Probabilistic Point Matching

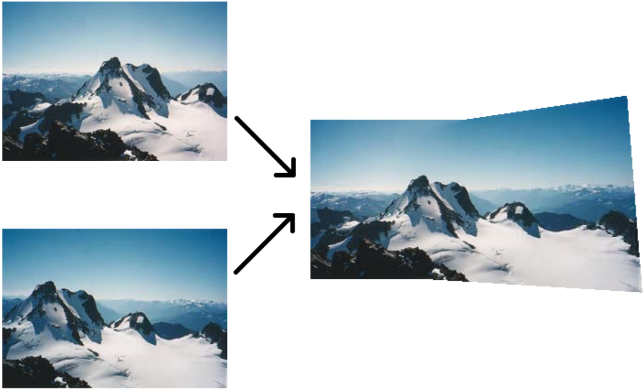
M.Sc. Defense

Gustavo T. Pfeiffer
Advisors: Ricardo G. Marroquim, Daniel R. Figueiredo

September 14th, 2015

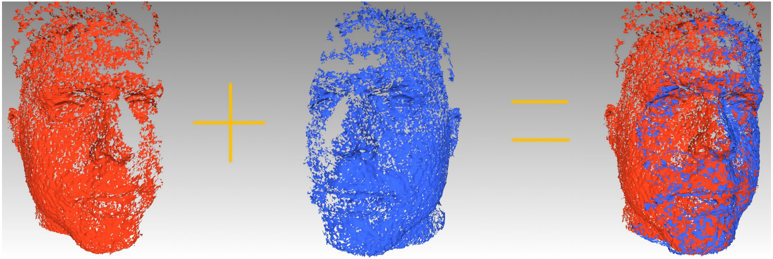
What is "*matching*"?

Image Stitching



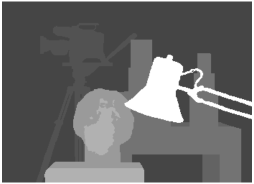
(Original images from *R. Szeliski, Computer Vision: Algorithms and Applications, 2010.*)

Point Cloud Alignment



(Image from *http://dynface4d.isr.uc.pt/images/database/MergePoints1Snap2_a.jpg*)

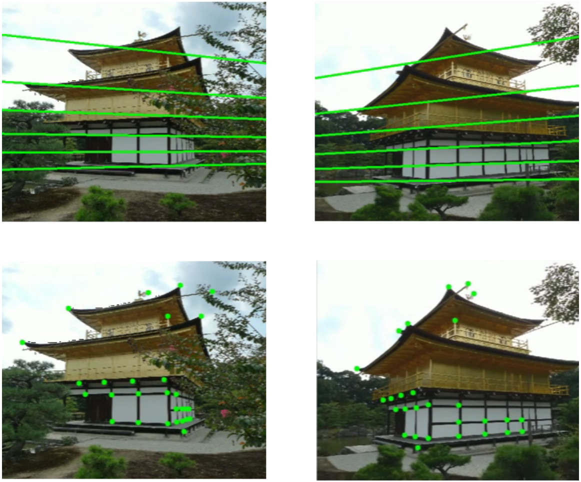
Stereo Reconstruction



(Images from *http://83.157.145.242:8080/projects/stereo/normalisation_tsu.png*

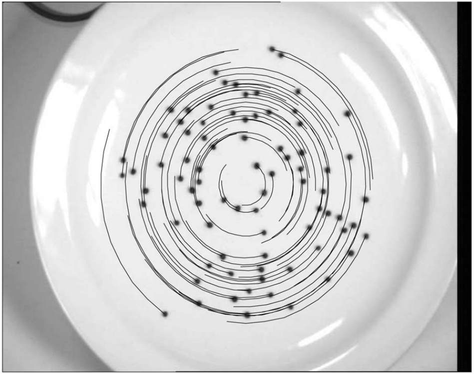
and *http://www.cs.cornell.edu/People/vnk/recon/gt.gif*)

Stereo Calibration



(Images adapted from *https://www.youtube.com/watch?v=QzYn0OPO0Yw*)

Point Tracking

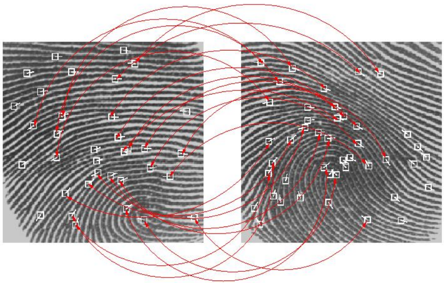


(Image from *Shafique, K., Shah, M. (2005). A Noniterative Greedy Algorithm for Multiframe Point Correspondence*)

Optical Character Recognition

(Image from *Belongie, S., Malik, J., Puzicha, J. (2002): Shape Matching and Object Recognition Using Shape Contexts*)

Fingerprint Recognition



(Image from *http://www.barcode.ro/tutorials/biometrics/img/fingermatch.jpg*)

"*Matching*" ...

- Ubiquitous in Computer Vision
- Varied problems

“*Matching*” ...

- Ubiquitous in Computer Vision
- Varied problems
  - Cannot be tackled all at once!

Our proposal

- Simple **probabilistic framework**

Our proposal

- Simple **probabilistic framework**
  - Provide optimal **methods** for matching problems

Our proposal

- Simple **probabilistic framework**
  - Provide optimal **methods** for matching problems
  - Explain **fundamental characteristics** of matching problems

Our proposal

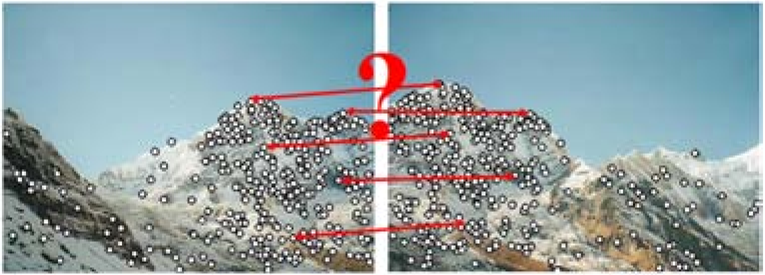
- Simple **probabilistic framework**
  - Provide optimal **methods** for matching problems
  - Explain **fundamental characteristics** of matching problems
  - **Evaluation** in computer vision applications

Our proposal

- Simple **probabilistic framework**
  - Provide optimal **methods** for matching problems
  - Explain **fundamental characteristics** of matching problems
  - **Evaluation** in computer vision applications
- Particularly well-suited to the *feature matching* problem.

