Segmentation of Subspace Arrangements – Robust GPCA

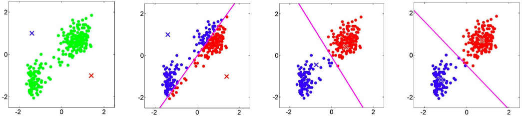
Allen Y. Yang <yang@eecs.berkeley.edu>

Workshop on GPCA, Dec. 11, 2007

1. GPCA-Voting
   - Noise issue
   - GPCA-Voting
   - Comparison
2. Robust GPCA
   - Outlier Issue
   - Robustify GPCA via MVT and Influence
   - Comparison
3. Applications
   - Affine Motion Detection
   - Vanishing-Point Detection

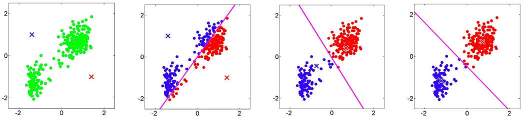
Usual Suspects in Statistics

- K-Means for segmenting K Gaussian clusters:



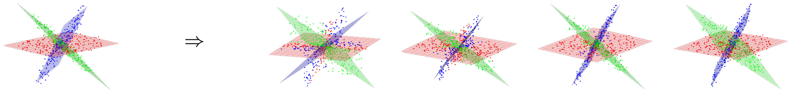
Usual Suspects in Statistics

- K-Means for segmenting K Gaussian clusters:



- K-Subspaces for subspace arrangements [Ho et al., 2003]:

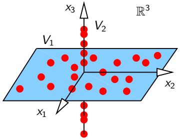
1. Initialization: Set initial values of orthogonal matrices $\hat{U}_i^{(0)} \in \mathbb{R}^{D \times d_i}$ for $i = 1, \ldots, N$. Let $m = 0$.
2. Segmentation: For each sample $\mathbf{z}_k$, assign it to group $\hat{X}_i^{(m)}$ if
$$i = \arg\min_l \|\mathbf{z}_k - \hat{U}_l^{(m)}(\hat{U}_l^{(m)})^T \mathbf{z}_k\|^2.$$
3. Estimation: Apply PCA to each subset $\hat{X}_i^{(m)}$ and obtain new estimates for the subspace bases $\hat{U}_i^{(m+1)}$.
4. Let $m \leftarrow m + 1$, and repeat step 2 and 3 until the segmentation does not change.



Generalized Principal Component Analysis (PDA)

- $\mathbf{x} \in V_1 \cup V_2 \Rightarrow (x_3 = 0) or (x_1 = x_2 = 0)$
  $\Rightarrow \{x_1x_3 = 0, x_2x_3 = 0\}$.

$$L_2 \doteq [\nu_2(\mathbf{x}_1), \ldots, \nu_2(\mathbf{x}_N)] \in \mathbb{R}^{M_2^{[3]} \times N} = \begin{bmatrix} \cdots & (x_1)^2 & \cdots \\ \cdots & (x_1x_2) & \cdots \\ \cdots & (x_1x_3) & \cdots \\ \cdots & (x_2)^2 & \cdots \\ \cdots & (x_2x_3) & \cdots \\ \cdots & (x_3)^2 & \cdots \end{bmatrix}$$

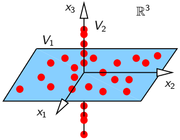


- The null space of L_2 is $\begin{array}{l} \mathbf{c}_1 = [0,0,1,0,0,0] \\ \mathbf{c}_2 = [0,0,0,0,1,0] \end{array} \Rightarrow \begin{array}{l} p_1 = \mathbf{c}_1\nu_2(\mathbf{x}) = x_1x_3 \\ p_2 = \mathbf{c}_2\nu_2(\mathbf{x}) = x_2x_3 \end{array}$
- $P(\mathbf{x}) \doteq [p_1(\mathbf{x}) \ p_2(\mathbf{x})] = [x_1x_3, x_2x_3]$, then
  $\nabla_{\mathbf{x}} P = [\,\nabla_{\mathbf{x}} p_1 \ \nabla_{\mathbf{x}} p_2\,] = \begin{bmatrix} x_3 & 0 \\ 0 & x_3 \\ x_1 & x_2 \end{bmatrix}$.
- $\nabla_{\mathbf{x}} P$ at one sample per subspace gives normal vectors that span $V_1^\perp$ and $V_2^\perp$.

