Robot Vision

Robust Estimation/RANSAC

Dr. Chen Feng

cfeng@nyu.edu

ROB-UY 3203, Spring 2024

Overview

- Feature Detection and Matching
- 2D line/3D plane fitting
  - Regular linear regression
  - Total least squares
- Robust fitting - RANSAC
  - Intuitions behind RANSAC
  - How RANSAC works
  - Why minimal solution is important
  - More example problems

References

- Hartley & Zisserman 2003:
  - Section 4.7
- Corke 2011:
  - Section 14.1, 14.2.3
- Forsyth & Ponce 2011:
  - Chapter 5, Section 10.4
- Szeliski 2022:
  - Section 7.1, 7.2, 8.1.4

Feature-based SfM Pipeline – Overview

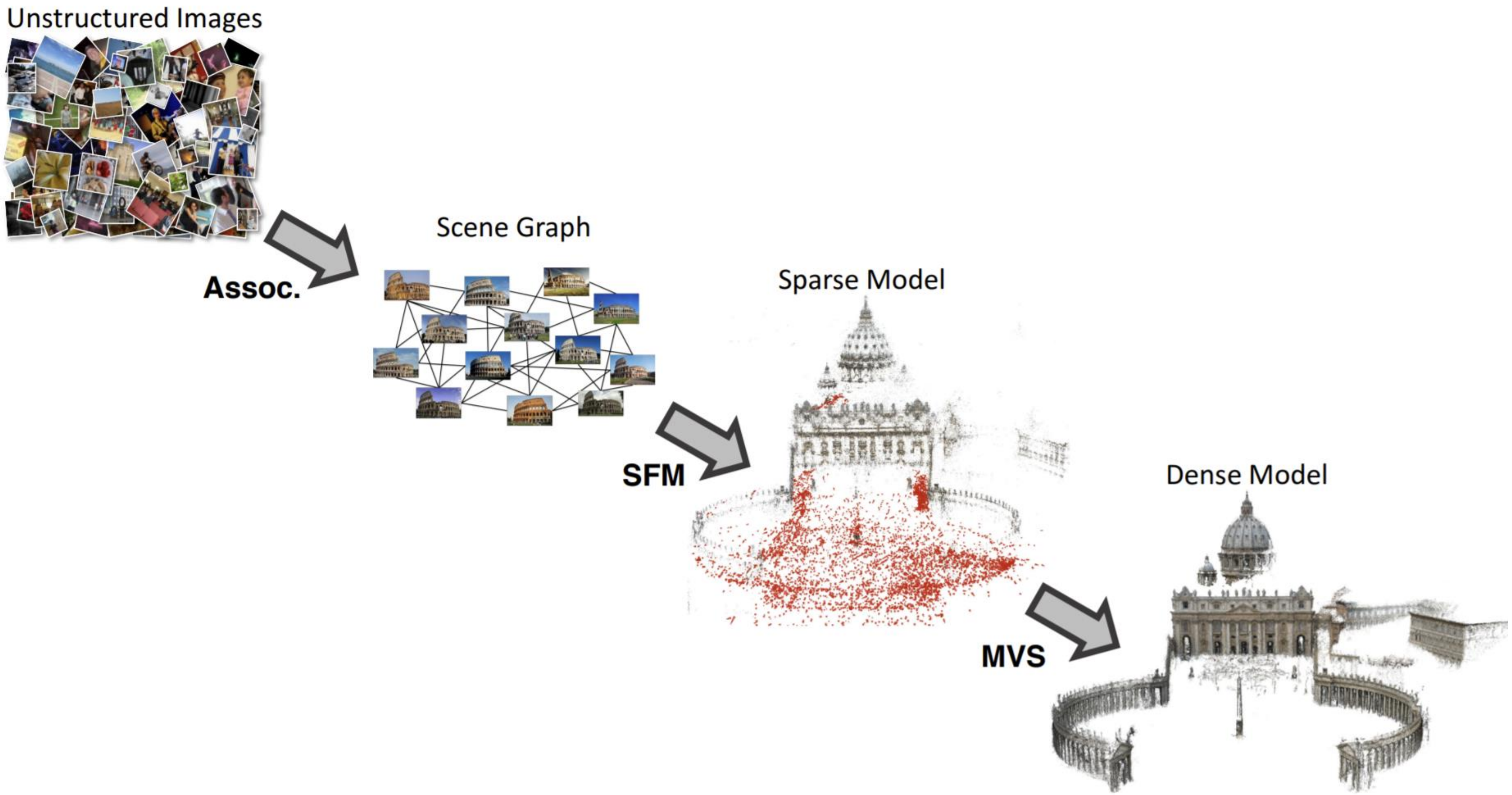
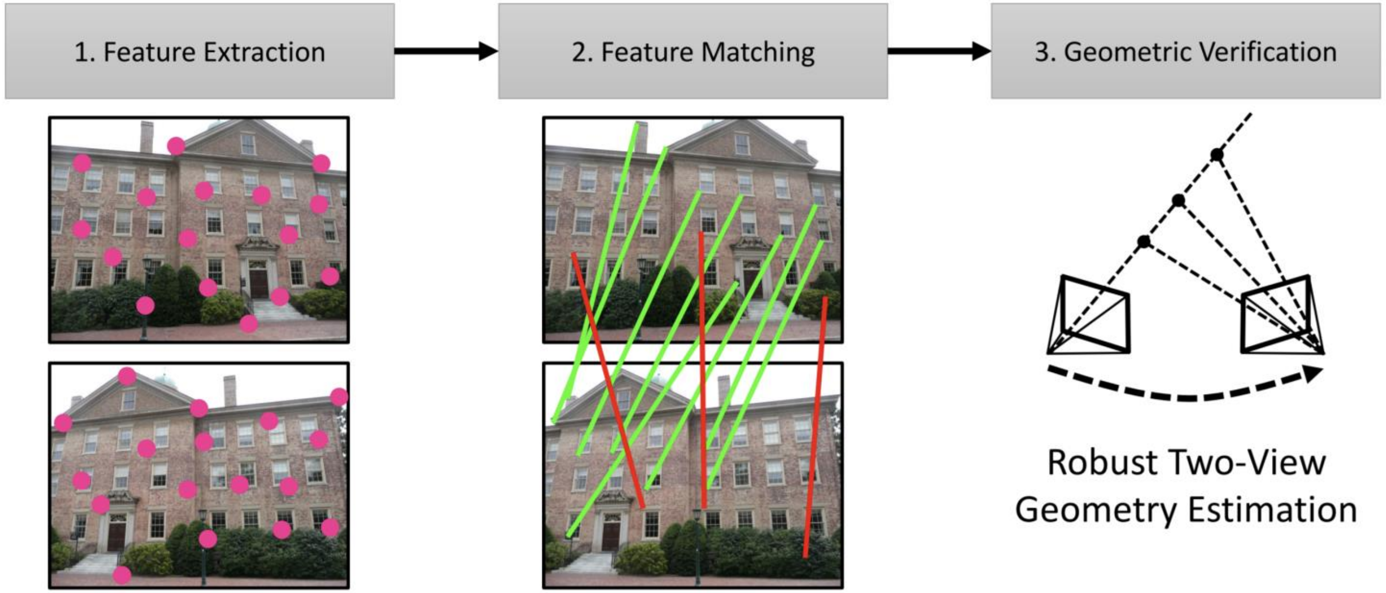


Image from: https://demuc.de/tutorials/cvpr2017/sparse-modeling.pdf

Data Association

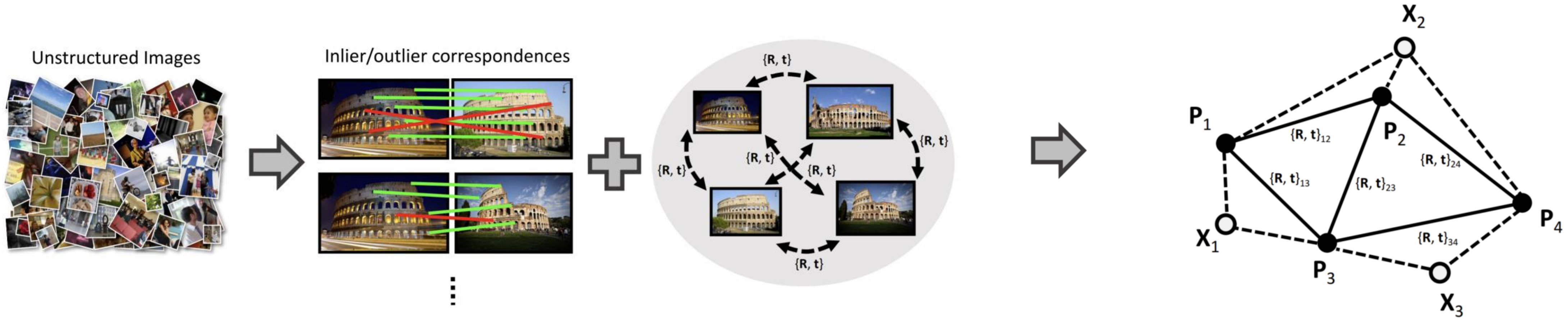


General	Planar	Panoramic
• Fundamental matrix F (*uncalibrated*) • Essential matrix E (*calibrated*)	• Homography H	• Homography H
• 7 correspondences • 5 correspondences	• 4 correspondences	• 4 correspondences

Images from: https://demuc.de/tutorials/cvpr2017/sparse-modeling.pdf

Data Association

- Data association creates a graph of cameras/views and landmark points

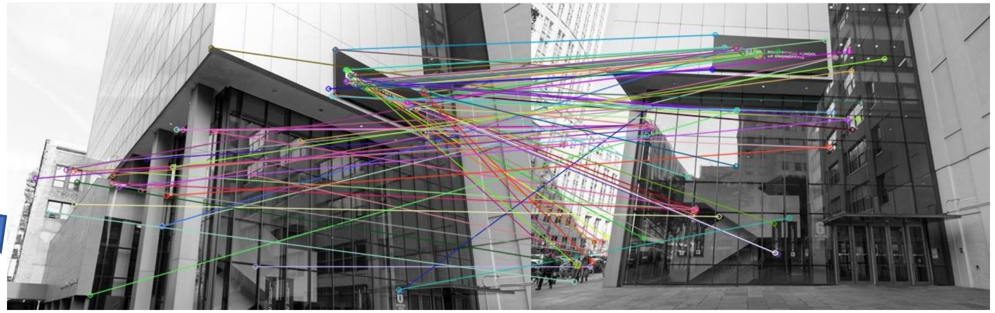


Images from: https://demuc.de/tutorials/cvpr2017/sparse-modeling.pdf

A Classic Vision Pipeline

Feature Detection
(Harris/FAST/SIFT/SURF/ORB/LSD)