The feature matching approach

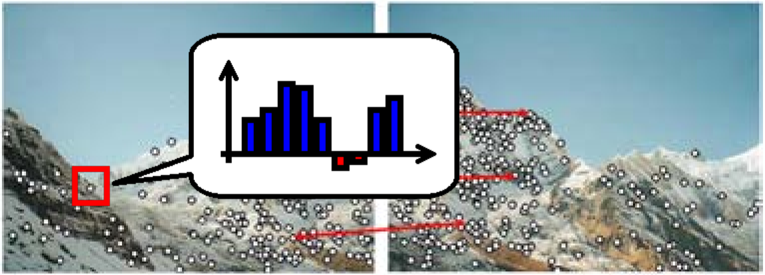
Detect, describe and match *feature* points



(Image from *R. Szeliski, Computer Vision: Algorithms and Applications, 2010.*)

The feature matching approach

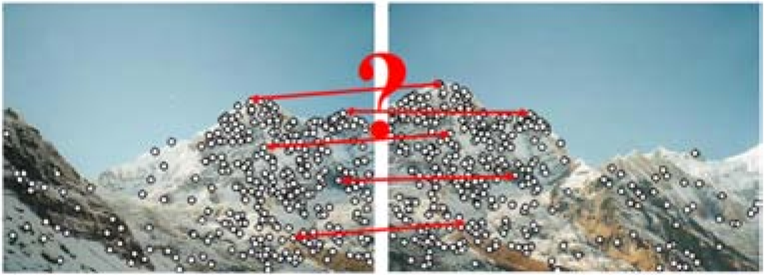
Detect, describe and match *feature* points



(Image from *R. Szeliski, Computer Vision: Algorithms and Applications, 2010.*)

The feature matching approach

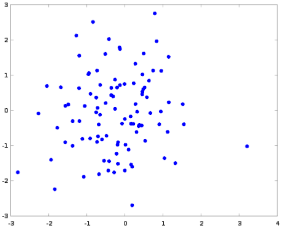
Detect, describe and match *feature* points

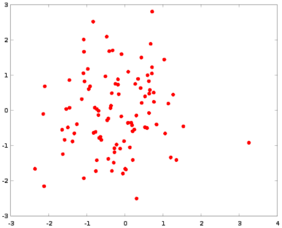


(Image from *R. Szeliski, Computer Vision: Algorithms and Applications, 2010.*)

Matching feature points: How to?

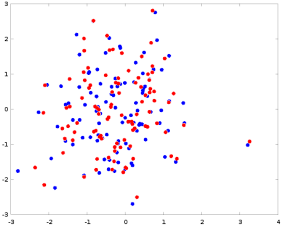
- Two sets of N points in $\mathbb{R}^n$ (very high n)
  - How to match them?





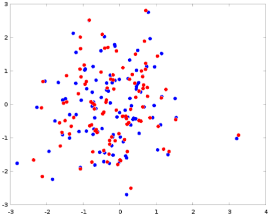
Matching feature points: How to?

- Two sets of N points in $\mathbb{R}^n$ (very high n)
  - How to match them?



Matching Strategies

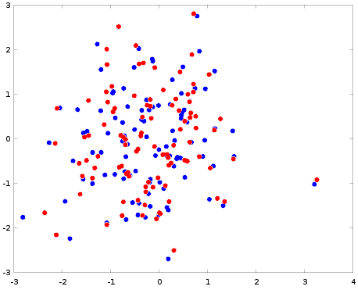
- Greedy / Heuristics (Cost: $< N^3$)
- Minimum Bipartite Matching (Cost: N^3)
- Graph-based (Cost: usually $> N^3$)



Matching Strategies

Greedy / Heuristics:

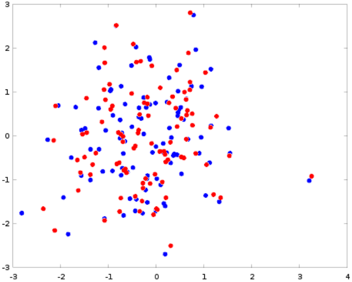
- e.g.: Select nearest point

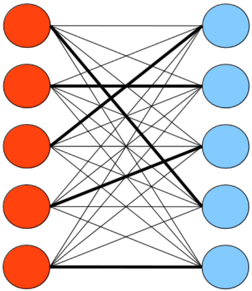


- Commonly coupled with the *two nearest neighbors* (2-NN) strategy.

Matching Strategies

Minimum bipartite matching:

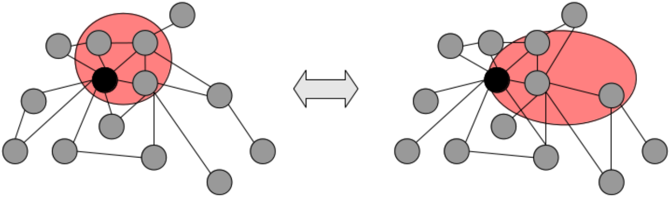




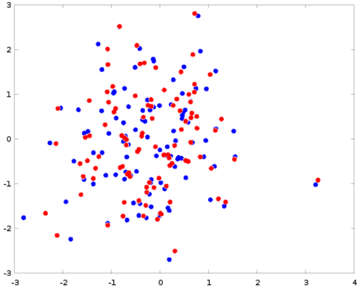
- Hungarian algorithm solves in $O(N^3)$

Matching Strategies

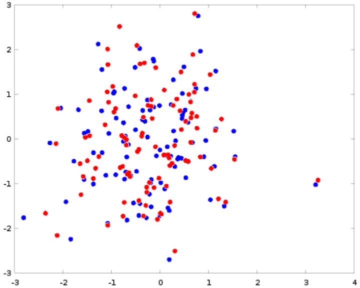
Graph-based methods:



Our Models



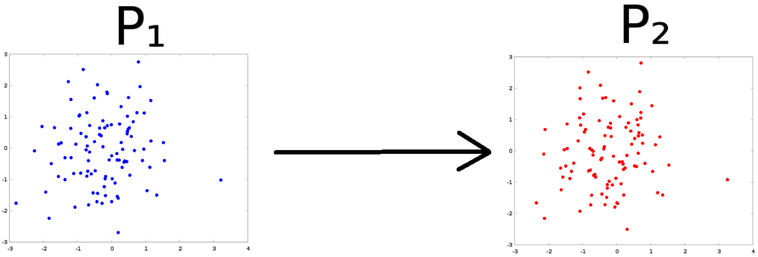
How to study this problem?



How to study this problem?

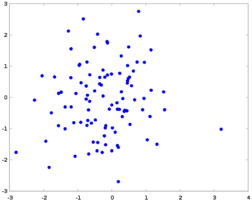
- Generative model

Direct model



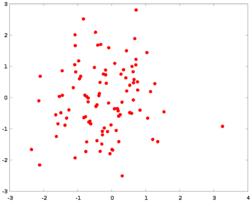
Direct model

P_1

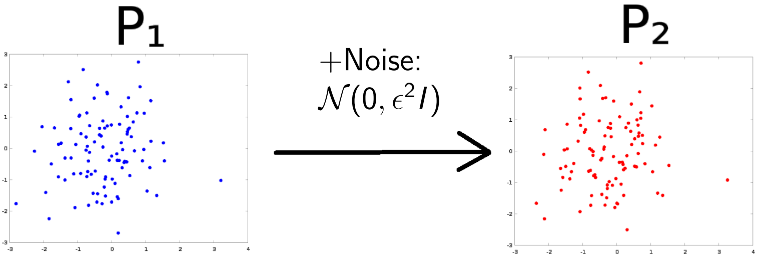


$+$Noise:
$\mathcal{N}(0, \epsilon^2 I)$

P_2

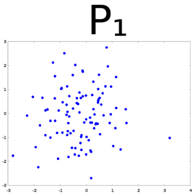


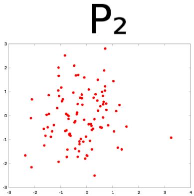
Direct model



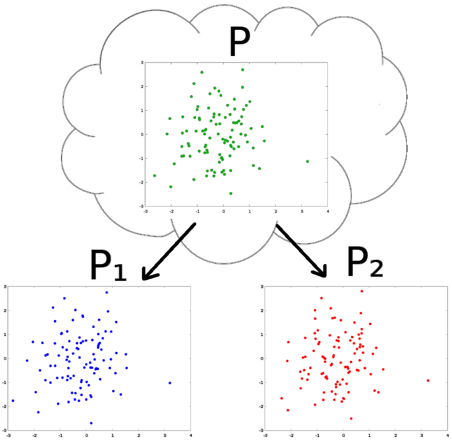
Asymmetric model: P_2 has higher variance

