsymbol{\ell}} = F \nu_n(\boldsymbol{x}_1) \in \mathbb{R}^{M_n}$
Epipolar line & point	$\boldsymbol{x}_2^T \boldsymbol{\ell} = 0$	$\nu_n(\boldsymbol{x}_2)^T \tilde{\boldsymbol{\ell}} = 0$
Epipolar line & epipole	$\boldsymbol{e}^T \boldsymbol{\ell} = 0$	$\tilde{\boldsymbol{e}}^T \nu_n(\boldsymbol{\ell}) = 0$

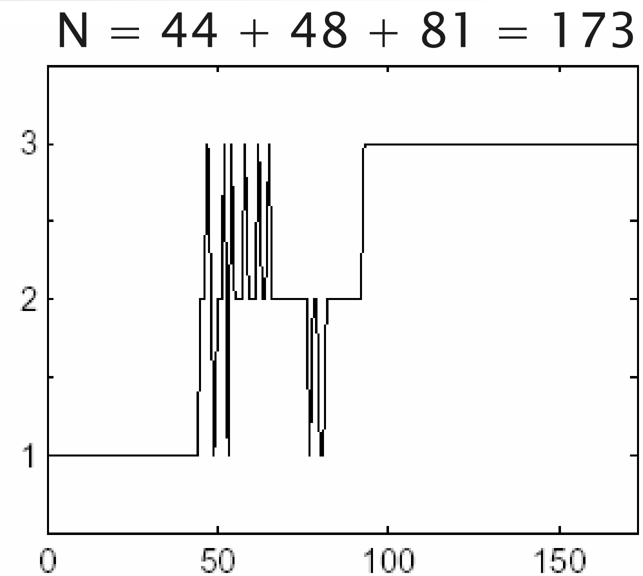
$$\mathcal{J}(F_i) = \frac{(\boldsymbol{x}_2^T F_i \boldsymbol{x}_1)^2}{\|\widehat{e_3} F_i \boldsymbol{x}_1\|^2 + \|\widehat{e_3} F_i^T \boldsymbol{x}_2\|^2}$$

$$\mathcal{J}(F_1, \ldots, F_n) = \frac{(\nu_n(\boldsymbol{x}_2)^T \mathcal{F} \nu_n(\boldsymbol{x}_1))^2}{\|\widehat{e_3} \mathcal{F} D\nu_n(\boldsymbol{x}_1)\|^2 + \|\widehat{e_3} \mathcal{F}^T D\nu_n(\boldsymbol{x}_2)\|^2}$$