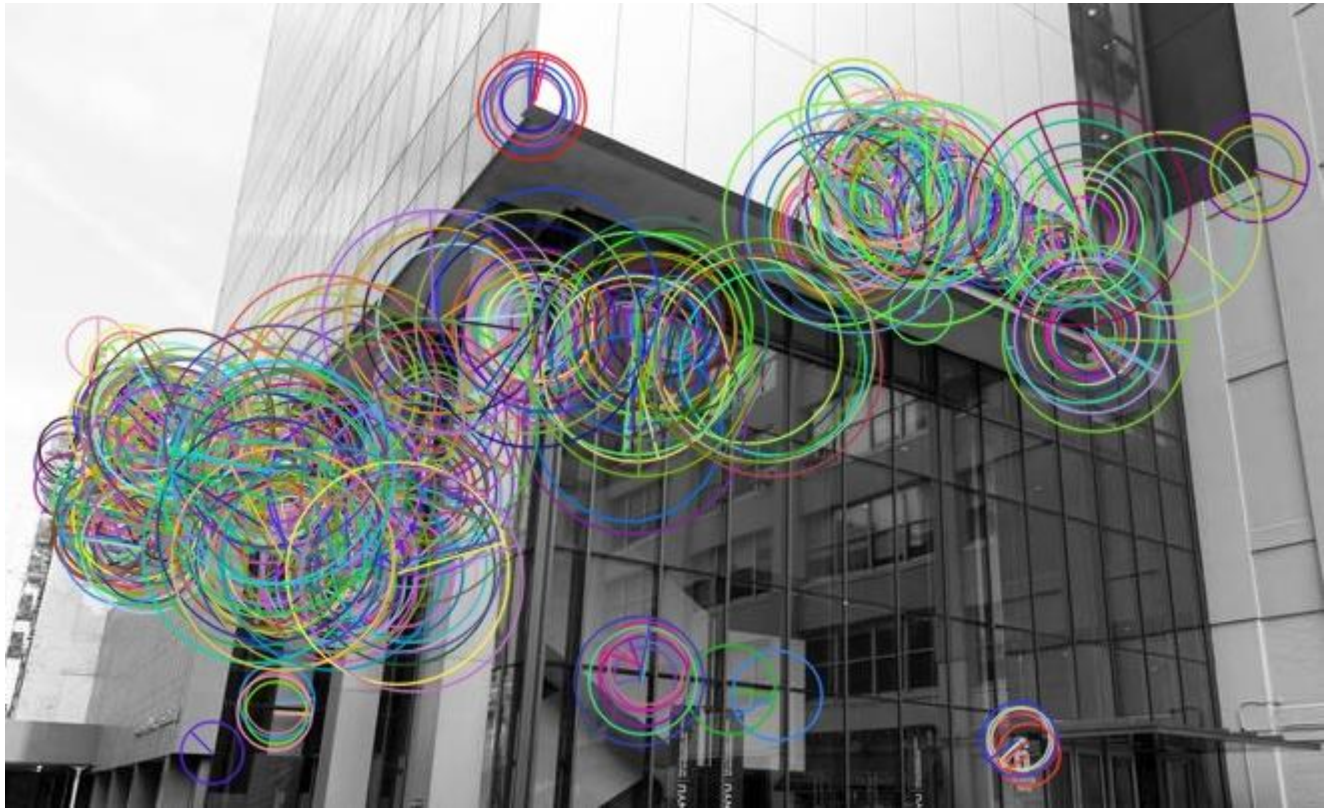
Feature Description/Matching
(SIFT/SURF/ORB)



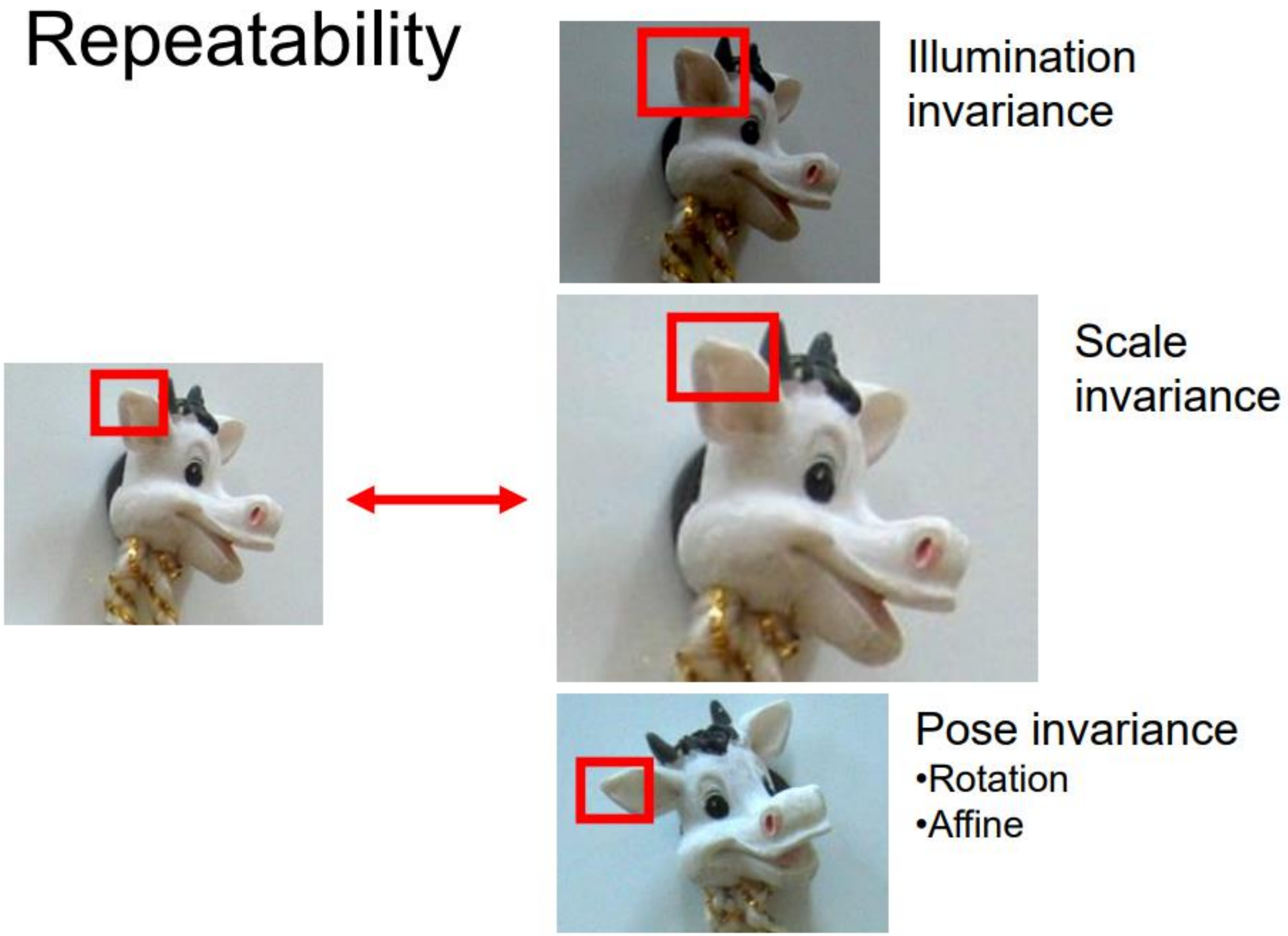
- Homography estimation
- F-matrix estimation
- PnP problem
- …

Image from: https://www.brooklynpaper.com/stories/38/43/dtg-nyu-poly-renaming-2015-10-23-bk.html

Corner/Blob Detection



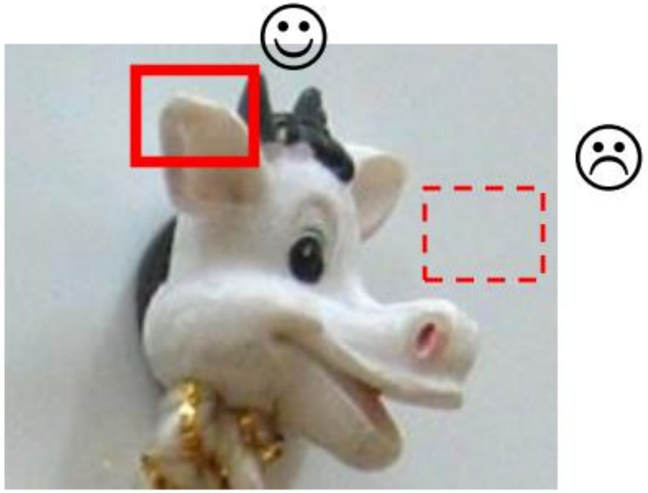
Corner Detection Criteria Illustrations



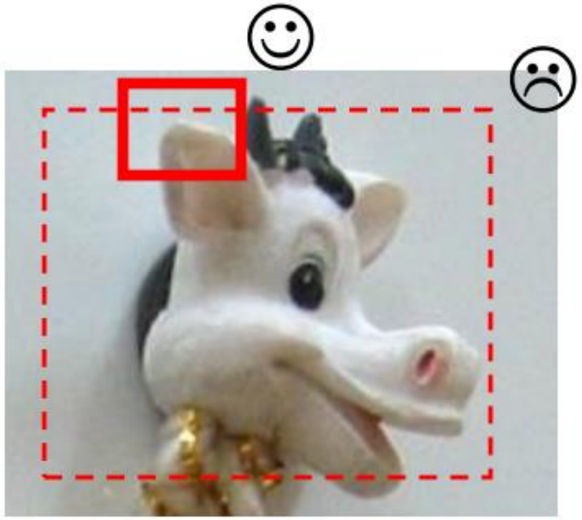
Slide from: Silvio, Stanford, 2013

Corner Detection Criteria Illustrations

- Saliency



- Locality



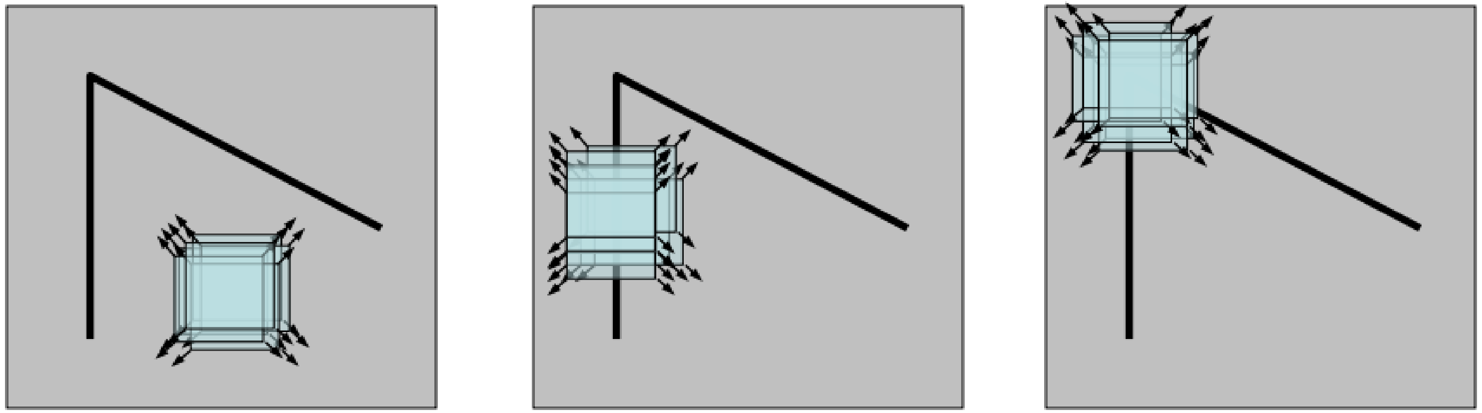
Slide from: Silvio, Stanford, 2013

Harris Corner

- C.Harris and M.Stephens. "A Combined Corner and Edge Detector.“ Proceedings of the 4th Alvey Vision Conference: pages 147--151.