Generator set model

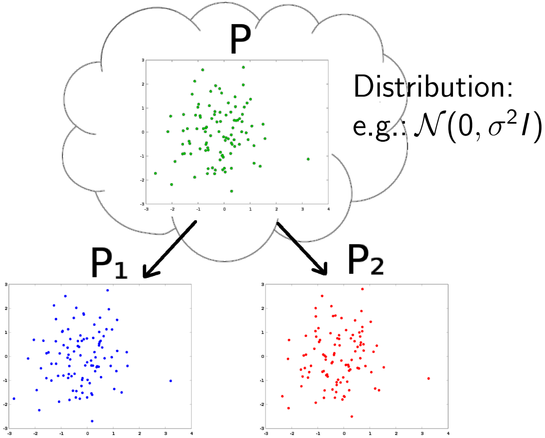




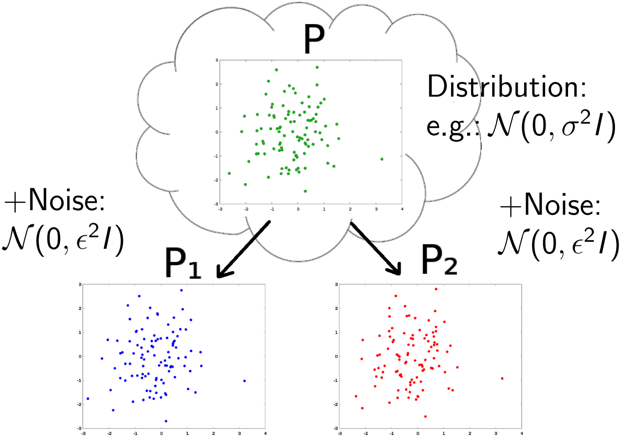
Generator set model



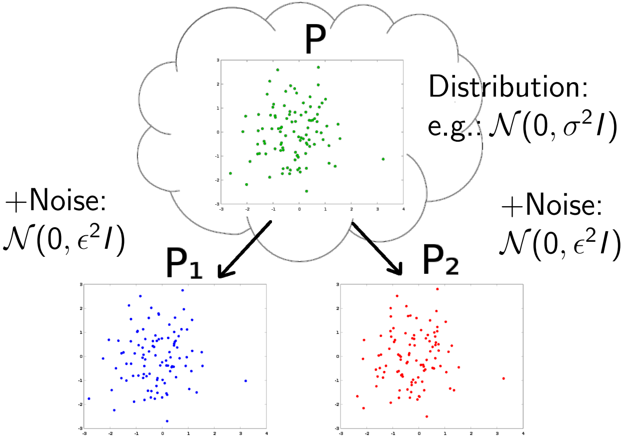
Generator set model



Generator set model

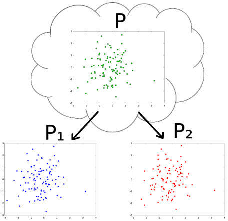


Generator set model

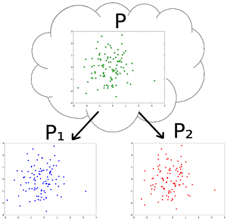


- Independence assumption

Parameters

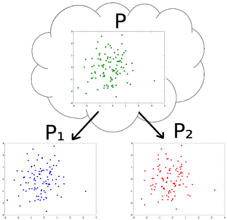


Parameters



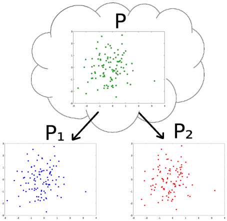
- N: number of points
- n: number of dimensions

Parameters



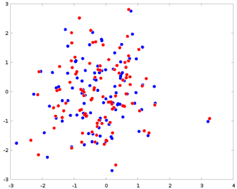
- N: number of points
- n: number of dimensions
- Generator set distribution:
  - Gaussian case: σ $\quad$ $(\text{pdf}[x] \propto e^{-\frac{1}{2}||x||^2/\sigma^2})$
  - exponential case: λ $\quad$ $(\text{pdf}[x] \propto e^{-\lambda||x||})$
  - power law case: m, α $\quad$ $(\text{pdf}[x] \propto ||x||^{-\alpha})$

Parameters



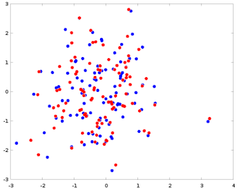
- N: number of points
- n: number of dimensions
- Generator set distribution:
  - Gaussian case: σ $\quad (\text{pdf}[x] \propto e^{-\frac{1}{2}||x||^2/\sigma^2})$
  - exponential case: λ $\quad (\text{pdf}[x] \propto e^{-\lambda||x||})$
  - power law case: m, α $\quad (\text{pdf}[x] \propto ||x||^{-\alpha})$
- Noise distribution:
  - always Gaussian: ϵ

Questions



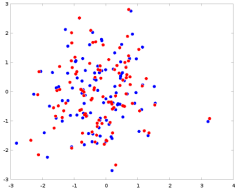
Questions

- How can we “solve” this problem?



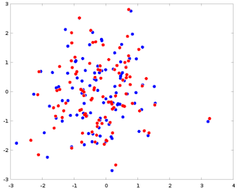
Questions

- How can we "solve" this problem?
- In which conditions can the problem be "solved"?



Questions

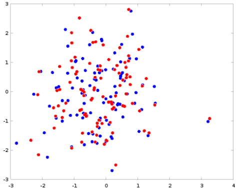
- How can we “solve” this problem?
- In which conditions can the problem be “solved”?

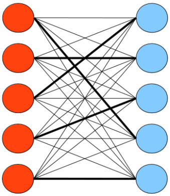


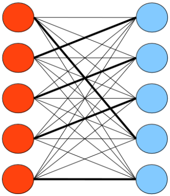
Hit count: number of correct matches

Questions

- How can we “solve” this problem?
- In which conditions can the problem be “solved”?





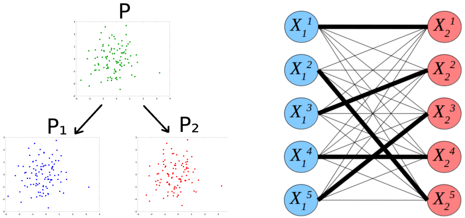


Hit count: number of correct matches

Our Methods

The "max-prob" method

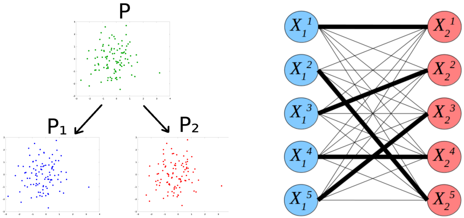
The "max-prob" method



Choose the most probable permutation:

$$\arg\max_{\Pi} P[\Pi | X_1, X_2]$$

The “max-prob” method



Choose the most probable permutation:

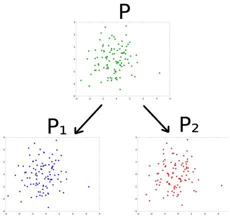
$$\arg\max_{\Pi} P[\Pi | X_1, X_2]$$

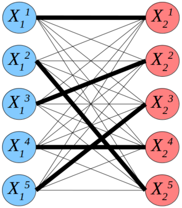
Can be solved using the Hungarian algorithm ($O(N^3)$)

$$C_{ij} = -\log \mathrm{pdf}[X_1^i, X_2^j | \Pi_{ij}]$$

The “max-expect” method

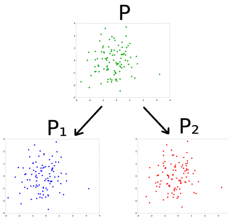
The "max-expect" method

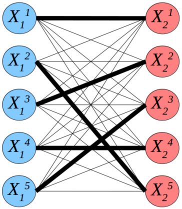




Choose the permutation with the highest expected *hit count*.