Generalized Principal Component Analysis (PDA)

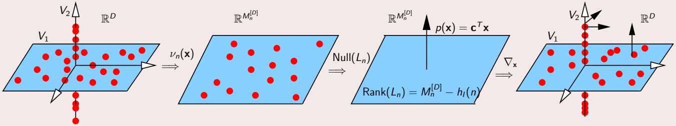
- $\mathbf{x} \in V_1 \cup V_2 \Rightarrow (x_3 = 0) or (x_1 = x_2 = 0)$
  $\Rightarrow \{x_1x_3 = 0, x_2x_3 = 0\}$.

$$L_2 \doteq [\nu_2(\mathbf{x}_1), \ldots, \nu_2(\mathbf{x}_N)] \in \mathbb{R}^{M_2^{[3]} \times N} = \begin{bmatrix} \cdots & (x_1)^2 & \cdots \\ \cdots & (x_1x_2) & \cdots \\ \cdots & (x_1x_3) & \cdots \\ \cdots & (x_2)^2 & \cdots \\ \cdots & (x_2x_3) & \cdots \\ \cdots & (x_3)^2 & \cdots \end{bmatrix}$$



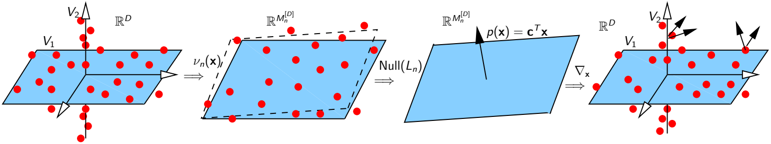
- The null space of L_2 is $\begin{matrix} \mathbf{c}_1 = [0, 0, 1, 0, 0, 0] \\ \mathbf{c}_2 = [0, 0, 0, 0, 1, 0] \end{matrix} \Rightarrow \begin{matrix} p_1 = \mathbf{c}_1\nu_2(\mathbf{x}) = x_1x_3 \\ p_2 = \mathbf{c}_2\nu_2(\mathbf{x}) = x_2x_3 \end{matrix}$
- $P(\mathbf{x}) \doteq [p_1(\mathbf{x}) \; p_2(\mathbf{x})] = [x_1x_3, x_2x_3]$, then
  $\nabla_{\mathbf{x}} P = [\, \nabla_{\mathbf{x}} p_1 \;\; \nabla_{\mathbf{x}} p_2 \,] = \begin{bmatrix} x_3 & 0 \\ 0 & x_3 \\ x_1 & x_2 \end{bmatrix}$.
- $\nabla_{\mathbf{x}} P$ at one sample per subspace gives normal vectors that span $V_1^\perp$ and $V_2^\perp$.

Diagram of GPCA



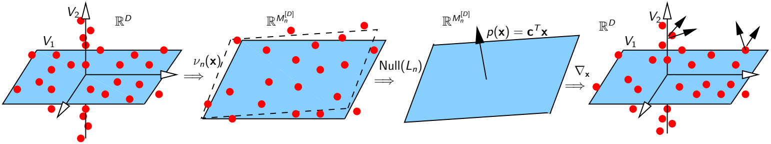
GPCA-Voting: A Stable Implementation

PDA on noisy data



GPCA-Voting: A Stable Implementation

PDA on noisy data

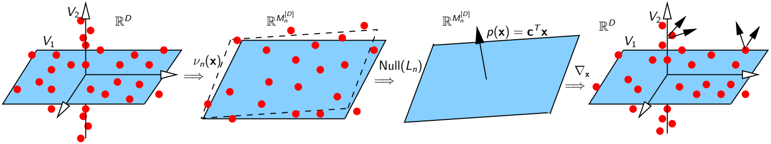


The noise affects the algebraic PDA process:

1. The data matrix $L_K(V)$ is always *full-rank*.
   Question: How many linearly independent vanishing polynomials from Null(L)?

GPCA-Voting: A Stable Implementation

PDA on noisy data



The noise affects the algebraic PDA process:

1. The data matrix $L_K(V)$ is always *full-rank*.
   Question: How many linearly independent vanishing polynomials from Null(L)?
2. How to choose (more than) one point per subspace for derivative evaluation?

Number of Linearly Independent Polynomials

1. Given a mixture of K subspaces, how many linearly independent Kth degree vanishing polynomials?
   - Trivial: linear products of 1-forms $p_1 = x_1 x_3$, $p_2 = x_2 x_3$ uniquely determine $V_1 \cup V_2$.
   - Not trivial: given $V_1 \cup V_2$, $p_1 = x_1 x_3$, $p_2 = x_2 x_3$ are a **max** set of linearly independent vanishing polynomials.

Number of Linearly Independent Polynomials

1. Given a mixture of K subspaces, how many linearly independent Kth degree vanishing polynomials?
   - Trivial: linear products of 1-forms $p_1 = x_1x_3, p_2 = x_2x_3$ uniquely determine $V_1 \cup V_2$.
   - Not trivial: given $V_1 \cup V_2$, $p_1 = x_1x_3, p_2 = x_2x_3$ are a **max** set of linearly independent vanishing polynomials.
2. Under a general position condition, the number is **combinatorial invariant** [Jessica Sidman 2002 & Harm Derksen 2005]

$$h(K) = \sum_S (-1)^{|S|} \binom{K+D-1-c_S}{D-1-c_S},$$

where $S \subseteq \{1, \cdots, K\}$ is an index set.