Harris Detector: Basic Idea

Explore intensity changes within a window
as the window changes location



"flat" region:
no change in
all directions

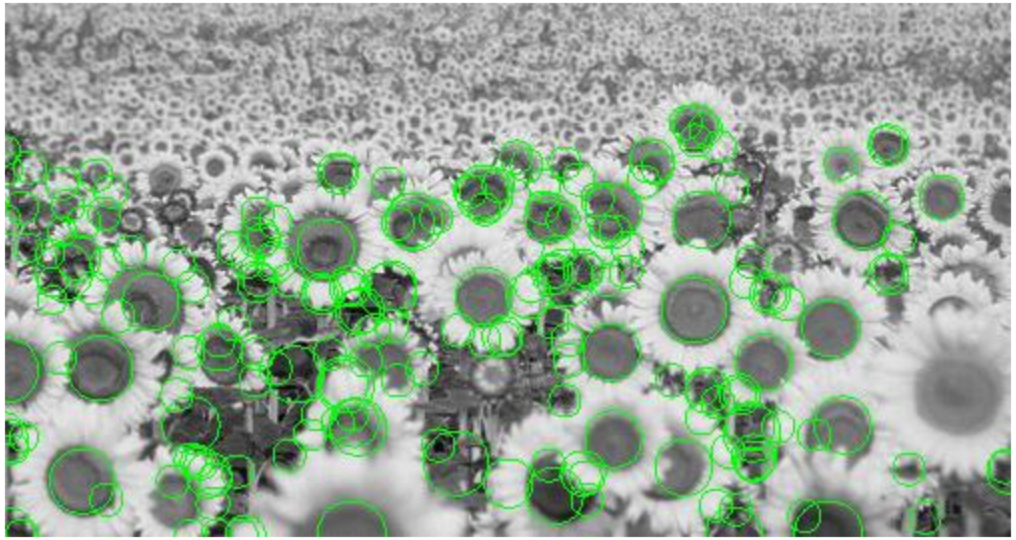
"edge":
no change
along the edge
direction

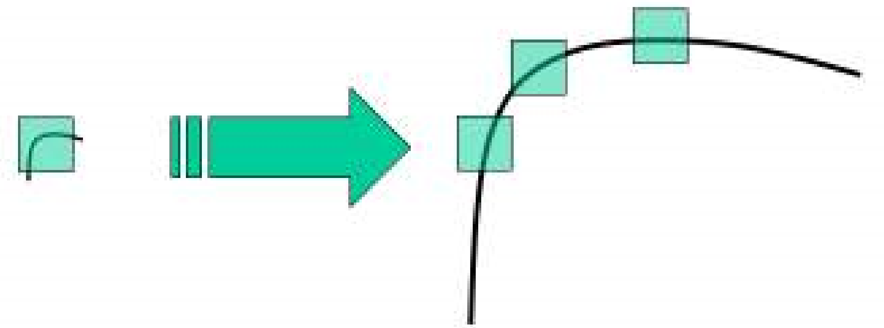
"corner":
significant
change in all
directions

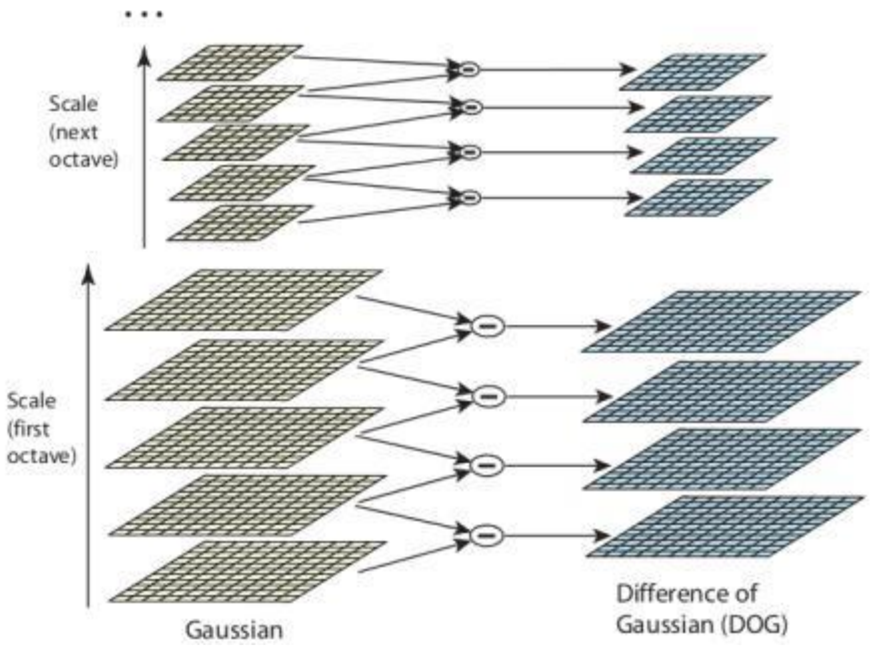
Slide from: Silvio, Stanford, 2013

Blob Detector: Difference of Gaussians (DoG)

- Harris corner is rotation invariant
  - But not scale-invariant





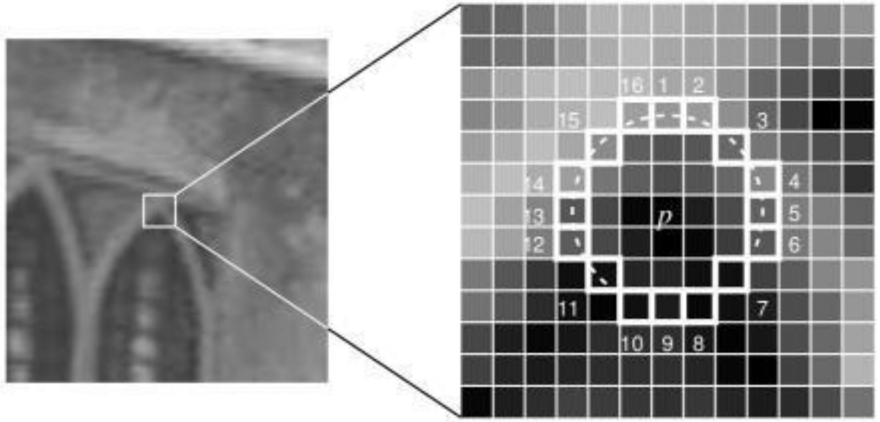


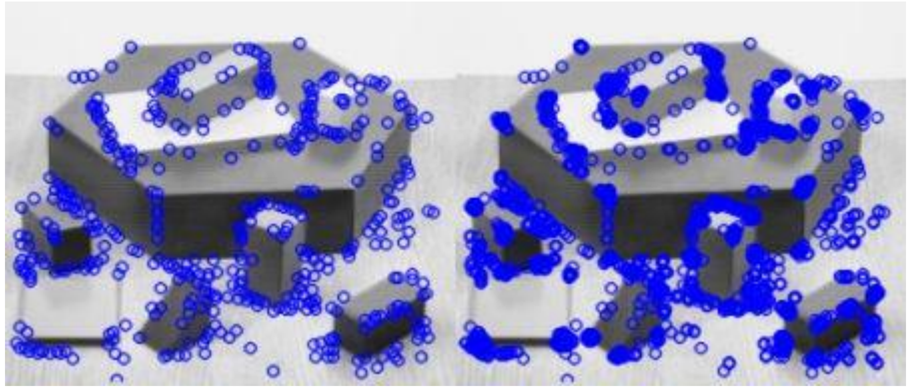
- DoG: find extrema in both 2D-space and scale-space

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, *60*(2), 91-110.

FAST: Machine Learning for Corner Detection

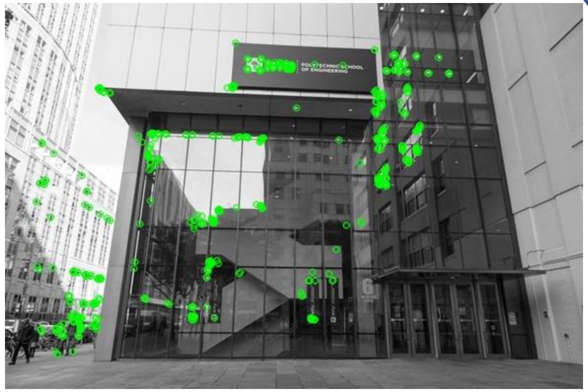
- FAST (Features from Accelerated Segment Test)
  - Corner: if there exists n=12 contiguous pixels in the circle which are all brighter or all darker than the center for a threshold t
- High-speed test
  - quickly exclude many non-corners
- Decision-tree based improvement
- Non-maximum suppression





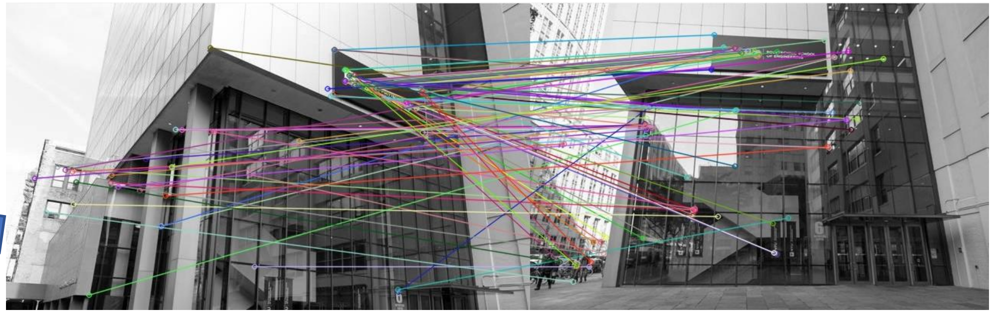
Edward Rosten, Reid Porter, and Tom Drummond, “Faster and better: a machine learning approach to corner detection” in IEEE Trans. Pattern Analysis and Machine Intelligence, 2010, vol 32, pp. 105-119.

A Classic Vision Pipeline: Description & Matching



Feature Detection
(Harris/FAST/SIFT/SURF/ORB
/LSD)

Feature Description/Matching
(SIFT/SURF/ORB)



- Homography estimation
- F-matrix estimation
- PnP problem
- …

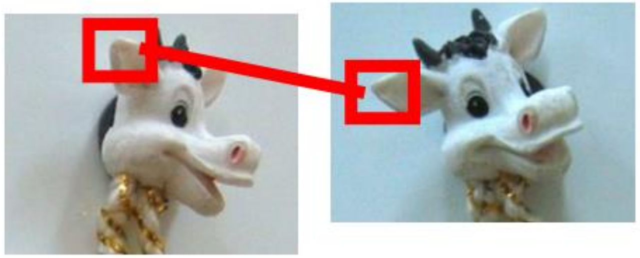
- RANSAC to reject matching outliers

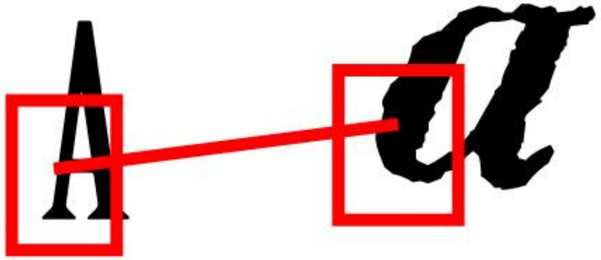
Image from: https://www.brooklynpaper.com/stories/38/43/dtg-nyu-poly-renaming-2015-10-23-bk.html

Feature Description

Depending on the application a descriptor must incorporate information that is:

- Invariant w.r.t:
  - Illumination
  - Pose
  - Scale
  - Intraclass variability





- **Highly distinctive** (allows a single feature to find its correct match with good probability in a large database of features)

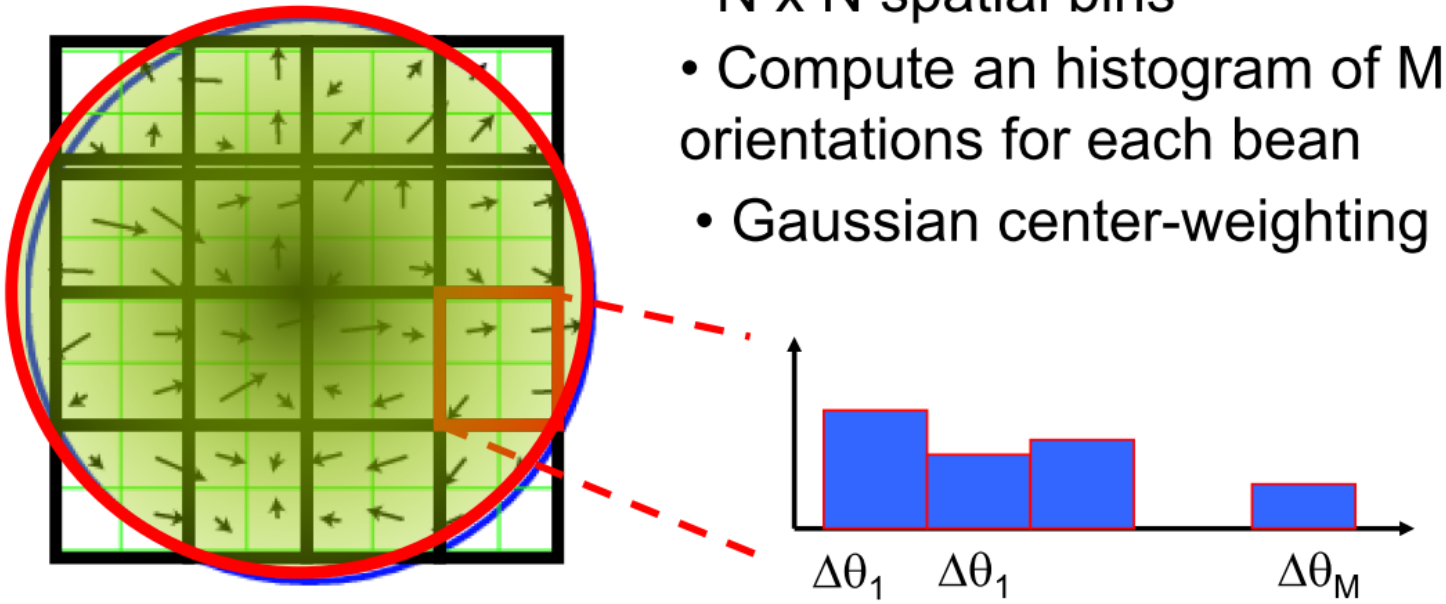
Slide from: Silvio, Stanford, 2013

SIFT Descriptor

- A standard (but non-free) descriptor
- Location and scale given by DoG detector (SIFT keypoints)