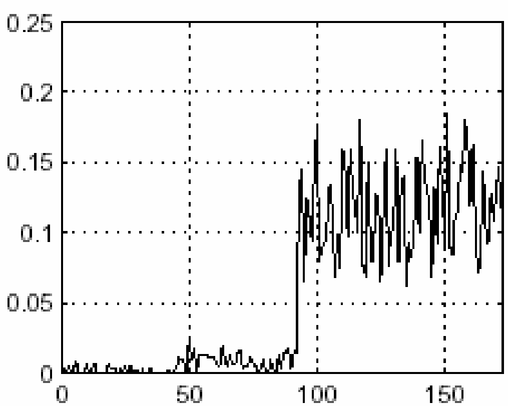
3D motion segmentation results



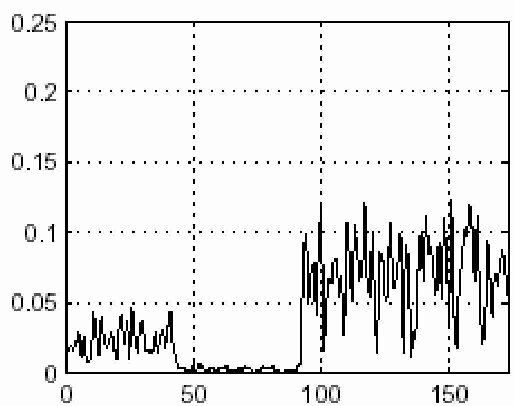
(a) First frame



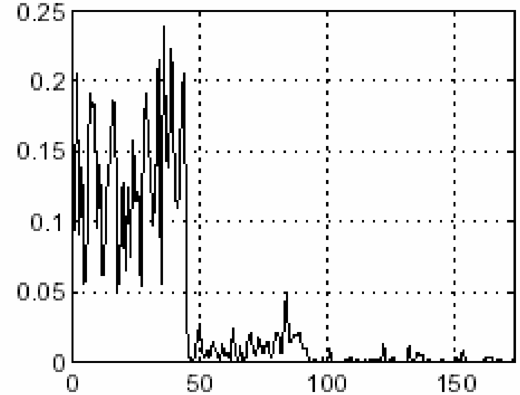
(b) Feature segmentation



(c) $E_1(\hat{F}_1)$



(d) $E_1(\hat{F}_2)$



(e) $E_1(\hat{F}_3)$

Other cases: linearly moving objects

$$x_2^T \widehat{e} x_1 = e^T(\widehat{x_2} x_1) = 0$$

$$\ell = \widehat{x_2} x_1, \qquad e^T \ell = 0$$

- Multibody epipole
- Recovery of epipoles
- Fundamental matrices
- Feature segmentation

$$\tilde{e}^T \nu_n(\ell) = 0$$

$$\tilde{e} \mapsto \{e_i\}_{i=1}^n$$

$$F_i = \widehat{e_i}$$

$$x_2^T \widehat{e_i} x_1 = 0$$

Minimum number of points

n	1	2	3	4
N	8	35	99	225

n	1	2	5	10
N	2	5	20	65

Other cases: affine flows

- In linear motions, geometric constraints are linear

$$\boldsymbol{b}_1^T\boldsymbol{x} = 0 \vee \cdots \vee \boldsymbol{b}_n^T\boldsymbol{x} = 0 \Leftrightarrow (\boldsymbol{b}_1^T\boldsymbol{x})\cdots(\boldsymbol{b}_n^T\boldsymbol{x}) = 0$$

- Two-view motion constraints could be bilinear!!!

Affine motion segmentation: constant brightness constraint

$$\begin{bmatrix} I_x & I_y & I_t \end{bmatrix} A \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0$$

$$\boldsymbol{y}^T \quad A \quad \boldsymbol{x} \quad = 0$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{3\times 3}$$

3D motion segmentation: epipolar constraint

$$\begin{bmatrix} x_2 & y_2 & 1 \end{bmatrix} F \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = 0$$

$$\boldsymbol{x}_2^T \quad F \quad \boldsymbol{x}_1 \quad = 0$$

$$F = \underbrace{\hat{T} \qquad R}_{so(3)\times SO(3)\subset\mathbb{R}^{3\times 3}}$$

Multiple affine views with missing data

- Affine camera model $\boldsymbol{x}_{fp} = \mathtt{A}_f \boldsymbol{X}_p$

- Motion of 1 rigid-body lives in a subspace of dimension 4

$$\mathtt{W} = \mathtt{M}\mathtt{S}^\top$$

$$\begin{bmatrix} \boldsymbol{x}_{11} \cdots \boldsymbol{x}_{1P} \\ \vdots \qquad \vdots \\ \boldsymbol{x}_{F1} \cdots \boldsymbol{x}_{FP} \end{bmatrix}_{2F\times P} = \begin{bmatrix} \mathtt{A}_1 \\ \vdots \\ \mathtt{A}_F \end{bmatrix}_{2F\times 4} \begin{bmatrix} \boldsymbol{X}_1 \cdots \boldsymbol{X}_P \end{bmatrix}_{4\times P}$$

- Motion segmentation is equivalent to clustering subspaces of dimension 2,3,4 in R^{2F}
  - Project to 5-D subspace: Power Factorization
  - Estimate multiple subspaces in R^5: GPCA

Mutiframe results

Misclassification error for different sequences.

Sequence	Points	Frames	Motions	Error
Boat	686	11	2	2.19%
Shirt-Book	170	3	2	1.18%
Wilshire	200	3	2	5.50%
Tea-Tins	84	3	2	1.19%
NEC	82	8	2	0.00%
3-Cars	173	15	3	4.62%
Puma	64	16	2	0.00%
Castle	56	11	2	0.00%