The "max-expect" method

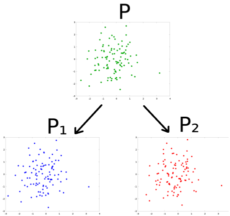


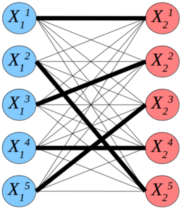


Choose the permutation with the highest expected *hit count*.

$$\arg\max_{\Pi'} E[\Pi' : \Pi | X_1, X_2]$$

The "max-expect" method





Choose the permutation with the highest expected *hit count*.

$$\arg\max_{\Pi'} E[\Pi' : \Pi | X_1, X_2]$$

- Solved using the Hungarian method ($O(N^3)$)...
  - ...but building the cost matrix is $O(2^N N^3)$

$$C_{ij} = R_{ij}\mathrm{Per}\left(R_{*ij}\right), \;\; R_{ij} = \mathrm{pdf}[X_1^i, X_2^j | \Pi_{ij}]$$

"max-prob" X "max-expect"

$\max_{\Pi} P[\Pi | X_1, X_2]$

$\max_{\Pi'} E[\Pi' : \Pi | X_1, X_2]$

$O(N^3)$

$O(N^3 2^N)$

“max-prob” X “max-expect”

$\max_{\Pi} P[\Pi | X_1, X_2]$ $\qquad$ $\max_{\Pi'} E[\Pi' : \Pi | X_1, X_2]$

$O(N^3)$ $\qquad$ $O(N^3 2^N)$

Maximize different evaluation metrics

- average hit count
- number of cases when $\#\text{hits} = N$

“max-prob” X “max-expect”

$\max_{\Pi} P[\Pi | X_1, X_2]$ $\qquad$ $\max_{\Pi'} E[\Pi' : \Pi | X_1, X_2]$

$O(N^3)$ $\qquad$ $O(N^3 2^N)$

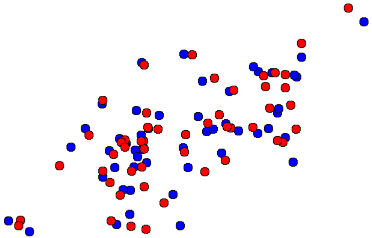
Maximize different evaluation metrics

- average hit count
- number of cases when $\#\text{hits} = N$

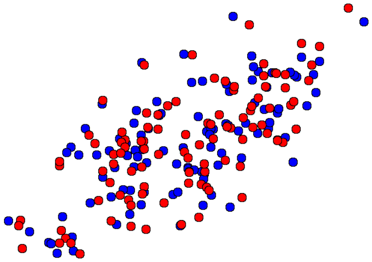
In practice, not much difference ($\sim 0.01\%$)

Theoretical Results

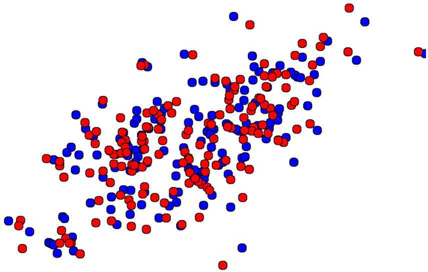
What happens when $N \to \infty$?



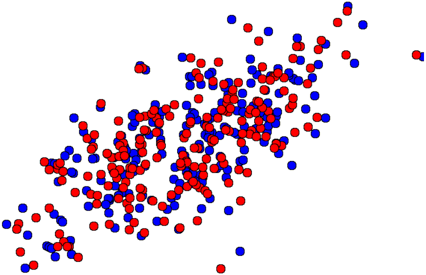
What happens when $N \to \infty$?



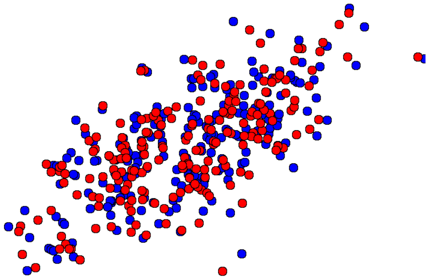
What happens when $N \to \infty$?



What happens when $N \to \infty$?

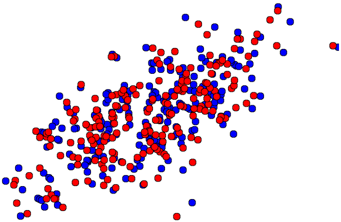


What happens when $N \to \infty$?



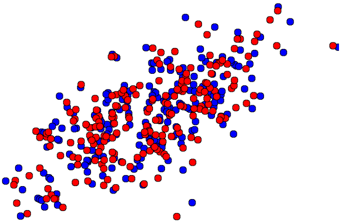
Hit rate is decreasing
What about the *hit count*?

Hit Count when $N \to \infty$



$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \Big|_{x_2 = x_2^*(x_1)}$$

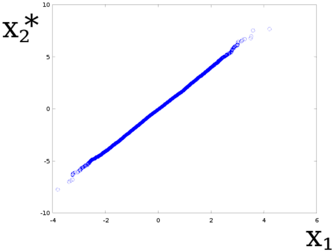
Hit Count when $N \to \infty$



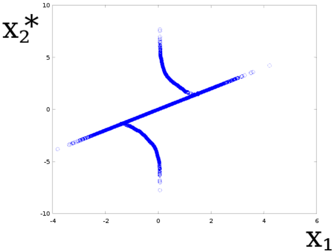
$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

$$x_2^*(x_1) = ?$$

What happens when $N \to \infty$?



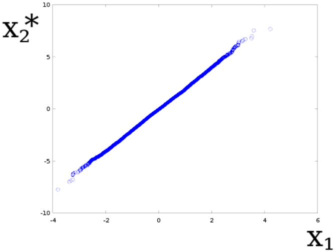
“max-prob”



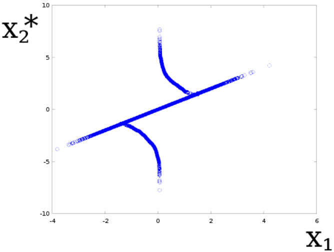
Greedy method

(Direct model)

What happens when $N \to \infty$?



"max-prob"

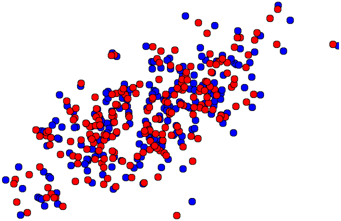


Greedy method

"Max-prob" becomes a variational calculus problem as $N \to \infty$. The solution converges to a Dirac delta:

$$\text{pdf}[x_2^*|x_1] = \delta(x_2^* - x_2^*(x_1))$$

What happens when $N \to \infty$?