•Compute gradient at each pixel

- N x N spatial bins
- Compute an histogram of M orientations for each bean
    - Gaussian center-weighting

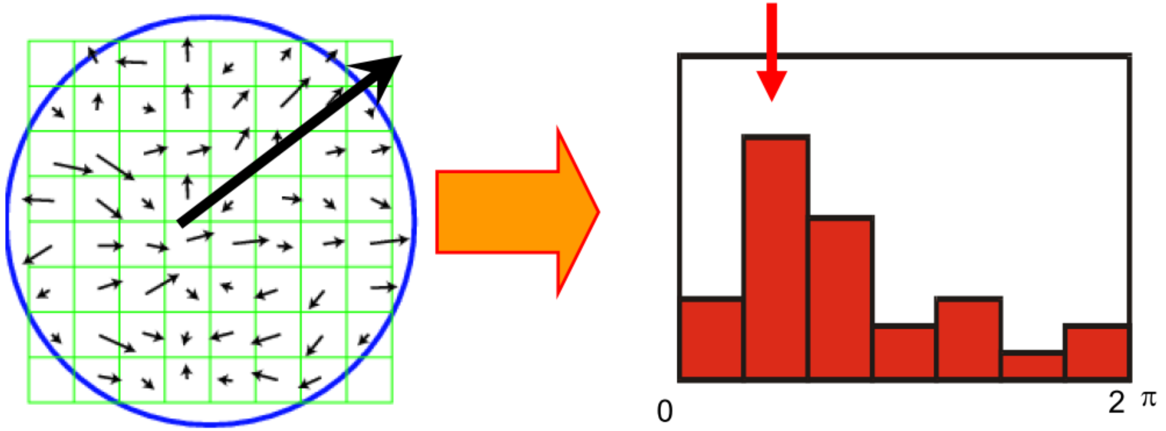


Typically M = 8; N= 4
1 x 128 descriptor

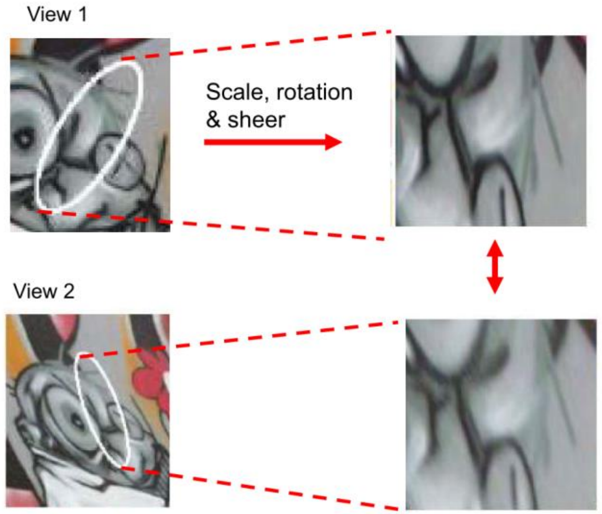
Slide modified from: Silvio, Stanford, 2013

SIFT Descriptor is Robust to Small Variations

- Illumination
  - gradient & normalization
- Pose (small affine variation)
  - orientation histogram
- Scale
  - fixed by DOG
- Intra-class variability
  - histograms (small variations)



This makes the SIFT descriptor rotational invariant

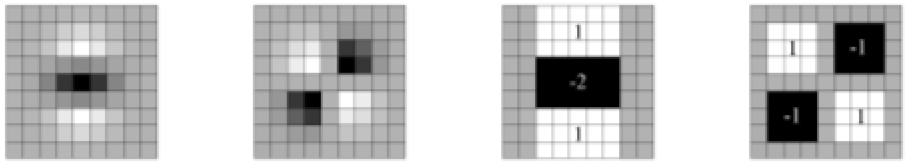


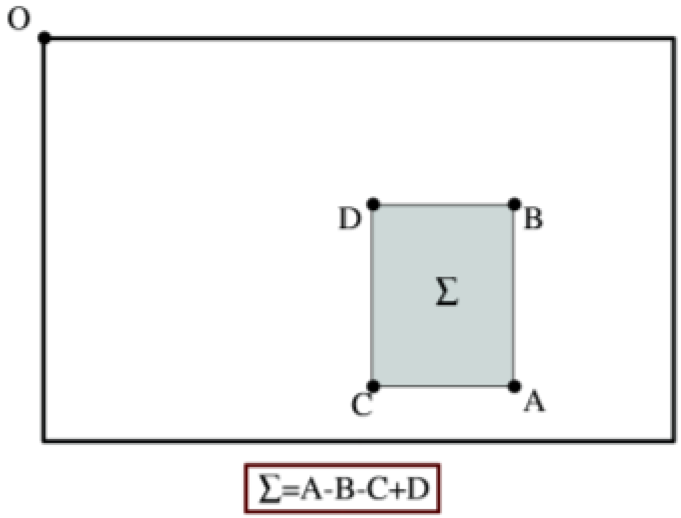
Slide modified from: Silvio, Stanford, 2013

From SIFT to SURF

- SIFT is good in terms of matching quality
  - But it is too slow for real-time applications

- SURF uses integral image to speed up the SIFT computation





Bay, H., Tuytelaars, T. and Van Gool, L., 2006, May. Surf: Speeded up robust features. In European conference on computer vision (pp. 404-417). Springer, Berlin, Heidelberg.

Faster than SURF? BRIEF/ORB

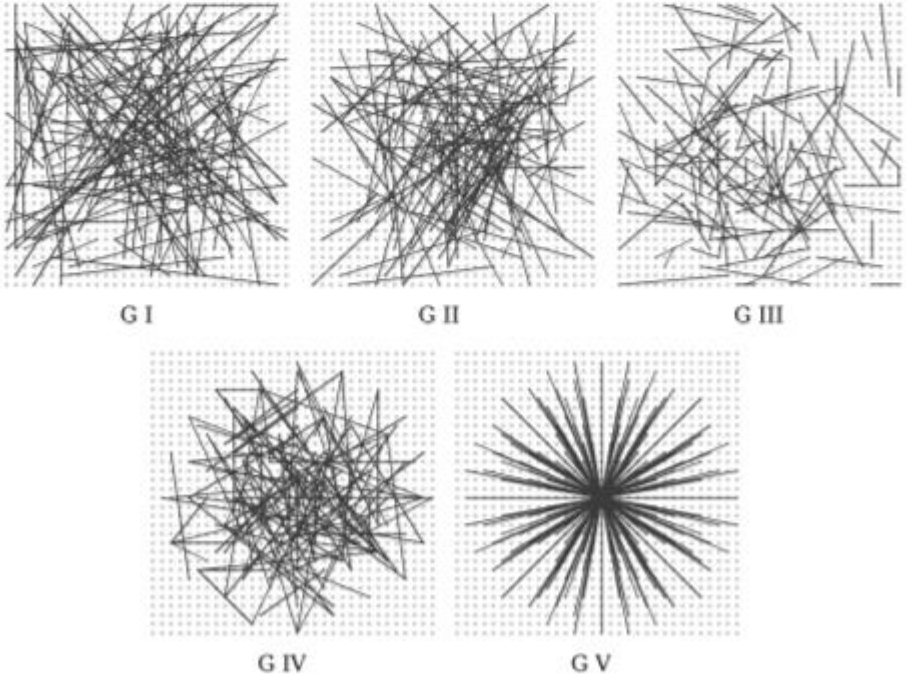
- Connecting feature description with machine learning

More specifically, we define test τ on patch $\mathbf{p}$ of size $S \times S$ as

$$\tau(\mathbf{p};\mathbf{x},\mathbf{y}) := \begin{cases} 1 & \text{if } \mathbf{p}(\mathbf{x}) < \mathbf{p}(\mathbf{y}) \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

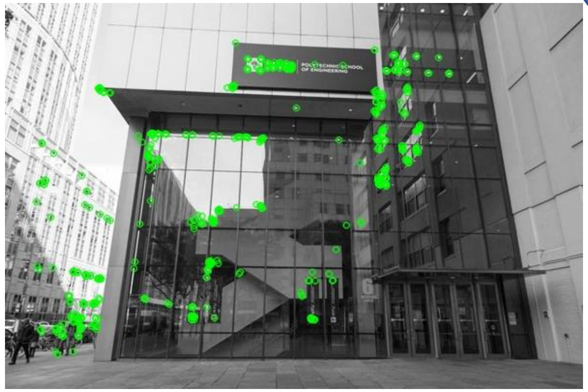
where $\mathbf{p}(\mathbf{x})$ is the pixel intensity in a smoothed version of $\mathbf{p}$ at $\mathbf{x} = (u, v)^\top$. Choosing a set of n_d $(\mathbf{x}, \mathbf{y})$-location pairs uniquely defines a set of binary tests. We take our BRIEF descriptor to be the n_d-dimensional bitstring

$$f_{n_d}(\mathbf{p}) := \sum_{1 \leq i \leq n_d} 2^{i-1}\, \tau(\mathbf{p};\mathbf{x}_i,\mathbf{y}_i) \ . \tag{2}$$



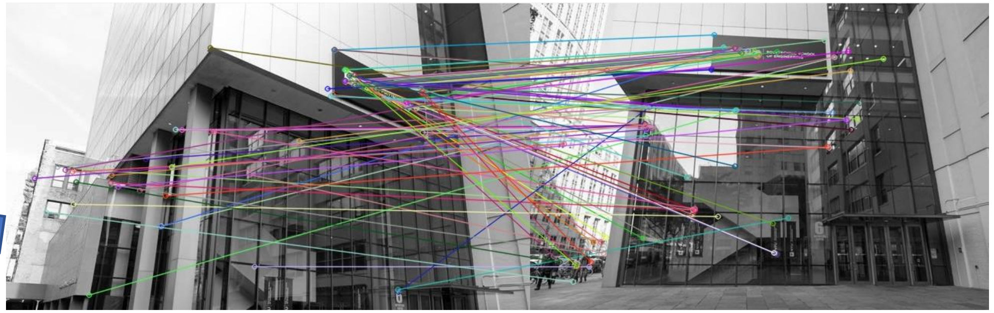
Calonder, Michael, et al. "Brief: Binary robust independent elementary features." European conference on computer vision. Springer, Berlin, Heidelberg, 2010.

How to Handle Wrong Matching?