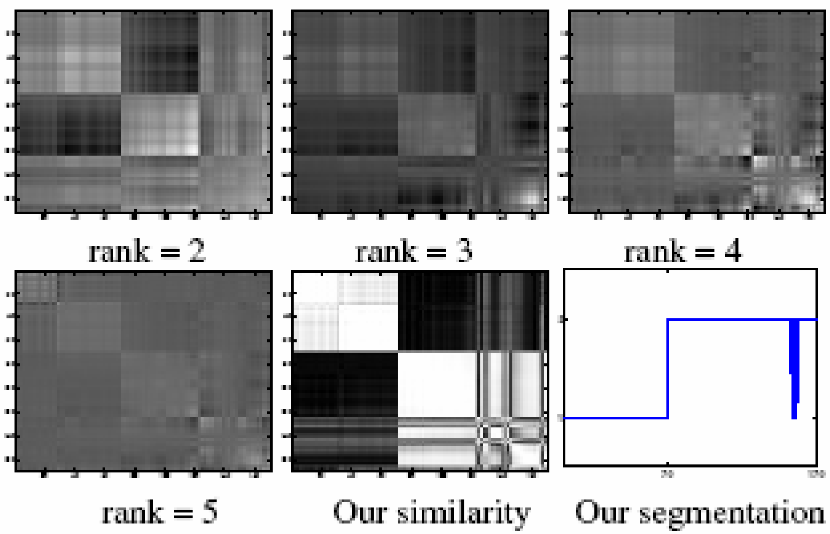


Figure 2: Similarity/Interaction matrices from the Costeira and Kanade algorithm for different rank approximations and from our algorithm for the Can-Book sequence.

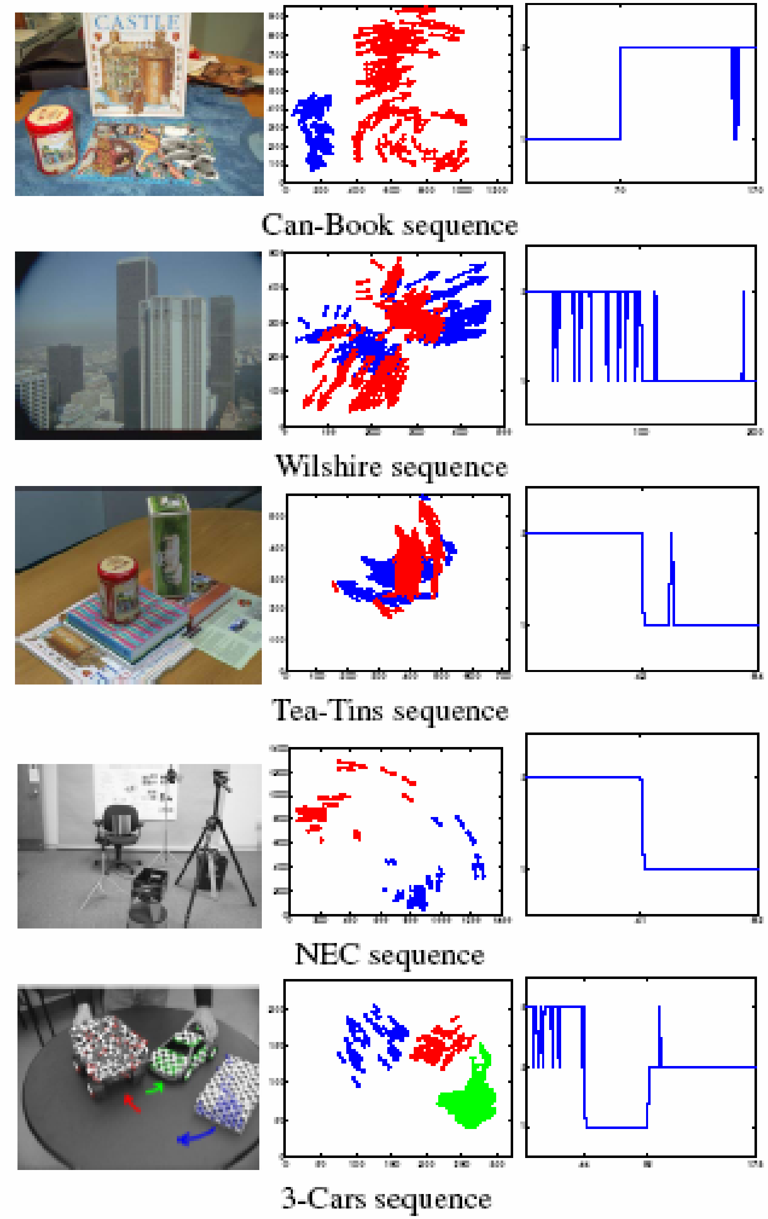


Figure 1: Motion segmentation results. Left: first frame of each sequence. Center: displacement of the correspondences between two views. Right: clustering of the correspondences given by our algorithm.