Generator set model with "max-prob" cost function implies identity transformation

$$x_2^* = x_2^*(x_1) = x_1$$

Strong result: applies to any distribution and noise model

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

$$\lim_{N \to \infty} E[\#\text{hits}] = (1 + \sigma^2/\epsilon^2)^n$$

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

$$\lim_{N \to \infty} E[\#\text{hits}] = (1 + \sigma^2/\epsilon^2)^n$$

- Exponential distribution

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \Big|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

$$\lim_{N \to \infty} E[\#\text{hits}] = (1 + \sigma^2/\epsilon^2)^n$$

- Exponential distribution

$$\lim_{N \to \infty} E[\#\text{hits}] = +\infty$$

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

$$\lim_{N \to \infty} E[\#\text{hits}] = (1 + \sigma^2/\epsilon^2)^n$$

- Exponential distribution

$$\lim_{N \to \infty} E[\#\text{hits}] = +\infty$$

- Power-law distribution

Hit Count when $N \to \infty$

$$E[\#\text{hits}] = \int_{\mathbb{R}^n} \frac{\text{pdf}[x_1, x_2]}{\text{pdf}[x_2]} dx_1 \bigg|_{x_2 = x_2^*(x_1)}$$

- Gaussian distribution?

$$\lim_{N \to \infty} E[\#\text{hits}] = (1 + \sigma^2/\epsilon^2)^n$$

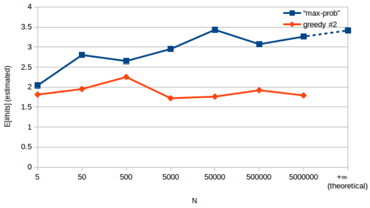
- Exponential distribution

$$\lim_{N \to \infty} E[\#\text{hits}] = +\infty$$

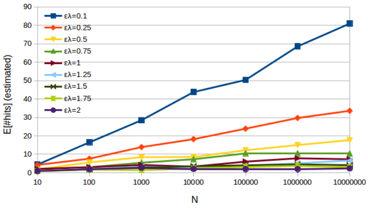
- Power-law distribution

$$\lim_{N \to \infty} E[\#\text{hits}] = +\infty$$

Hit Count when $N \to \infty$



Gaussian

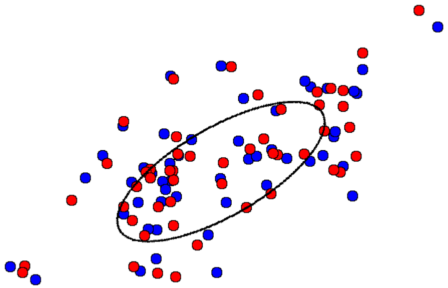


Exponential

Asymptotic Hit Count: Lower Bound

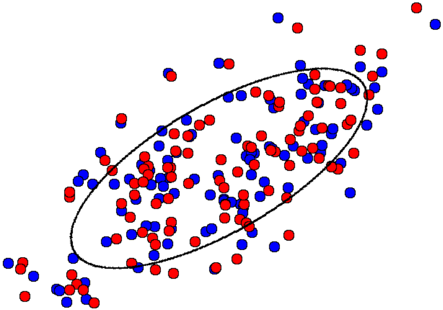
Asymptotic Hit Count: Lower Bound

- Method: Restrict to points with high hit probability
  - Region with low point density



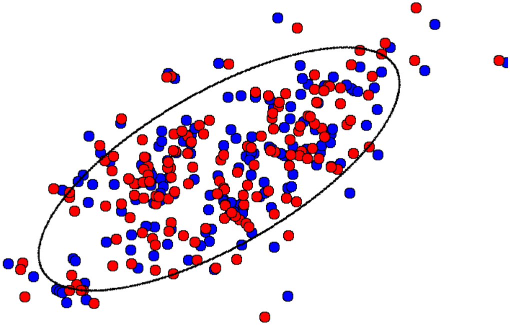
Asymptotic Hit Count: Lower Bound

- Method: Restrict to points with high hit probability
  - Region with low point density



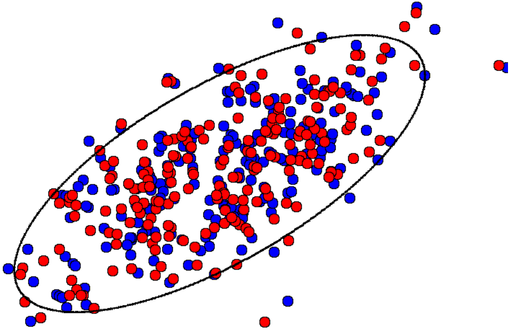
Asymptotic Hit Count: Lower Bound

- Method: Restrict to points with high hit probability
    - Region with low point density



Asymptotic Hit Count: Lower Bound

- Method: Restrict to points with high hit probability
  - Region with low point density



Asymptotic Hit Count: Lower Bound

- Power law:

$$E[\#\text{hits}] = \Omega(N^{n/\alpha})$$

Asymptotic Hit Count: Lower Bound

- Power law:
$$E[\#\text{hits}] = \Omega(N^{n/\alpha})$$

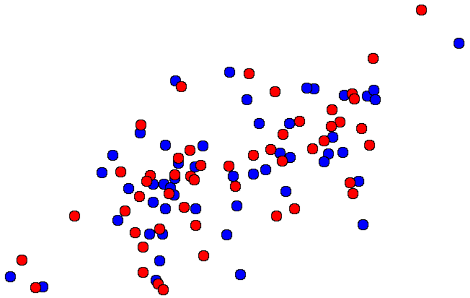
- Exponential:
$$E[\#\text{hits}] = \Omega((\log N)^{n-1}) \text{ (loose bound)}$$

Noise vs. Number of Points

- Increasing N reduces hit rate
- Reducing noise (ϵ) increases hit rate
- What is the tradeoff?

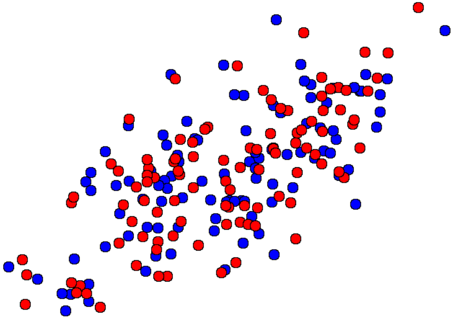
Noise vs. Number of Points

- Increasing N reduces hit rate
- Reducing noise (ϵ) increases hit rate
- What is the tradeoff?



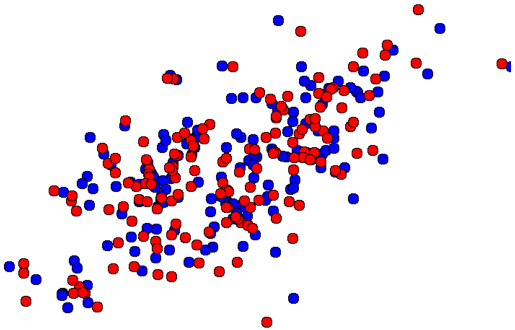
Noise vs. Number of Points

- Increasing N reduces hit rate
- Reducing noise (ϵ) increases hit rate
- What is the tradeoff?



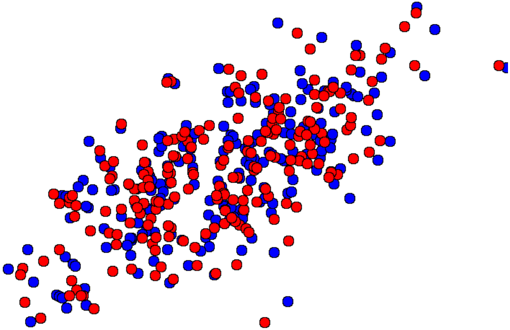
Noise vs. Number of Points

- Increasing N reduces hit rate
- Reducing noise (ϵ) increases hit rate
- What is the tradeoff?



Noise vs. Number of Points

- Increasing N reduces hit rate
- Reducing noise (ϵ) increases hit rate
- What is the tradeoff?