Feature Detection
(Harris/FAST/SIFT/SURF/ORB
/LSD)

Feature Description/Matching
(SIFT/SURF/ORB)

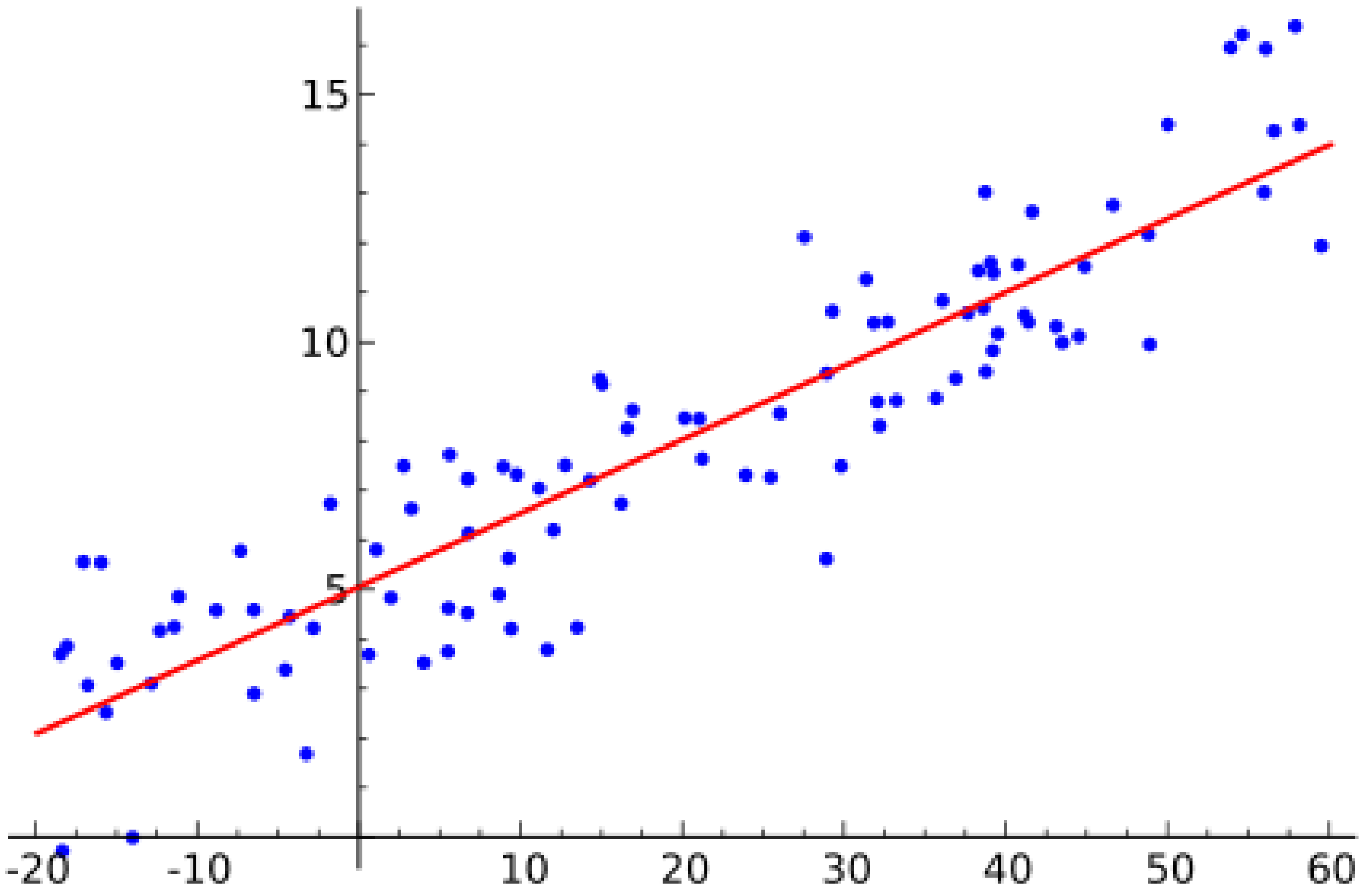


- Homography estimation
- F-matrix estimation
- PnP problem
- …

- **RANSAC to reject matching outliers**

Image from: https://www.brooklynpaper.com/stories/38/43/dtg-nyu-poly-renaming-2015-10-23-bk.html

Let’s start from fitting a 2D line

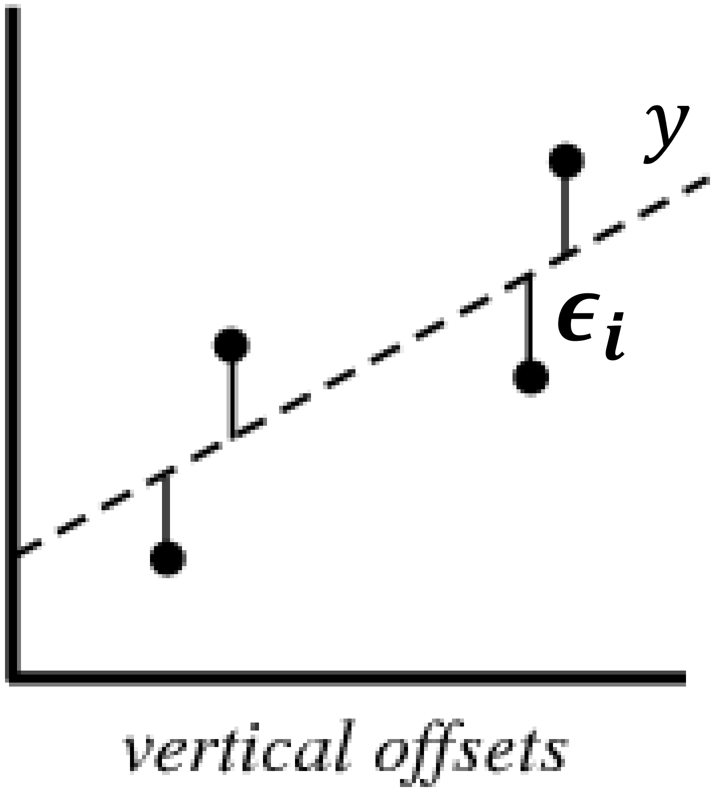


Linear Regression (Linear Least Squares)

$y = ax + b$

$\boldsymbol{\epsilon_i}$

$\boldsymbol{y}$



$\boldsymbol{x}$

$$\boldsymbol{\epsilon} = \boldsymbol{y} - a\boldsymbol{x} - b$$

$$[a, b]^* = \underset{a,b}{\operatorname{argmin}} \|\boldsymbol{\epsilon}\|^2$$

Image from: http://mathworld.wolfram.com/LeastSquaresFitting.html

Solving Ax=b

- A: design matrix
  - shape: m x n
  - m>>n
  - Typically full column-rank
- x: unknowns
  - shape: n x 1
- b: observed data
  - shape: m x 1
- Solve by least squares: x* = inv(A'A)A'b
  - Solving normal equation: A'Ax=A'b

Orthogonal Regression (Total Least Squares)

$$\epsilon_i = \boldsymbol{n}^T(\boldsymbol{p}_i - \boldsymbol{q})$$

$$[\boldsymbol{n}, \boldsymbol{q}]^* = \underset{\boldsymbol{n},\boldsymbol{q}}{\operatorname{argmin}} \sum_{\boldsymbol{i}} \|\epsilon_i\|^{\mathbf{2}}$$
$$s.t. \|\boldsymbol{n}\|^{\mathbf{2}} = 1$$

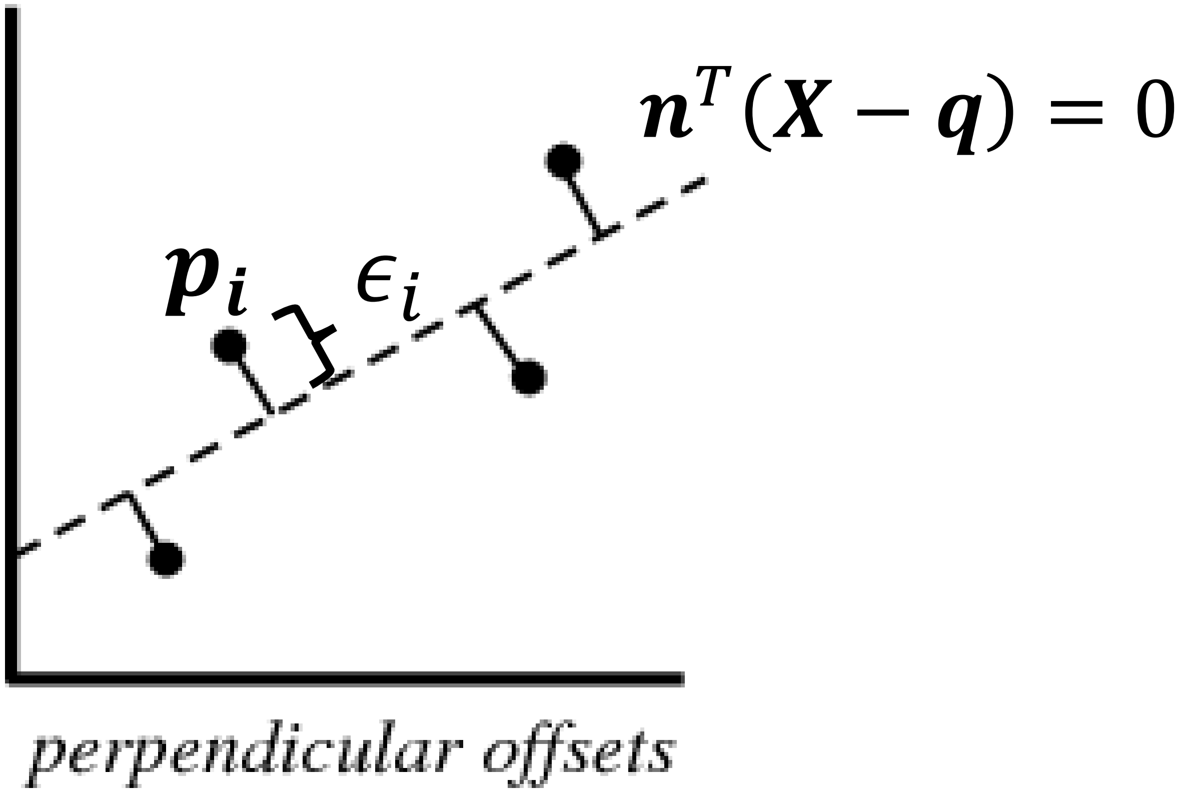


Image from: http://mathworld.wolfram.com/LeastSquaresFitting.html

Orthogonal Regression for Line Estimation

- Given a set of 3D points $\{\boldsymbol{p}_i\}$, we want to find out a line/plane (i.e., unit normal **n** and center **q**) that describes this set of points as

$$n^\top(p_i - q) = 0, \forall i$$

$$\begin{aligned}
\mathrm{cost}(n, q) &\triangleq \sum_i \mathrm{dist}^2(p_i; n, q) \\
&= \sum_i (n^\top(p_i - q))^2 \\
&= n^\top[\cdots, p_i - q, \cdots][\cdots, p_i^\top - q^\top, \cdots]^\top n \\
&= n^\top A(q) A(q)^\top n
\end{aligned}$$

Orthogonal Regression for Line Estimation

- Solving **q**

$$\mathbf{0} = \frac{\partial \text{cost}(n, q)}{\partial q} \equiv \sum_i (2nn^\top q - 2nn^\top p_i)$$

$$q^* = \frac{1}{|\{p_i\}|} \sum_i p_i$$

Orthogonal Regression for Line Estimation

- Solving **n**

$$\mathrm{cost}(n; q^*) \triangleq n^\top A(q^*)A(q^*)^\top n = n^\top B(q^*)n$$

- Equivalent to solve:

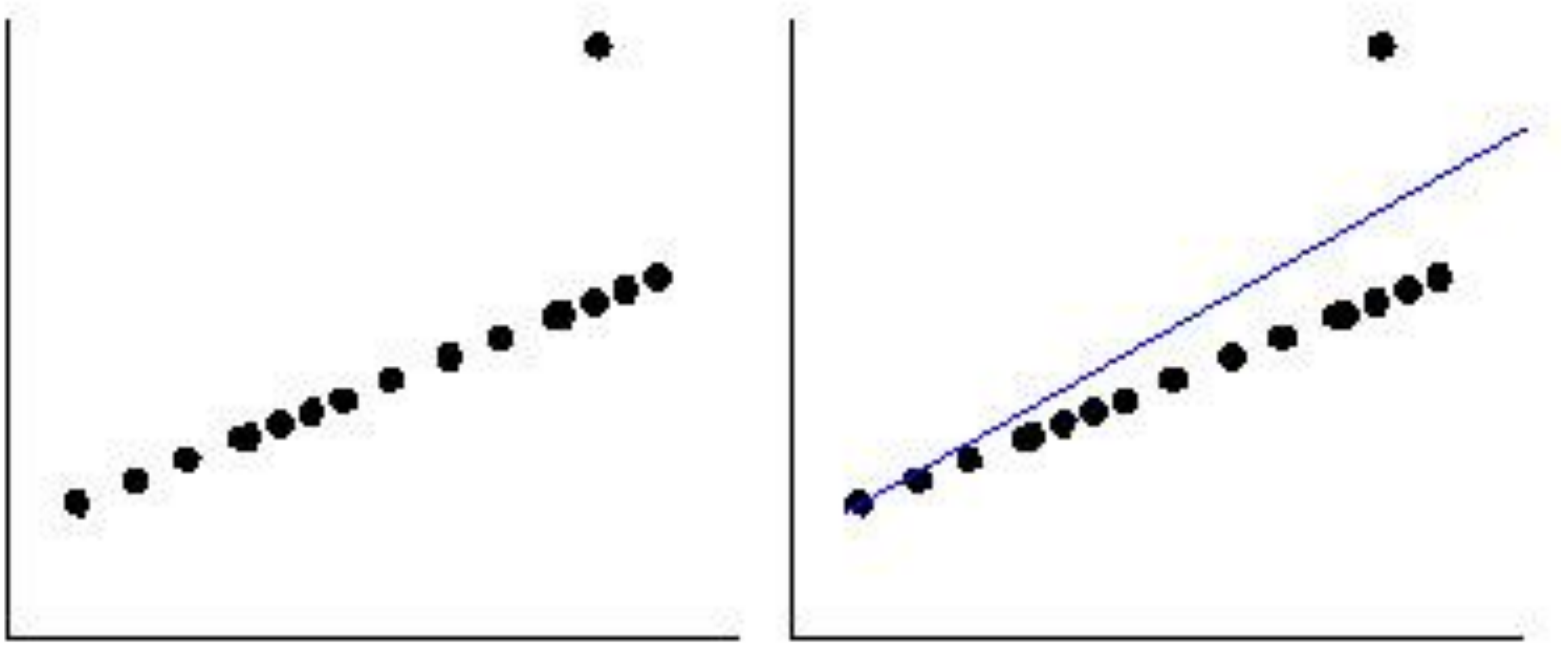
$$n^* = \arg\min_n \quad n^\top B(q^*)n \qquad \text{s.t.} \quad n^\top n = 1$$

- Solve by SVD
  - Optimal **n** is **B**’s eigenvector corresponding to the smallest eigenvalue.
  - So, this is also referred to as the PCA-based solution.

Solving Ax=0

- A: data matrix
  - shape: m x n
  - m >> n
  - rank(A) = n when data contains noise: full column-rank
- x: unknowns
  - shape: n x 1
- Seek for an approximation instead of exact non-trivial solutions
- Add a constraint: ||x||=1
- Solve by SVD: A=UDV’
  - x*=last column of V,   if diag(D) is descending order

But Least Squares is NOT Robust to Outliers!



http://www.unige.ch/ses/sococ/cl//////stat/action/nonlin5.jpg

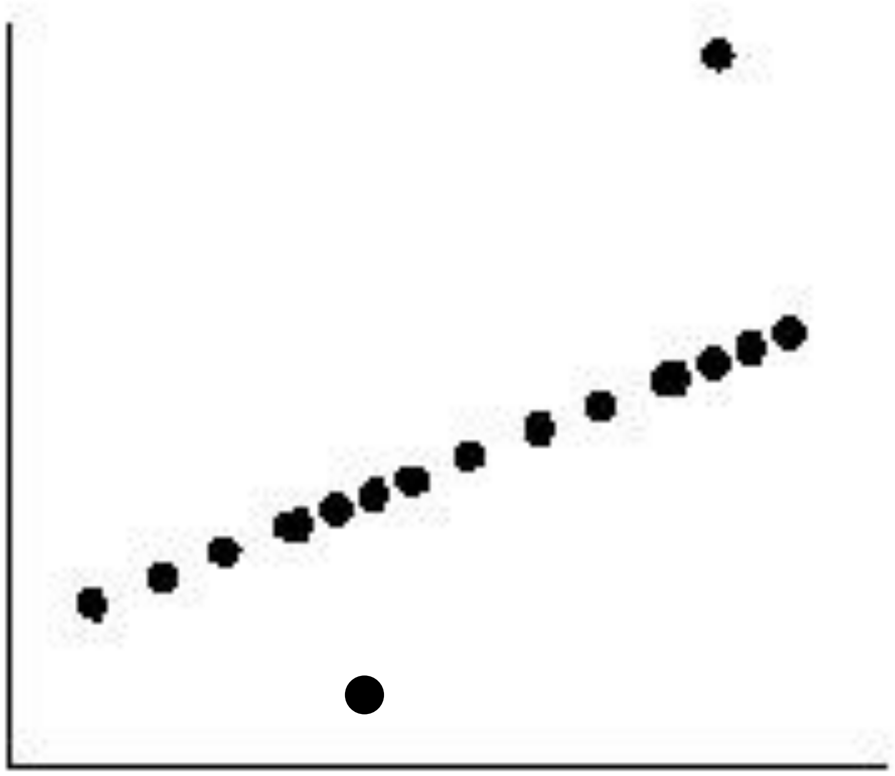
How to Solve This Intuitively?

- Mitigating outlier's influence/weight in the estimation
  - Iteratively Re-weighted Least Squares (IRLS)

- Detect outlier and remove it from estimation
  - RANdom SAmple Consensus (RANSAC)

How to Detect an Outlier?

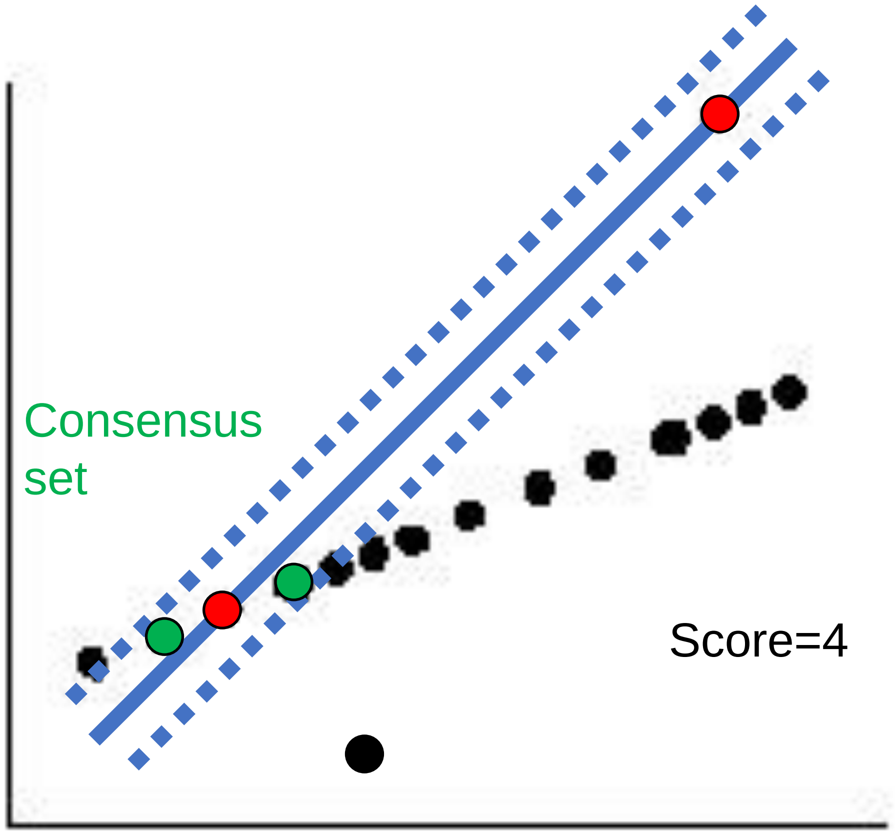
- Enumeration strategy
  - Leave one out
- Voting strategy
  - Hypothesis and test



http://www.unige.ch/ses/sococ/cl//////stat/action/nonlin5.jpg

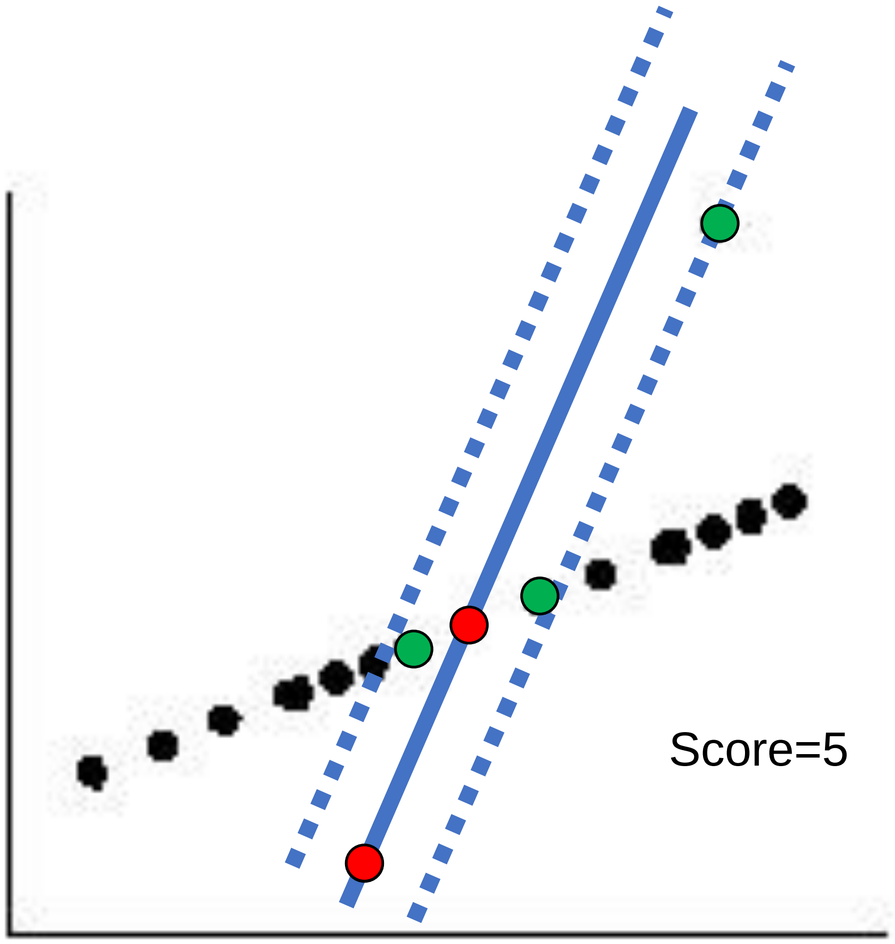
Hypothesis and Test

- Randomly select two points to generate a hypothesis line
- Test how good the current hypothesis is
  - Consensus set = {supporting points}
  - Score by #(supporting points)= |Consensus set|



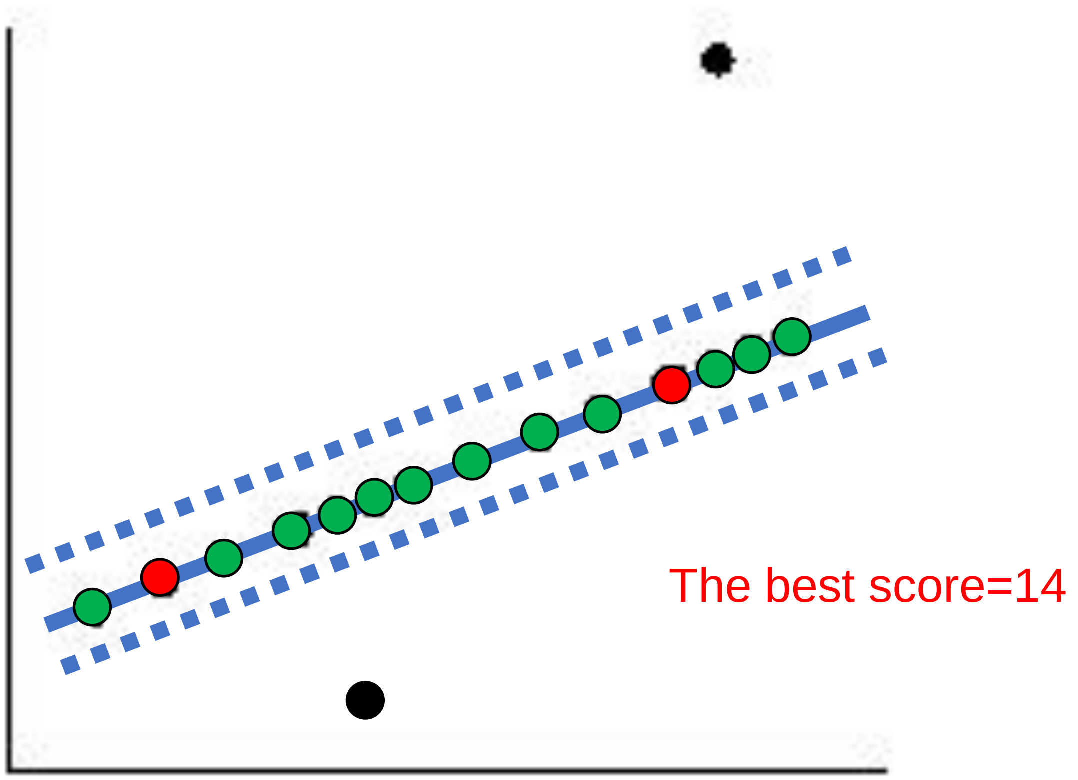
Hypothesis and Test

- Randomly select two points to generate a hypothesis line
- Test how good the current hypothesis is



Hypothesis and Test

- Randomly select two points to generate a hypothesis line
- Test how good the current hypothesis is



RANSAC Framework

Objective

Line/Plane/Homography/etc.