Number of Linearly Independent Polynomials

1. Given a mixture of K subspaces, how many linearly independent Kth degree vanishing polynomials?
   - Trivial: linear products of 1-forms $p_1 = x_1 x_3$, $p_2 = x_2 x_3$ uniquely determine $V_1 \cup V_2$.
   - Not trivial: given $V_1 \cup V_2$, $p_1 = x_1 x_3$, $p_2 = x_2 x_3$ are a **max** set of linearly independent vanishing polynomials.
2. Under a general position condition, the number is **combinatorial invariant** [Jessica Sidman 2002 & Harm Derksen 2005]

$$h(K) = \sum_S (-1)^{|S|} \binom{K+D-1-c_S}{D-1-c_S},$$

   where $S \subseteq \{1, \cdots, K\}$ is an index set.
3. Example: linearly independent 3rd degree vanishing polynomials for 3 mixture subspaces.

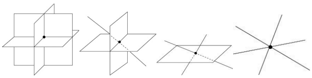


Figure: Four possible configurations in $\mathbb{R}^3$.

d_1	d_2	d_3	$h(3)$
2	2	2	1
2	2	1	2
2	1	1	4
1	1	1	7

Number of Linearly Independent Polynomials

1. Given a mixture of K subspaces, how many linearly independent Kth degree vanishing polynomials?
   - Trivial: linear products of 1-forms $p_1 = x_1 x_3, p_2 = x_2 x_3$ uniquely determine $V_1 \cup V_2$.
   - Not trivial: given $V_1 \cup V_2$, $p_1 = x_1 x_3, p_2 = x_2 x_3$ are a **max** set of linearly independent vanishing polynomials.
2. Under a general position condition, the number is **combinatorial invariant** [Jessica Sidman 2002 & Harm Derksen 2005]

$$h(K) = \sum_S (-1)^{|S|} \binom{K+D-1-c_S}{D-1-c_S},$$

   where $S \subseteq \{1, \cdots, K\}$ is an index set.
3. Example: linearly independent 3rd degree vanishing polynomials for 3 mixture subspaces.

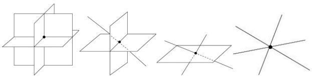


Figure: Four possible configurations in $\mathbb{R}^3$.

d_1	d_2	d_3	$h(3)$
2	2	2	1
2	2	1	2
2	1	1	4
1	1	1	7

Polynomial Estimation from Noisy Data

Given $d_1, \cdots, d_K$, use **SVD** to recover $h(K)$ vanishing polynomials from the smallest eigenspace of L.

A Voting Scheme

- Goal:
  1. Averaging $\nabla_{\mathbf{x}} P$ at more samples of a subspace.
  2. Recover correct rank of $\nabla_{\mathbf{x}} P$.

A Voting Scheme

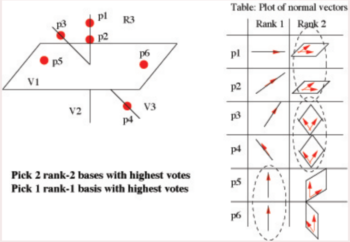
- Goal:
  1. Averaging $\nabla_{\mathbf{x}} P$ at more samples of a subspace.
  2. Recover correct rank of $\nabla_{\mathbf{x}} P$.
- Difficulty: Do not know which samples belong to the same subspace, yet.

A Voting Scheme

- Goal:
  1. Averaging $\nabla_{\mathbf{x}} P$ at more samples of a subspace.
  2. Recover correct rank of $\nabla_{\mathbf{x}} P$.
- Difficulty: Do not know which samples belong to the same subspace, yet.

GPCA-Voting (a simple example)

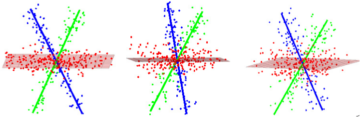
1. Assume subspaces $(2,1,1)$ in $\mathbb{R}^3$.
2. $h_I(3) = 4$ vanishing polynomials $\Rightarrow \nabla_{\mathbf{x}} P \in \mathbb{R}^{3\times 4}$.
3. Vote on rank-1 & rank-2 codimensions with a tolerance threshold τ



4. Average normal vectors associated with highest votes.
5. (optional) Iteratively refine the segmentation via EM or K-Subspaces.

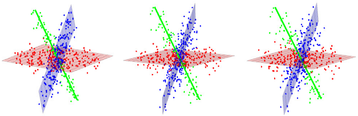
Simulation Results

1. Illustrations



(a) 8% (b) 12% (c) 16%

Figure: $(2, 1, 1) \in \mathbb{R}^3$.

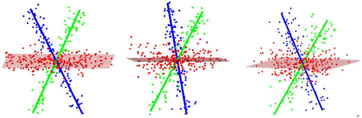


(a) 8% (b) 12% (c) 16%

Figure: $(2, 2, 1) \in \mathbb{R}^3$.