Condition for Minimum Hit Rate

- $\epsilon^n < C/N \Rightarrow E[\#\text{hits}] \gtrsim \bar{Q}N$

Condition for Minimum Hit Rate

- $\epsilon^n < C/N \Rightarrow E[\#\text{hits}] \gtrsim \bar{Q}N$
- $\epsilon^n = o(1/N) \Rightarrow E[\#\text{hits}] \sim N$

Condition for Minimum Hit Rate

- $\epsilon^n < C/N \Rightarrow E[\#\text{hits}] \gtrsim \bar{Q}N$
- $\epsilon^n = o(1/N) \Rightarrow E[\#\text{hits}] \sim N$
- Applies to distributions satisfying $\max_x \text{pdf}[x] < +\infty$.

Hitting all Pairs

Hitting all Pairs

Similar result:

$$\epsilon^n < C/N^2 \Rightarrow P[\#\text{hits} = N] \gtrsim \bar{Q}$$

$$\epsilon^n = o(1/N^2) \Rightarrow P[\#\text{hits} = N] \sim 1$$

Hitting all Pairs

Similar result:

$$\epsilon^n < C/N^2 \Rightarrow P[\#\text{hits} = N] \gtrsim \bar{Q}$$

$$\epsilon^n = o(1/N^2) \Rightarrow P[\#\text{hits} = N] \sim 1$$

What is the difference?

- $\epsilon^n = o(1/N) \Rightarrow E[\#hits] \sim N$
- $\epsilon^n = o(1/N^2) \Rightarrow P[\#hits = N] \sim 1$

What happens between $\epsilon^n = o(1/N)$ and $\epsilon^n = \omega(1/N^2)$?

Hitting all Pairs

Similar result:

$$\epsilon^n < C/N^2 \Rightarrow P[\#\text{hits} = N] \gtrsim \bar{Q}$$

$$\epsilon^n = o(1/N^2) \Rightarrow P[\#\text{hits} = N] \sim 1$$

What is the difference?

- $\epsilon^n = o(1/N) \Rightarrow E[\#hits] \sim N$
- $\epsilon^n = o(1/N^2) \Rightarrow P[\#hits = N] \sim 1$

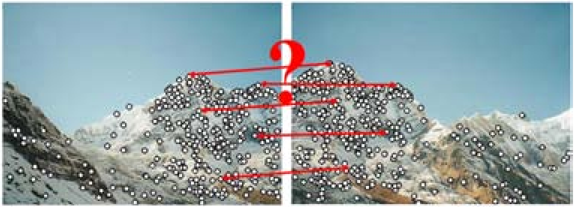
What happens between $\epsilon^n = o(1/N)$ and $\epsilon^n = \omega(1/N^2)$?

- The *miss count* is $o(N)$ but $\omega(1)$
- So $E[\#hits] \sim N$ but $P[\#hits = N] \to 0$.

Application

Overview

Instantiation in feature matching

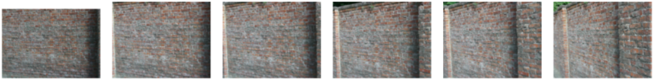


- Models for Harris/NCC and RootSIFT features
- Evaluation using Mikolajczyk's dataset

Mikolajczyk's Dataset

(a) graf-1 (b) graf-2 (c) graf-3 (d) graf-4 (e) graf-5 (f) graf-6

(g) bikes-1 (h) bikes-2 (i) bikes-3 (j) bikes-4 (k) bikes-5 (l) bikes-6



(m) wall-1 (n) wall-2 (o) wall-3 (p) wall-4 (q) wall-5 (r) wall-6

(s) trees-1 (t) trees-2 (u) trees-3 (v) trees-4 (w) trees-5 (x) trees-6

Instantiating “max-prob”...

Issues with real world data:

Instantiating "max-prob"...

Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Instantiating "max-prob"...

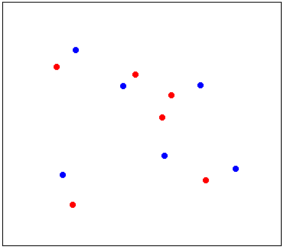
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;

Instantiating "max-prob"...



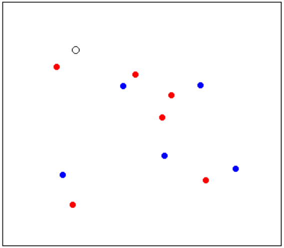
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;

Instantiating "max-prob"...



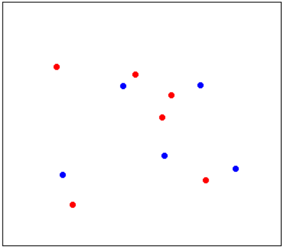
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;

Instantiating "max-prob"...



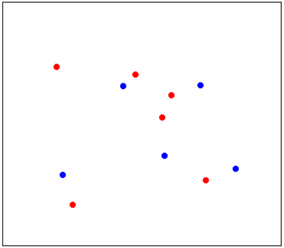
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;

Instantiating “max-prob”...



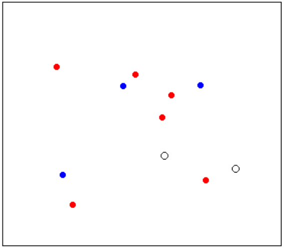
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;
- In addition: Probability q of point becoming outlier (point is generated again)

Instantiating "max-prob"...



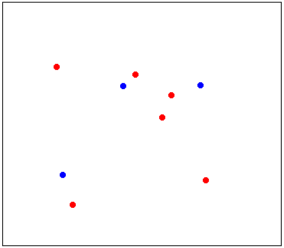
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;
- In addition: Probability q of point becoming outlier (point is generated again)

Instantiating "max-prob"...



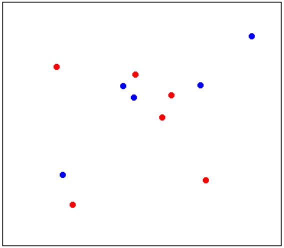
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;
- In addition: Probability q of point becoming outlier (point is generated again)

Instantiating "max-prob"...



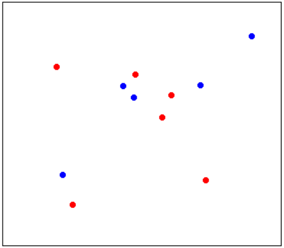
Issues with real world data:

- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;
- In addition: Probability q of point becoming outlier (point is generated again)

Instantiating "max-prob"...



Issues with real world data:

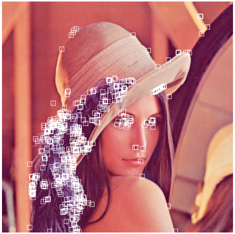
- Sets of different sizes (N_1, N_2)
- Outliers (points without matches)

Adaptation:

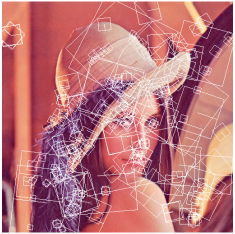
- Generator set has $\max\{N_1, N_2\}$ points, one of the sets has $|N_2 - N_1|$ points occluded;
- In addition: Probability q of point becoming outlier (point is generated again)

Analogous algorithms

Harris/NCC and RootSIFT features



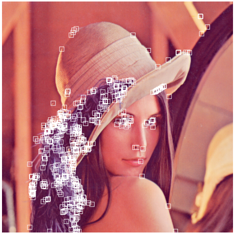
Harris/NCC



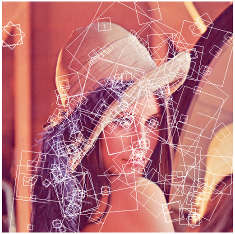
SIFT

- NCC descriptors have zero mean and unitary L2 norm
- SIFT descriptor has positive entries (histogram-like)
    - After RootSIFT normalization: unitary L2 norm

Harris/NCC and RootSIFT features



Harris/NCC



SIFT

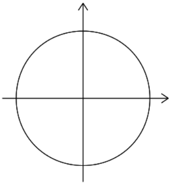
- NCC descriptors have zero mean and unitary L2 norm
- SIFT descriptor has positive entries (histogram-like)
    - After RootSIFT normalization: unitary L2 norm

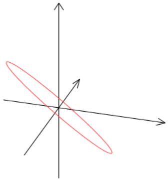
Gaussian model is not appropriate!