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

Algorithm

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

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

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

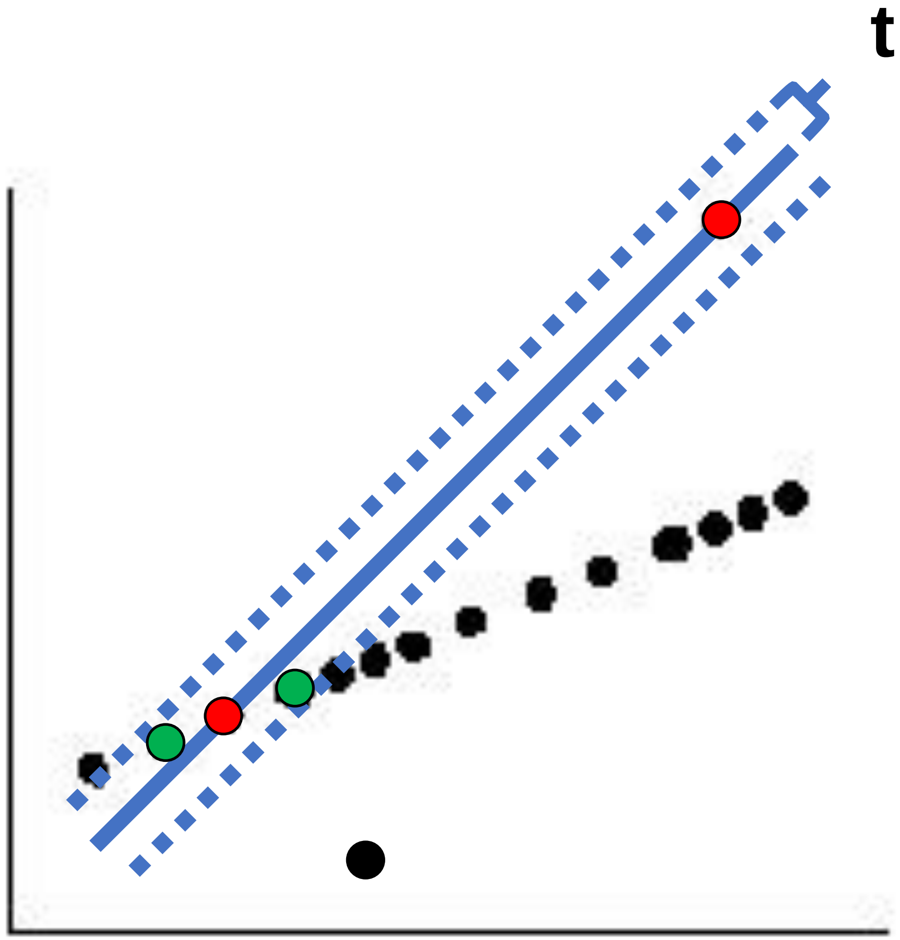
(iv) If the size of S_i is less than T, select a new subset and repeat the above.

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

Often simplified as:
(iii) Update S_i if the size grows.
(iv) repeat (i), (ii), (iii) for N times.

Hartley & Zisserman 2003

RANSAC Details – Distant Threshold t



- If data is known to be distributed as a Gaussian of standard deviation σ:
  - Use the 3σ rule
- Otherwise:
  - Determined manually from experience
  - Try-and-error

RANSAC Details – #Samples N

- Basic idea: more #trials (N) is better
  - Because it means larger RANSAC success probability p (**at least one** successful sampling)
  - Success sample: all the s sampled data points are inliers

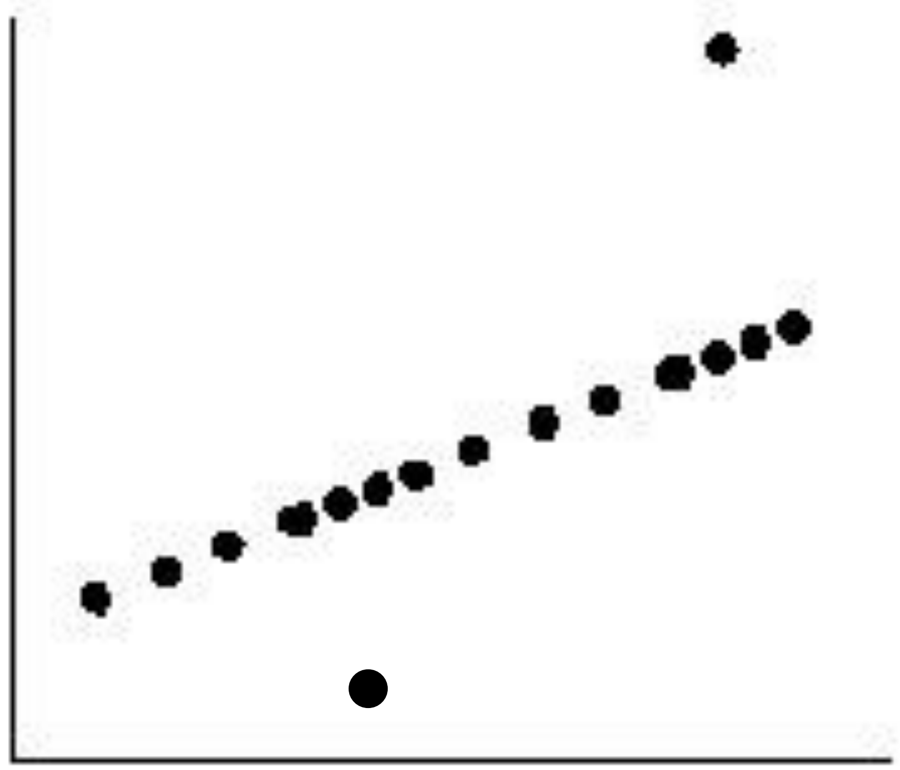
- If we want to achieve a particular RANSAC success probability p
- And assume the outlier ratio ϵ is known
- How many #trials do we need i.e., $\mathrm{N} = ?$

- Probability of one successful sampling: $(1-\epsilon)^s$
- Probability of at least one successful sampling: $1-[1-(1-\epsilon)^s]^N$
- Thus: $\mathrm{N} = \ln(1-p) \,/\, \ln[1-(1-\epsilon)^s]$

Why Minimal Solution is Important

- For p=0.99

Sample size	Proportion of outliers ϵ						
s	5%	10%	20%	25%	30%	40%	50%
2	2	3	5	6	7	11	17
3	3	4	7	9	11	19	35
4	3	5	9	13	17	34	72
5	4	6	12	17	26	57	146
6	4	7	16	24	37	97	293
7	4	8	20	33	54	163	588
8	5	9	26	44	78	272	1177



Hartley & Zisserman 2003

RANSAC Details – Adaptive Sampling

- What if outlier ratio ϵ is NOT known?

- $N = \infty$, sample_count$= 0$.
- While $N >$ sample_count Repeat
  - Choose a sample and count the number of inliers.
  - Set $\epsilon = 1 - (\text{number of inliers})/(\text{total number of points})$
  - Set N from ϵ and (4.18) with $p = 0.99$.
  - Increment the sample_count by 1.
- Terminate.

Hartley & Zisserman 2003

More Examples – Homography

Objective

Compute the 2D homography between two images.

Algorithm

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

(ii) **Putative correspondences:** Compute a set of interest point matches based on proximity and similarity of their intensity neighbourhood.

(iii) **RANSAC robust estimation:** Repeat for N samples, where N is determined adaptively as in algorithm 4.5:

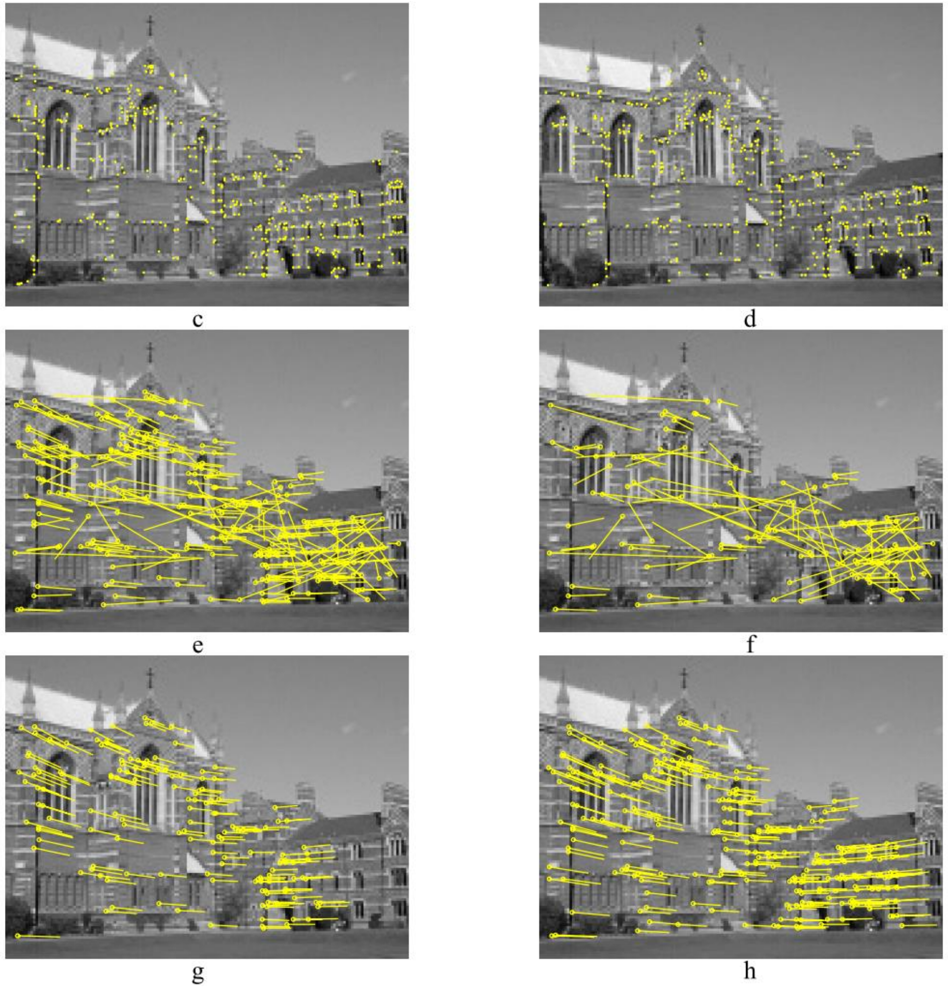
- (a) Select a random sample of 4 correspondences and compute the homography $\mathtt{H}$.
- (b) Calculate the distance $d_\perp$ for each putative correspondence.
- (c) Compute the number of inliers consistent with $\mathtt{H}$ by the number of correspondences for which $d_\perp < t = \sqrt{5.99}\,\sigma$ pixels.

Choose the $\mathtt{H}$ with the largest number of inliers. In the case of ties choose the solution that has the lowest standard deviation of inliers.

(iv) **Optimal estimation:** re-estimate $\mathtt{H}$ from all correspondences classified as inliers, by minimizing the ML cost function (4.8–*p*95) using the Levenberg–Marquardt algorithm of section A6.2(*p*600).