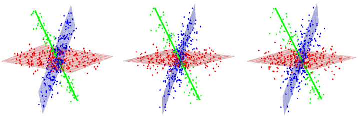
Simulation Results

1. Illustrations



(a) 8% (b) 12% (c) 16%

Figure: $(2, 1, 1) \in \mathbb{R}^3$.



(a) 8% (b) 12% (c) 16%

Figure: $(2, 2, 1) \in \mathbb{R}^3$.

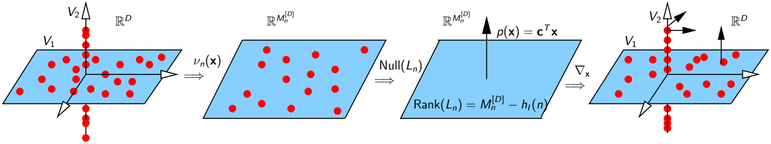
2. Segmentation simulations

Table: Segmentation errors. 4% Gaussian noise is added.

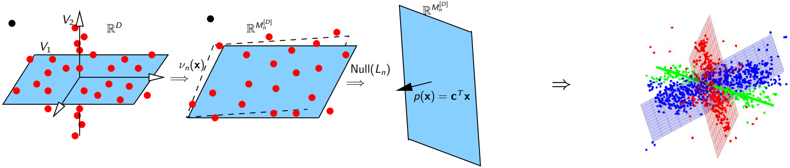
Subspace Dimensions	K-Subspaces	PDA	Voting	Voting+K-Subspaces
$(2, 2, 1)$ in $\mathbb{R}^3$	27%	13.2%	6.4%	5.4%
$(4, 2, 2, 1)$ in $\mathbb{R}^5$	57%	39.8%	5.7%	5.7%
$(4, 4, 4, 4)$ in $\mathbb{R}^5$	25%	25.3%	17%	11%

Outlier Issue

- GPCA process:

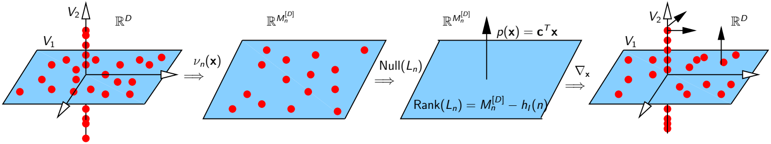


- Breakdown of GPCA is 0%: a large outlier can arbitrarily perturb $\mathrm{Null}(L_n)$

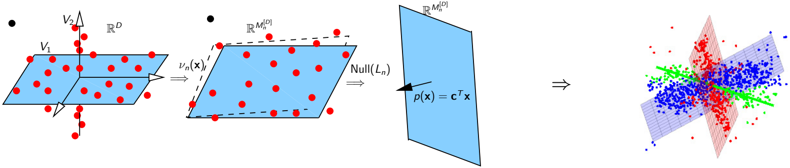


Outlier Issue

- GPCA process:



- Breakdown of GPCA is 0%: a large outlier can arbitrarily perturb $\mathrm{Null}(L_n)$

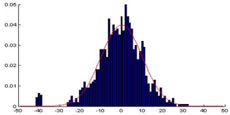


- Because breakdown of PCA is 0% $\Rightarrow$ Seek a robust PCA to estimate $\mathrm{Null}(L_n)$, where $L_n = [\nu_n(\mathbf{x}_1), \cdots, \nu_n(\mathbf{x}_N)]$.

Three approaches to tackle outliers:

1. Probability-based: small-probability samples.

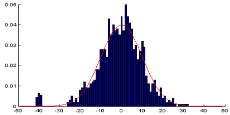
   Probability plots: [Healy 1968, Cox 1968]
   PCs: [Rao 1964, Ganadesikan & Kettenring 1972]
   M-estimators: [Huber 1981, Campbell 1980]
   multivariate trimming (MVT): [Ganadesikan & Kettenring 1972]



Three approaches to tackle outliers:

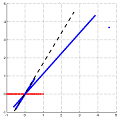
1. Probability-based: small-probability samples.

   Probability plots: [Healy 1968, Cox 1968]
   PCs: [Rao 1964, Ganadesikan & Kettenring 1972]
   M-estimators: [Huber 1981, Campbell 1980]
   multivariate trimming (MVT): [Ganadesikan & Kettenring 1972]



2. Influence-based: large influence on model parameters.

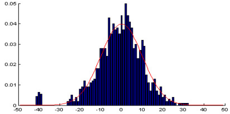
   Parameter difference with and without a sample: [Hampel et al. 1986, Critchley 1985]



Three approaches to tackle outliers:

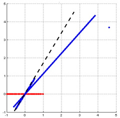
1. Probability-based: small-probability samples.