Conclusions

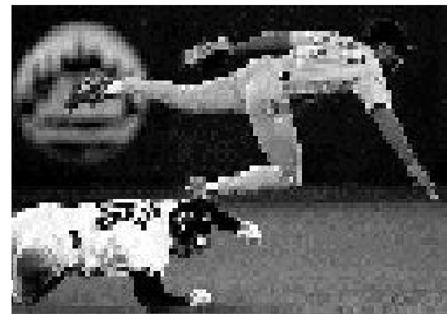
- There is an analytic solution to 3-D motion segmentation based on
  - Fit a polynomial to all the image data
  - Differentiate polynomial to obtain motion parameters
- Applies to most motion models in computer vision

Table 1. 2-D and 3-D motion models considered in this paper.

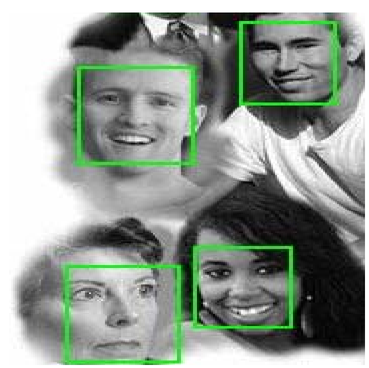
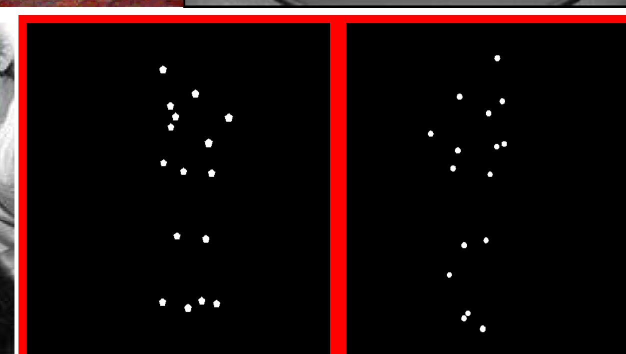
Motion models	Model equations	Model parameters	Equivalent to clustering
2-D translational	$\boldsymbol{x}_2 = \boldsymbol{x}_1 + T_i$	$\{T_i \in \mathbb{R}^2\}_{i=1}^n$	Hyperplanes in $\mathbb{C}^2$
2-D similarity	$\boldsymbol{x}_2 = \lambda_i R_i \boldsymbol{x}_1 + T_i$	$\{(R_i, T_i) \in SE(2), \lambda_i \in \mathbb{R}^+\}_{i=1}^n$	Hyperplanes in $\mathbb{C}^3$
2-D affine	$\boldsymbol{x}_2 = A_i \begin{bmatrix} \boldsymbol{x}_1 \\ 1 \end{bmatrix}$	$\{A_i \in \mathbb{R}^{2\times 3}\}_{i=1}^n$	Hyperplanes in $\mathbb{C}^4$
3-D translational	$0 = \boldsymbol{x}_2^T [T_i]_\times \boldsymbol{x}_1$	$\{T_i \in \mathbb{R}^3\}_{i=1}^n$	Hyperplanes in $\mathbb{R}^3$
3-D rigid-body	$0 = \boldsymbol{x}_2^T F_i \boldsymbol{x}_1$	$\{F_i \in \mathbb{R}^{3\times 3} : \text{rank}(F_i) = 2\}_{i=1}^n$	Bilinear forms in $\mathbb{R}^{3\times 3}$
3-D homography	$\boldsymbol{x}_2 \sim H_i \boldsymbol{x}_1$	$\{H_i \in \mathbb{R}^{3\times 3}\}_{i=1}^n$	Bilinear forms in $\mathbb{C}^{2\times 3}$

Some possible applications of GPCA

- Geometry
  - Vanishing points
- Segmentation
  - Intensity
  - Texture
  - 2D Motion
  - 3D Motion
- Recognition
  - Faces (Eigenfaces)
    - Man - Woman
  - Human activities
    - Running, walking
- Image and video compression



References

- R. Vidal, Y. Ma, S. Soatto and S. Sastry. *Two-view multibody structure from motion*, International Journal of Computer Vision, 2004
- R. Vidal and Y. Ma. *A unified algebraic approach to 2-D and 3-D motion segmentation*, European Conf. on Computer Vision 2004
- R. Vidal and R. Hartley. *Motion segmentation using Power Factorization and GPCA*, IEEE Conference on Computer Vision and Pattern Recognition 2004
- R. Vidal and S. Sastry. *Optimal segmentation of dynamic scenes from two perspective views,* International Conference on Computer Vision and Pattern Recognition, 2003
- R. Vidal and S. Sastry. *Segmentation of dynamic scenes from image intensities,* IEEE Workshop on Vision and Motion Computing, 2002.

PRIMARY REFERENCE