Harris/NCC and RootSIFT models

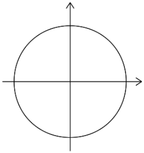
Our model:

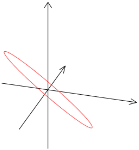
- Gaussian variables
- But normalized to satisfy zero mean (Harris/NCC case only) and unitary norm





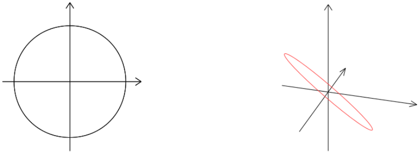
Harris/NCC and RootSIFT models





Allows anisotropic distributions

Harris/NCC and RootSIFT models



Allows anisotropic distributions

Efficient MLE method is provided

- Estimates a covariance matrix C

$$\mathrm{pdf}[x] \propto \frac{1}{(x^T C^{-1} x)^{n/2}}$$

- Feeds MLE with the input sets of the matching problem

Evaluation

Evaluation

Harris/NCC

- Greedy method
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- Harris/NCC model (isotropic)

Evaluation

Harris/NCC

- Greedy method
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- Harris/NCC model (isotropic)

RootSIFT

- Greedy method
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- isotropic RootSIFT model
- anisotropic RootSIFT model
- hybrid RootSIFT model (anisotropic G.S. and isotropic noise)

Methodology

- Parameters of our methods:
  - Outlier rate $q \in]0,1[$
  - Noise/signal rate $\chi \in]0,1[$

Methodology

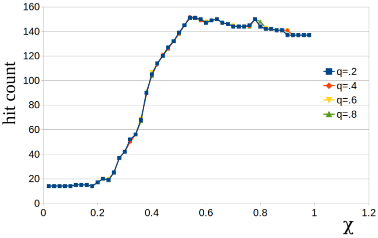
- Parameters of our methods:
  - Outlier rate $q \in]0,1[\quad \rightarrow$ (not required in practice)
  - Noise/signal rate $\chi \in]0,1[$

Methodology

- Parameters of our methods:
  - Outlier rate $q \in]0,1[\quad \rightarrow$ (not required in practice)
  - Noise/signal rate $\chi \in]0,1[$
- The other methods require no parameters

Methodology

- Parameters of our methods:
  - Outlier rate $q \in]0,1[\quad \rightarrow \quad$ (not required in practice)
  - Noise/signal rate $\chi \in]0,1[$
- The other methods require no parameters
- Methodology: Vary $\chi \in \{.02, .04, \ldots, .98\}$ and analyze the **median** and **maximum** hit counts



Harris/NCC

- Greedy method
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- Harris/NCC model (isotropic)

RootSIFT

- Greedy method
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- isotropic RootSIFT model
- anisotropic RootSIFT model
- hybrid RootSIFT model (anisotropic G.S. and isotropic noise)

Selected findings

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;

Selected findings

Harris/NCC

- Greedy method
- **MBM with L2 distance**
- MBM with squared L2 distance
- Gaussian model
- **Harris/NCC model (isotropic)**

RootSIFT

- **Greedy method**
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- isotropic RootSIFT model
- anisotropic RootSIFT model
- **hybrid RootSIFT model (anisotropic G.S. and isotropic noise)**

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;

Selected findings

Harris/NCC

- Greedy method
- **MBM with L2 distance**
- MBM with squared L2 distance
- Gaussian model
- **Harris/NCC model (isotropic)**

RootSIFT

- **Greedy method**
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- isotropic RootSIFT model
- anisotropic RootSIFT model
- **hybrid RootSIFT model (anisotropic G.S. and isotropic noise)**

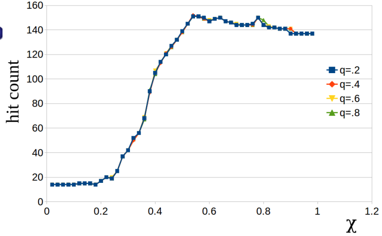
Selected findings

Harris/NCC

- Greedy method
- **MBM with L2 distance**
- MBM with squared L2 distance
- Gaussian model
-

RootSIFT

- **Greedy method**
- MBM with L2 distance
- MBM with squared L2 distance
- Gaussian model
- isotropic RootSIFT model
- nisotropic RootSIFT model
- **ybrid RootSIFT model (anisotropic G.S. and sotropic noise)**



Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;
- The difference in hit count between the methods is minute ($\sim 1\%$)

Table : Hit count comparison for Harris/NCC features

case	#features	G2	E2	E1	GA	HN
graf1-2	478 $\times$ 488	128	137	**148**	**152**|**137**	**152**|**137**
graf1-3	478 $\times$ 483	**71**	69	70	**71**|63	**71**|**66**
graf1-4	478 $\times$ 482	**10**	**10**	**10**	**11**|**8**	**11**|**8**
graf1-5	478 $\times$ 484	22	**31**	30	**33**|27	31|**28**
graf1-6	478 $\times$ 468	7	**8**	6	**10**|4	9|**7**
bikes1-2	483 $\times$ 495	329	338	**344**	**343**|338	**343**|**339**
bikes1-3	483 $\times$ 489	301	308	**311**	310|308	**313**|**311**
bikes1-4	483 $\times$ 489	221	230	**236**	235|229	**236**|**232**
bikes1-5	483 $\times$ 485	143	149	**155**	157|144	**163**|**149**
bikes1-6	483 $\times$ 482	67	**80**	76	83|76	**86**|**77**
wall1-2	480 $\times$ 490	**337**	334	336	**337**|331	**337**|**334**
wall1-3	480 $\times$ 483	**298**	297	297	**301**|281	300|**292**
wall1-4	480 $\times$ 478	**194**	192	**194**	**195**|170	**195**|**185**
wall1-5	480 $\times$ 487	113	121	**128**	**127**|83	125|**106**
wall1-6	480 $\times$ 492	34	41	**42**	**42**|17	**42**|**23**
trees1-2	487 $\times$ 482	**210**	206	207	208|199	**209**|**206**
trees1-3	487 $\times$ 488	**168**	167	**168**	167|164	**169**|**166**
trees1-4	487 $\times$ 489	74	76	**77**	**77**|73	**77**|**74**
trees1-5	487 $\times$ 475	44	**47**	45	**49**|**45**	48|**45**
trees1-6	487 $\times$ 486	15	**18**	17	19|16	**21**|**17**
	bold count	7	6	11	13|3	15|20