   Probability plots: [Healy 1968, Cox 1968]
   PCs: [Rao 1964, Ganadesikan & Kettenring 1972]
   M-estimators: [Huber 1981, Campbell 1980]
   multivariate trimming (MVT): [Ganadesikan & Kettenring 1972]



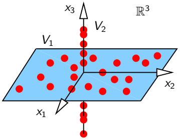
2. Influence-based: large influence on model parameters.

   Parameter difference with and without a sample: [Hampel et al. 1986, Critchley 1985]



3. Consensus-based: not consistent with models of high consensus.

   Hough: [Ballard 1981, Lowe 1999]
   RANSAC: [Fischler & Bolles 1981, Torr 1997]
   Least Median Estimate (LME): [Rousseeuw 1984, Steward 1999]



Robust GPCA

STEP 1: Given the outlier percentage $\alpha\%$, robustify PCA:

- Multivariate-trimming (MVT):
  Assuming a Gaussian distribution, samples with large *Mahalanobis* distance more likely to be outliers.

1. Compute a robust mean $\bar{u}$. $\mathbf{v}_i = \mathbf{u}_i - \bar{u}$. $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{R}^{M_n^{[D]}}$
2. Initialize $\Sigma_0 = I_{M_n^{[D]} \times M_n^{[D]}}$.
3. In kth iteration, sort $\mathbf{v}_1, \ldots, \mathbf{v}_N$ by the *Mahalanobis* distance:
$$d_i = \mathbf{v}_i^T \Sigma_{k-1}^{-1} \mathbf{v}_i.$$
4. Update Σ_k from $(100-\alpha)\%$ samples with smallest distances.
5. Iteration stops when $\|\Sigma_{k-1} - \Sigma_k\|$ is small.

Robust GPCA

STEP 1: Given the outlier percentage $\alpha\%$, robustify PCA:

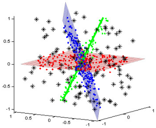
- Multivariate-trimming (MVT):
  Assuming a Gaussian distribution, samples with large *Mahalanobis* distance more likely to be outliers.

  1. Compute a robust mean $\bar{u}$. $\mathbf{v}_i = \mathbf{u}_i - \bar{u}$. $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{R}^{M_n^{[D]}}$
  2. Initialize $\Sigma_0 = I_{M_n^{[D]} \times M_n^{[D]}}$.
  3. In kth iteration, sort $\mathbf{v}_1, \dots, \mathbf{v}_N$ by the *Mahalanobis* distance:
  $$d_i = \mathbf{v}_i^T \Sigma_{k-1}^{-1} \mathbf{v}_i.$$
  4. Update Σ_k from $(100 - \alpha)\%$ samples with smallest distances.
  5. Iteration stops when $\|\Sigma_{k-1} - \Sigma_k\|$ is small.

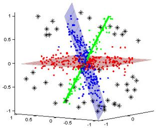
- Influence function:

  1. Compute null space $C = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m\}$ for $L_n = [\nu_n(\mathbf{x}_1) \cdots \nu_n(\mathbf{x}_N)]$.
  2. For $\mathbf{x}_i$, compute $C^{(i)}$ for $L_n^{(i)} = [\nu_n(\mathbf{x}_1) \cdots \hat{i} \cdots \nu_n(\mathbf{x}_N)]$.
  3. $I(\mathbf{x}_i) \doteq \langle C, C^{(i)} \rangle$.
  4. Reject top $\alpha\%$ samples with highest influence.

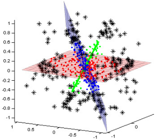
STEP 2: Estimating the outlier percentage $\alpha\%$: Do we need the exact percentage for estimation?



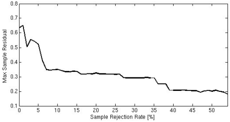
(a) 16% outliers



(b) 7% rejected

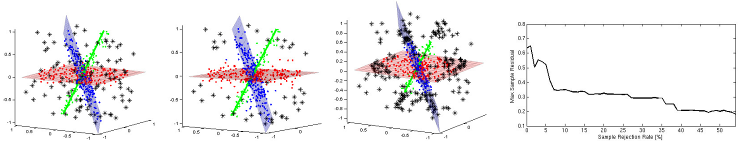


(c) 38% rejected



(d) Maximal sample residuals.

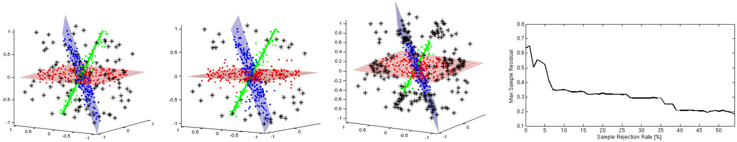
STEP 2: Estimating the outlier percentage $\alpha\%$: Do we need the exact percentage for estimation?



(a) 16% outliers (b) 7% rejected (c) 38% rejected (d) Maximal sample residuals.