(v) **Guided matching:** Further interest point correspondences are now determined using the estimated $\mathtt{H}$ to define a search region about the transferred point position.

The last two steps can be iterated until the number of correspondences is stable.



Hartley & Zisserman 2003

More Examples – F-matrix

Objective Compute the fundamental matrix between two images.

Algorithm

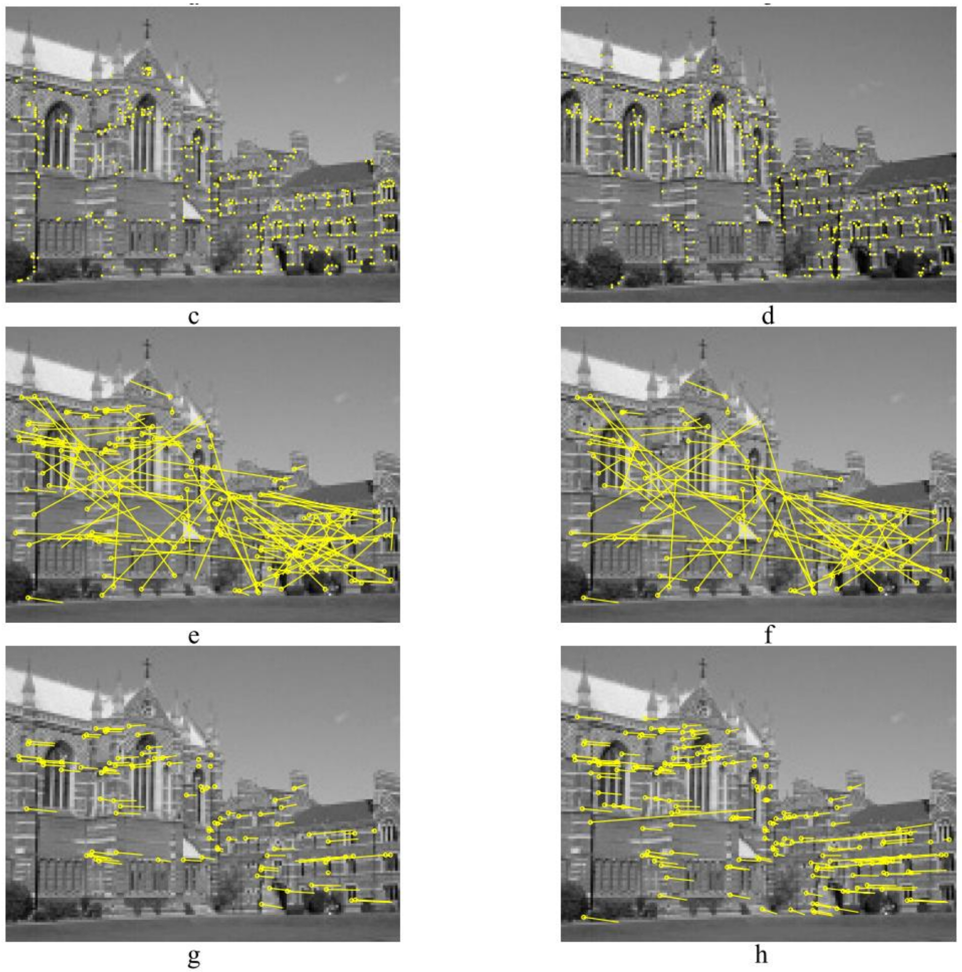
(i) **Interest points:** Compute interest points in each image.
(ii) **Putative correspondences:** Compute a set of interest point matches based on proximity and similarity of their intensity neighbourhood.
(iii) **RANSAC robust estimation:** Repeat for N samples, where N is determined adaptively as in algorithm 4.5(p121):

(a) Select a random sample of 7 correspondences and compute the fundamental matrix $\mathtt{F}$ as described in section 11.1.2. There will be one or three real solutions.
(b) Calculate the distance $d_\perp$ for each putative correspondence.
(c) Compute the number of inliers consistent with $\mathtt{F}$ by the number of correspondences for which $d_\perp < t$ pixels.
(d) If there are three real solutions for $\mathtt{F}$ the number of inliers is computed for each solution, and the solution with most inliers retained.

Choose the $\mathtt{F}$ with the largest number of inliers. In the case of ties choose the solution that has the lowest standard deviation of inliers.

(iv) **Non-linear estimation:** re-estimate $\mathtt{F}$ from all correspondences classified as inliers by minimizing a cost function, e.g. (11.6), using the Levenberg–Marquardt algorithm of section A6.2(p600).
(v) **Guided matching:** Further interest point correspondences are now determined using the estimated $\mathtt{F}$ to define a search strip about the epipolar line.

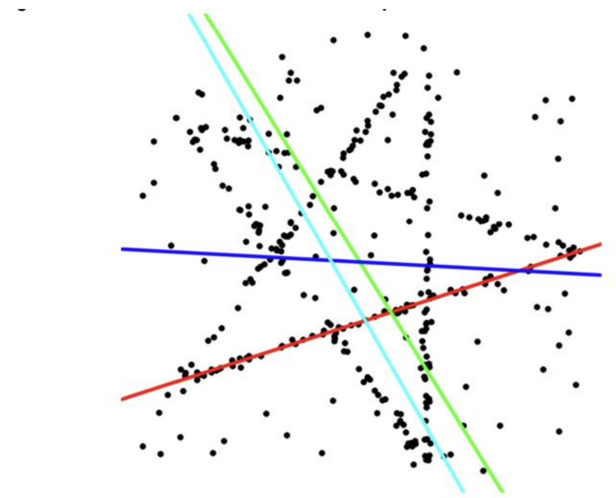
The last two steps can be iterated until the number of correspondences is stable.



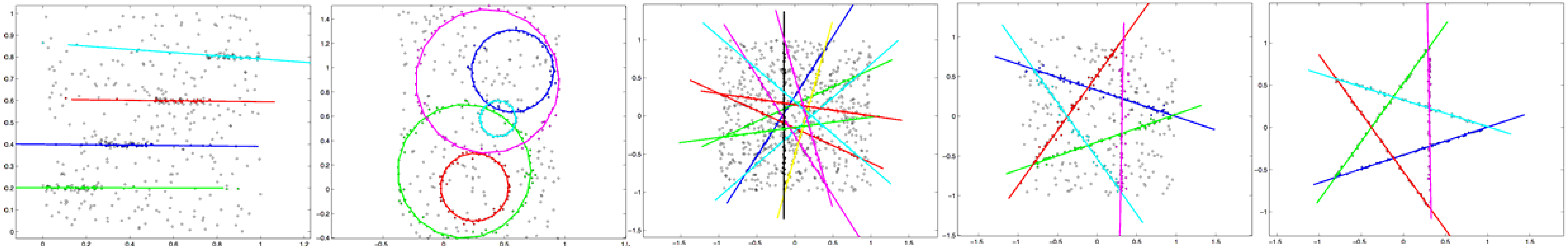
Hartley & Zisserman 2003

Issues in RANSAC

- RANSAC could be **time-consuming**
  - Too many trials
- RANSAC will fail if the problem is **multi-model**
  - i.e., the data is sampled from multiple models (lines/planes/homography/…)

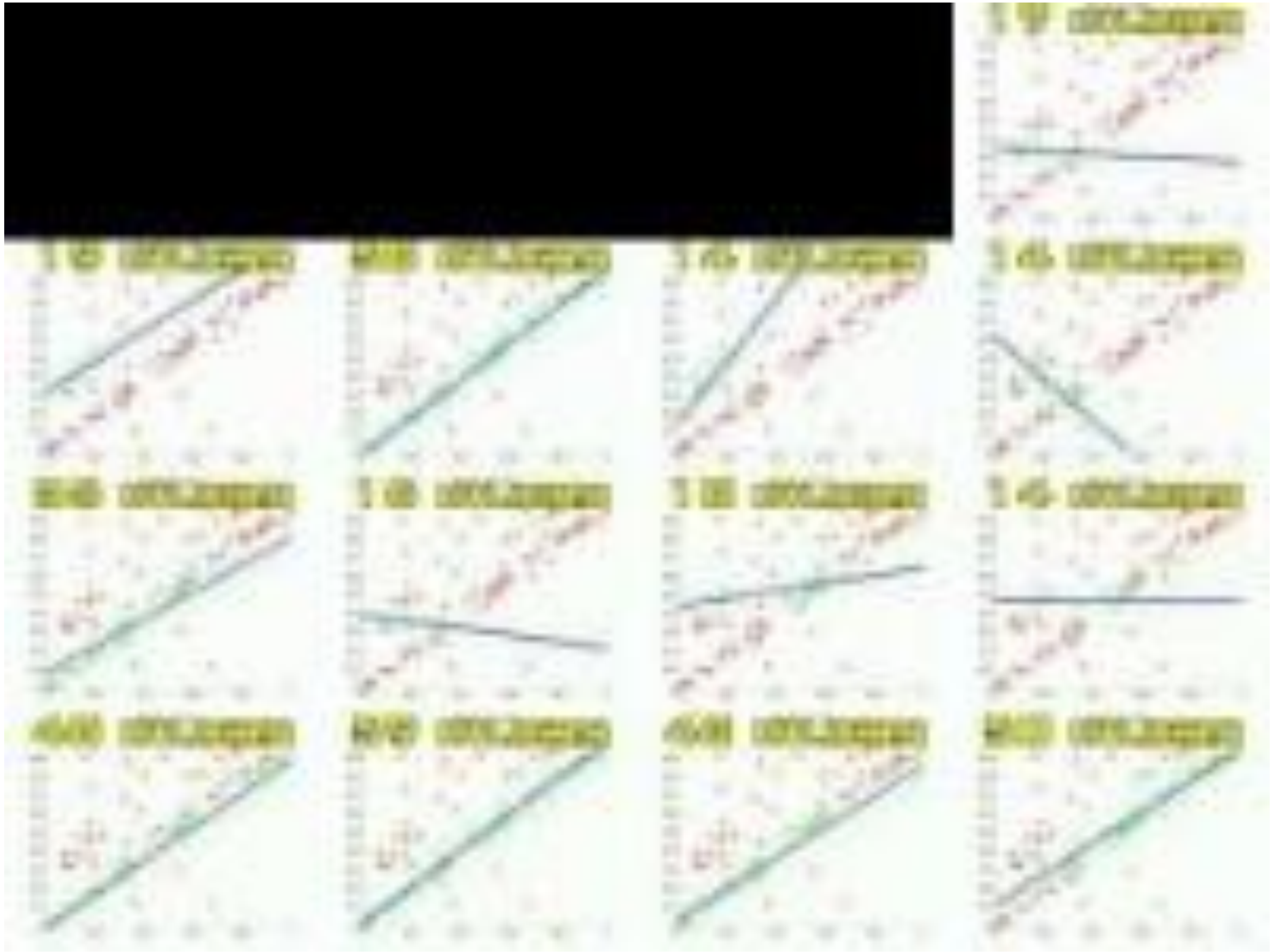


Ordinary Least Squares (O.L.S.), Total Least Squares (T.L.S.) (via PCA), Least Median of Squares (LMedS), Random Sample Consensus (RANSAC)



Images from: R. Toldo and L. Magri (http://www.diegm.uniud.it/fusiello/demo/jlk/), and David F. Fouhey (https://web.eecs.umich.edu/~fouhey/earlier/thesis/dfouhey_thesisPresentation.pdf)

The RANSAC Song



https://youtu.be/1YNjMxxXO-E

References

- Szeliski 2011
  - Section 14.4

- (BoVW)Jupyter notebook by Olga Vysotska: https://github.com/ovysotska/in_simple_english/bl ob/master/bag_of_visual_words.ipynb
- (VLAD) Jégou, H., Douze, M., Schmid, C. and Pérez, P., 2010, June. Aggregating local descriptors into a compact image representation. In *2010 IEEE computer society conference on computer vision and pattern recognition* (pp. 3304-3311). IEEE.
- (NetVLAD) Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T. and Sivic, J., 2016. NetVLAD: CNN architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5297-5307).
- Kendall, A., Grimes, M. and Cipolla, R., 2015. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision* (pp. 2938-2946).