Table : Hit count comparison for Harris/NCC features

case	#features	G2	E2	E1	GA	HN
graf1-2	478 × 488	128	137	**148**	**152**|**137**	**152**|**137**
graf1-3	478 × 483	**71**	69	70	**71**|63	**71**|**66**
graf1-4	478 × 482	**10**	**10**	**10**	**11**|**8**	**11**|**8**
graf1-5	478 × 484	22	**31**	30	**33**|27	31|**28**
graf1-6	478 × 468	7	**8**	6	**10**|4	9|**7**
bikes1-2	483 × 495	329	338	**344**	**343**|338	**343**|**339**
bikes1-3	483 × 489	301	308	**311**	310|308	**313**|**311**
bikes1-4	483 × 489	221	230	**236**	235|229	**236**|**232**
bikes1-5	483 × 485	143	149	**155**	157|144	**163**|**149**
bikes1-6	483 × 482	67	**80**	76	83|76	**86**|**77**
wall1-2	480 × 490	**337**	334	336	**337**|331	**337**|**334**
wall1-3	480 × 483	**298**	297	297	**301**|281	300|**292**
wall1-4	480 × 478	**194**	192	**194**	**195**|170	**195**|**185**
wall1-5	480 × 487	113	121	**128**	**127**|83	125|**106**
wall1-6	480 × 492	34	41	**42**	**42**|17	**42**|**23**
trees1-2	487 × 482	**210**	206	207	208|199	**209**|**206**
trees1-3	487 × 488	**168**	167	**168**	167|164	**169**|**166**
trees1-4	487 × 489	74	76	**77**	**77**|73	**77**|**74**
trees1-5	487 × 475	44	**47**	45	**49**|**45**	48|**45**
trees1-6	487 × 486	15	**18**	17	19|16	**21**|**17**
	bold count	7	6	11	13|3	15|20

Table : Hit count comparison for RootSIFT features.

case	#features	G2	E2	E1	GA		II		AI		AA	
G1-2	636×742	**341**	338	338	338	**338**	**339**	**338**	**339**	**338**	338	337
G1-3	636×885	**211**	207	206	212	207	212	207	**213**	**209**	210	203
G1-4	636×909	**76**	74	**76**	79	74	**80**	**75**	79	74	75	73
G1-5	636×1009	**19**	**19**	**19**	19	**19**	19	**19**	**21**	18	15	13
G1-6	636×1120	7	7	**8**	7	**7**	8	**7**	**9**	**7**	6	6
B1-2	653×428	310	**313**	**313**	313	313	**314**	313	313	312	**314**	**314**
B1-3	653×268	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**
B1-4	653×143	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**
B1-5	653×102	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**
B1-6	653×68	**50**	49	**50**	**50**	49	**50**	49	**50**	**50**	**50**	**50**
W-2	514×650	**288**	286	286	286	**286**	287	**286**	**288**	**286**	286	285
W1-3	514×635	**215**	214	**215**	**215**	214	**215**	**215**	**215**	**215**	214	214
W1-4	514×612	136	135	**137**	**137**	135	**137**	**136**	**137**	**136**	**137**	**136**
W1-5	514×657	**90**	83	83	**90**	83	**90**	**84**	**90**	**84**	89	83
W1-6	514×629	**19**	**19**	**19**	20	**19**	20	**19**	**22**	**19**	17	16
T1-2	797×742	**289**	287	287	289	**287**	289	**287**	289	**287**	**290**	**287**
T1-3	797×934	297	297	**300**	298	297	**300**	297	**300**	**299**	297	295
T1-4	797×700	**192**	188	188	**195**	**188**	194	**188**	**195**	**188**	**195**	**188**
T1-5	797×361	**103**	101	100	102	101	**103**	101	**103**	**102**	102	101
T1-6	797×227	60	**61**	**61**	61	61	61	61	61	61	**62**	**62**
	bold count	15	7	13	8	10	12	14	16	16	9	9

Table : Hit count comparison for RootSIFT features.

case	#features	G2	E2	E1	GA		II		AI		AA	
G1-2	636 × 742	**341**	338	338	338	**338**	**339**	**338**	**339**	**338**	338	337
G1-3	636 × 885	**211**	207	206	212	207	212	207	**213**	**209**	210	203
G1-4	636 × 909	**76**	74	**76**	79	74	**80**	**75**	79	74	75	73
G1-5	636 × 1009	**19**	**19**	**19**	19	**19**	19	**19**	**21**	18	15	13
G1-6	636 × 1120	7	7	**8**	7	**7**	8	**7**	**9**	**7**	6	6
B1-2	653 × 428	310	**313**	**313**	313	313	**314**	313	313	312	**314**	**314**
B1-3	653 × 268	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**	**206**
B1-4	653 × 143	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**	**105**
B1-5	653 × 102	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**	**68**
B1-6	653 × 68	**50**	49	**50**	**50**	49	**50**	49	**50**	**50**	**50**	**50**
W-2	514 × 650	**288**	286	286	286	**286**	287	**286**	**288**	**286**	286	285
W1-3	514 × 635	**215**	214	**215**	**215**	214	**215**	**215**	**215**	**215**	214	214
W1-4	514 × 612	136	135	**137**	**137**	135	**137**	**136**	**137**	**136**	**137**	**136**
W1-5	514 × 657	**90**	83	83	**90**	83	**90**	**84**	**90**	**84**	89	83
W1-6	514 × 629	**19**	**19**	**19**	20	**19**	20	**19**	**22**	**19**	17	16
T1-2	797 × 742	**289**	287	287	289	**287**	289	**287**	289	**287**	**290**	**287**
T1-3	797 × 934	297	297	**300**	298	297	**300**	297	**300**	**299**	297	295
T1-4	797 × 700	**192**	188	188	**195**	**188**	194	**188**	**195**	**188**	**195**	**188**
T1-5	797 × 361	**103**	101	100	102	101	**103**	101	**103**	**102**	102	101
T1-6	797 × 227	60	**61**	**61**	61	61	61	61	61	61	**62**	**62**
	bold count	15	7	13	8	10	12	14	16	16	9	9

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;
- The difference in hit count between the methods is minute ($\sim 1\%$)

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;
- The difference in hit count between the methods is minute ($\sim 1\%$)
  - No significant improvement for the final application

Selected findings

- Among our methods, the Harris/NCC model (isotropic) and the hybrid RootSIFT model had the best results;
- The *maximum* hit count of our methods was better than the other (non-parametric) methods, but the *median* was worse;
- The difference in hit count between the methods is minute ($\sim 1\%$)
    - No significant improvement for the final application
    - Greedy algorithm is already very good for feature matching

Conclusions

Conclusions

- A probabilistic framework for matching problems

Conclusions

- A probabilistic framework for matching problems
- Optimal methods

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties
  - Tradeoff between noise and number of points

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties
    - Tradeoff between noise and number of points
- Instantiation in feature matching

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties
  - Tradeoff between noise and number of points
- Instantiation in feature matching
  - Not better than the state-of-art

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties
  - Tradeoff between noise and number of points
- Instantiation in feature matching
  - Not better than the state-of-art
    - But not worse either!

Conclusions

- A probabilistic framework for matching problems
- Optimal methods
- Asymptotic properties
  - Tradeoff between noise and number of points
- Instantiation in feature matching
  - Not better than the state-of-art
    - But not worse either!
  - Possibly due to high dimensionality

Future Work

Future Work

- Other applications, possibly out of Computer Vision

Future Work

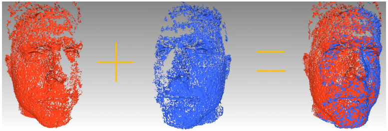
- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)

Future Work

- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)
  - May better explain other matching problems such as *point set registration*

Future Work

- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)
  - May better explain other matching problems such as *point set registration*



(Image from *http://dynface4d.isr.uc.pt/images/database/MergePoints1Snap2_a.jpg*)

Future Work

- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)
  - May better explain other matching problems such as *point set registration*

Future Work

- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)
  - May better explain other matching problems such as *point set registration*
- Analysis of methods that detect outliers (e.g.: 2-NN)

Future Work

- Other applications, possibly out of Computer Vision
- Models for correlated noise (e.g.: bias)
  - May better explain other matching problems such as *point set registration*
- Analysis of methods that detect outliers (e.g.: 2-NN)
- Upper bounds

Thank you for your attention!

Questions?