1. With noise and outliers present, unnecessary to distinguish outliers close to subspaces.
2. Robust PCA is moderately stable when the outlier percentage is over-estimated.

STEP 2: Estimating the outlier percentage $\alpha\%$: Do we need the exact percentage for estimation?



(a) 16% outliers (b) 7% rejected (c) 38% rejected (d) Maximal sample residuals.

1. With noise and outliers present, unnecessary to distinguish outliers close to subspaces.
2. Robust PCA is moderately stable when the outlier percentage is over-estimated.

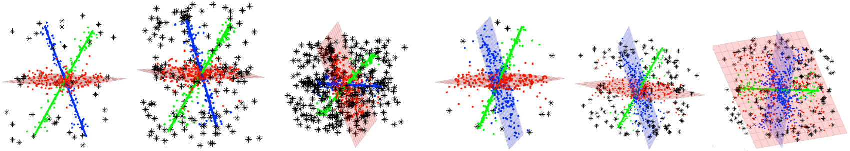
Outlier Percentage Test based on the Influence Function Principle

Further rejection only results in small changes in the model parameters and sample residuals (w.r.t. boundary threshold σ), i.e., the arrangement model stabilizes.

Influence

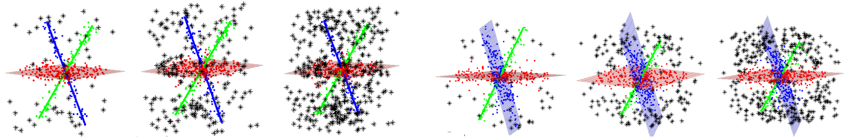
Simulations on Robust GPCA

- RGPCA-Influence



(a) 12% (b) 32% (c) 48% (d) 12% (e) 32% (f) 48%

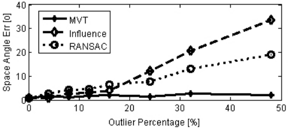
- RGPCA-MVT



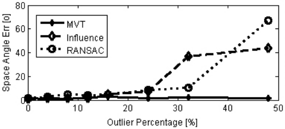
(a) 12% (b) 32% (c) 48% (d) 12% (e) 32% (f) 48%

Comparison with RANSAC

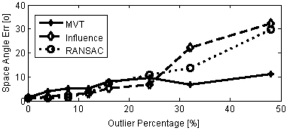
- Accuracy on simulated data



(a) $(2,2,1)$ in $\mathbb{R}^3$



(b) $(4,2,2,1)$ in $\mathbb{R}^5$



(c) $(5,5,5)$ in $\mathbb{R}^6$

- Speed

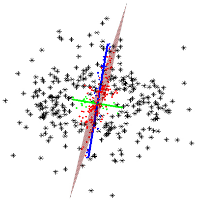
Table: Average time of RANSAC and RGPCA with 24% outliers.

Arrangement	(2,2,1) in $\mathbb{R}^3$	(4,2,2,1) in $\mathbb{R}^5$	(5,5,5) in $\mathbb{R}^6$
RANSAC	44s	5.1m	3.4m
MVT	46s	23m	8m
Influence	3m	58m	146m

Limitations of RGPCA

1. Hardware limits for high subspace dimension (> 10) or subspace number (> 6) in MATLAB.
2. Need to know the number of subspaces and dimensions.
3. Overfitting when percentage is overestimated, especially for MVT.

Animation $\Longrightarrow$



Motion Segmentation under 3-D Affine Projection

- Problem formulation:

1. Object features $\mathbf{p}_1, \ldots, \mathbf{p}_N \in \mathbb{R}^3$ are tracked in F frames.
2. Denote $\mathbf{m}_{ij} \in \mathbb{R}^2$ as the image under 3-D affine projection:

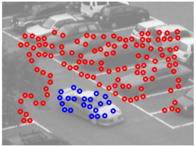
$$\mathbf{m}_{ij} = A_j \mathbf{p}_i + \mathbf{b}_j \in \mathbb{R}^2, \quad i = 1, \ldots, N; j = 1, \ldots, F.$$

3. For each p_i,

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{m}_{i1} \\ \vdots \\ \mathbf{m}_{iF} \end{bmatrix} \in \mathbb{R}^{2F}, \quad i = 1, \ldots, N.$$

4. Segment $\mathbf{x}_1, \ldots, \mathbf{x}_N$ that belong to different motions.

parking-lot movie



Suppose $\mathbf{p}_1, \ldots, \mathbf{p}_N \in \mathbb{R}^3$ are from a single object:

- Stack corresponding images in F frames: $\mathbf{x}_i = \begin{bmatrix} \mathbf{m}_{i1} \\ \mathbf{m}_{i2} \\ \vdots \\ \mathbf{m}_{iF} \end{bmatrix} \in \mathbb{R}^{2F}$, $i = 1, \cdots, N$:

$$W \doteq [\mathbf{x}_1 \cdots \mathbf{x}_N]_{2F\times N} = \begin{bmatrix} A_1 & \mathbf{b}_1 \\ \vdots & \vdots \\ A_F & \mathbf{b}_F \end{bmatrix}_{2F\times 4} \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_N \\ 1 & \cdots & 1 \end{bmatrix}_{4\times N}.$$

$\Rightarrow \mathbf{x}_i \in \mathbb{R}^{2F}$ lives in a subspace of dimension 4.

Suppose $\mathbf{p}_1, \ldots, \mathbf{p}_N \in \mathbb{R}^3$ are from a single object:

- Stack corresponding images in F frames: $\mathbf{x}_i = \begin{bmatrix} \mathbf{m}_{i1} \\ \mathbf{m}_{i2} \\ \vdots \\ \mathbf{m}_{iF} \end{bmatrix} \in \mathbb{R}^{2F}$, $i = 1, \cdots, N$:

$$W \doteq [\mathbf{x}_1 \cdots \mathbf{x}_N]_{2F \times N} = \begin{bmatrix} A_1 & \mathbf{b}_1 \\ \vdots & \vdots \\ A_F & \mathbf{b}_F \end{bmatrix}_{2F \times 4} \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_N \\ 1 & \cdots & 1 \end{bmatrix}_{4 \times N}.$$

$\Rightarrow \mathbf{x}_i \in \mathbb{R}^{2F}$ lives in a subspace of dimension 4.

- When all $\mathbf{p}_i$'s are coplanar, there exists a world coordinate system such that $\mathbf{p}_i = \begin{bmatrix} x_i \\ y_i \\ 0 \end{bmatrix}$.

$$W \doteq [\mathbf{x}_1 \cdots \mathbf{x}_N]_{2F \times N} = \begin{bmatrix} A_1 & \mathbf{b}_1 \\ \vdots & \vdots \\ A_F & \mathbf{b}_F \end{bmatrix}_{2F \times 4} \begin{bmatrix} x_1 & \cdots & x_N \\ y_1 & \cdots & y_N \\ 0 & \cdots & 0 \\ 1 & \cdots & 1 \end{bmatrix}_{4 \times N}.$$

$\Rightarrow \mathbf{x}_i \in \mathbb{R}^{2F}$ lives in a subspace of dimension 3.

Suppose $\mathbf{p}_1, \ldots, \mathbf{p}_N \in \mathbb{R}^3$ are from a single object:

- Stack corresponding images in F frames: $\mathbf{x}_i = \begin{bmatrix} \mathbf{m}_{i1} \\ \mathbf{m}_{i2} \\ \vdots \\ \mathbf{m}_{iF} \end{bmatrix} \in \mathbb{R}^{2F}$, $i = 1, \cdots, N$:

$$W \doteq [\mathbf{x}_1 \cdots \mathbf{x}_N]_{2F \times N} = \begin{bmatrix} A_1 & \mathbf{b}_1 \\ \vdots & \vdots \\ A_F & \mathbf{b}_F \end{bmatrix}_{2F \times 4} \begin{bmatrix} \mathbf{p}_1 & \cdots & \mathbf{p}_N \\ 1 & \cdots & 1 \end{bmatrix}_{4 \times N}.$$

$\Rightarrow \mathbf{x}_i \in \mathbb{R}^{2F}$ lives in a subspace of dimension 4.

- When all $\mathbf{p}_i$'s are coplanar, there exists a world coordinate system such that $\mathbf{p}_i = \begin{bmatrix} x_i \\ y_i \\ 0 \end{bmatrix}$.

$$W \doteq [\mathbf{x}_1 \cdots \mathbf{x}_N]_{2F \times N} = \begin{bmatrix} A_1 & \mathbf{b}_1 \\ \vdots & \vdots \\ A_F & \mathbf{b}_F \end{bmatrix}_{2F \times 4} \begin{bmatrix} x_1 & \cdots & x_N \\ y_1 & \cdots & y_N \\ 0 & \cdots & 0 \\ 1 & \cdots & 1 \end{bmatrix}_{4 \times N}.$$

$\Rightarrow \mathbf{x}_i \in \mathbb{R}^{2F}$ lives in a subspace of dimension 3.

Multiple rigid bodies under affine projection

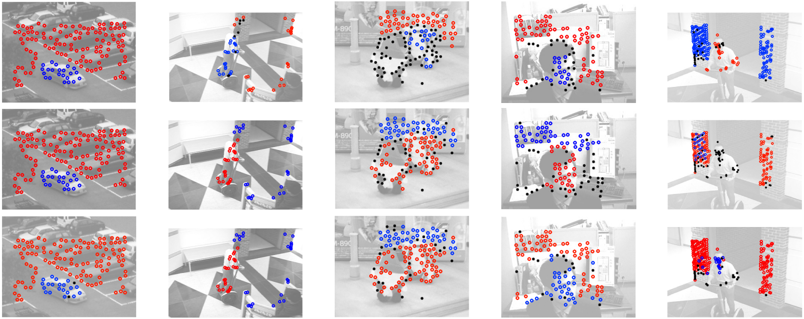
Segmenting multiple rigid bodies under affine camera projection is equivalent to segmenting multiple subspaces of dimension 3 or 4.

Sequences: parking-lot | segway | toys | man | segway3

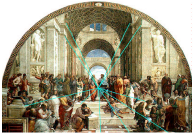
RANSAC:

MVT:

Influence:



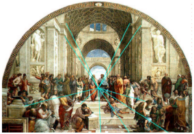
Vanishing-Point Detection



1. Perspective projection of 2 parallel lines in space intersect at *vanishing point* in image plane.

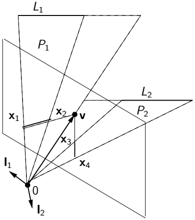
Vanishing-Point Detection

1. Perspective projection of 2 parallel lines in space intersect at *vanishing point* in image plane.



2. Geometry of a family of parallel lines:
   - The co-image of a line L_1 is $\mathbf{l}_1 = \mathbf{x}_1 \times \mathbf{x}_2$.
   - Given images of two parallel lines $(\mathbf{x}_1, \mathbf{x}_2)$ and $(\mathbf{x}_3, \mathbf{x}_4)$, $(0, \mathbf{v})$ is on the intersection of P_1 and P_2.

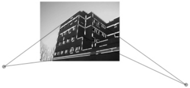
   $\Rightarrow$ Any co-image $\mathbf{l}_i$ of a line parallel to L_1 satisfies: $\mathbf{l}_i \perp \mathbf{v}$.



3. Multiple vanishing points:
   - Multiple families of parallel lines correspond to multiple vanishing points $\mathbf{v}_1, \dots, \mathbf{v}_n$.
   - Any co-image of a line in the families must satisfy

$$(\mathbf{l}^T \mathbf{v}_1)(\mathbf{l}^T \mathbf{v}_2) \cdots (\mathbf{l}^T \mathbf{v}_n) = 0.$$

   $\Rightarrow$ Segmenting parallel line families is equivalent to segmenting 2-D subspaces in $\mathbb{R}^3$.

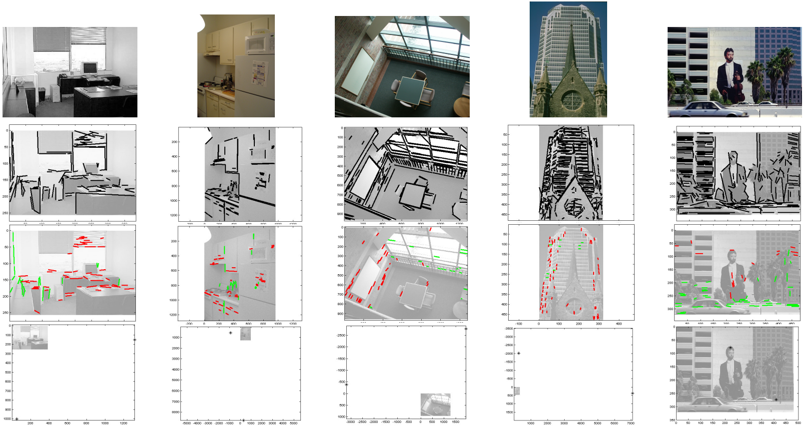


RGPCA-Influence

Images:

Segments:

Influence:

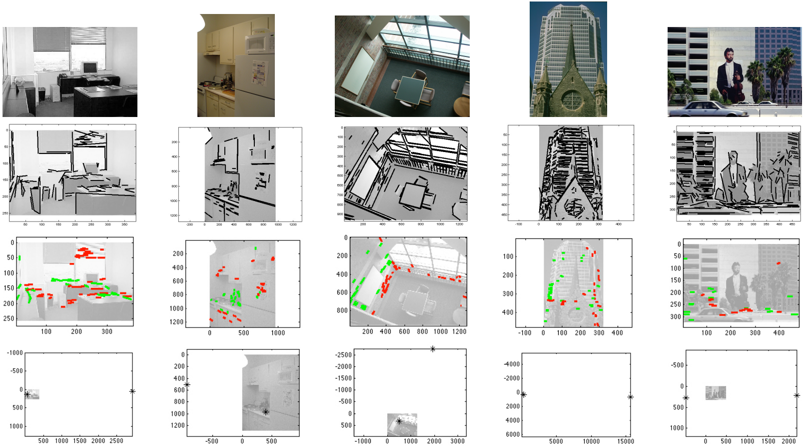


RGPCA-MVT

Images:

Segments:

MVT:



RANSAC-on-Subspaces

Images:

Segments:

RANSAC: