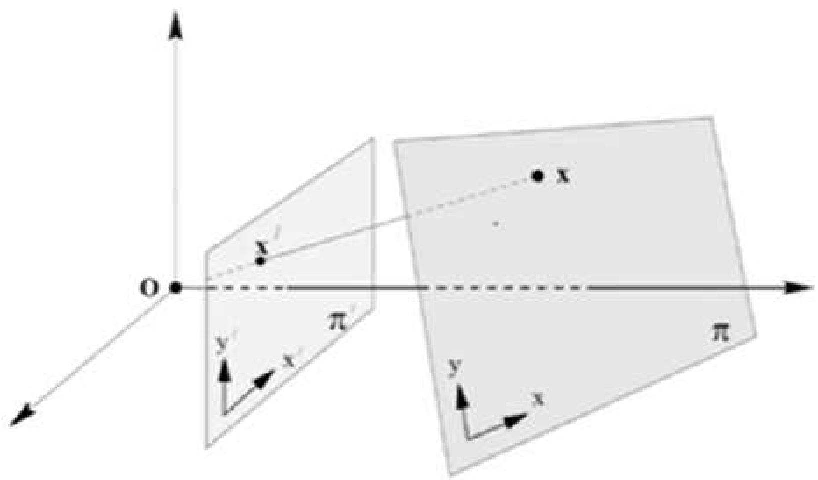
Projective Homography

RANSAC Algorithm

Some slides were adapted/taken from various sources, including 3D Computer Vision of Prof. Hee, NUS, Air Lab Summer School, The Robotic Institute, CMU, Computer Vision of Prof. Mubarak Shah, UCF, Computer Vision of Prof. William Hoff, Colorado School of Mines and many more. We thankfully acknowledge them. Students are requested to use this material for their study only and **NOT** to distribute it.

Planner Projective Transformations

- Central projection maps points on one plane to points on another plane.
- And represented by a linear mapping of homogeneous coordinates $\mathbf{x}' = H\mathbf{x}$.

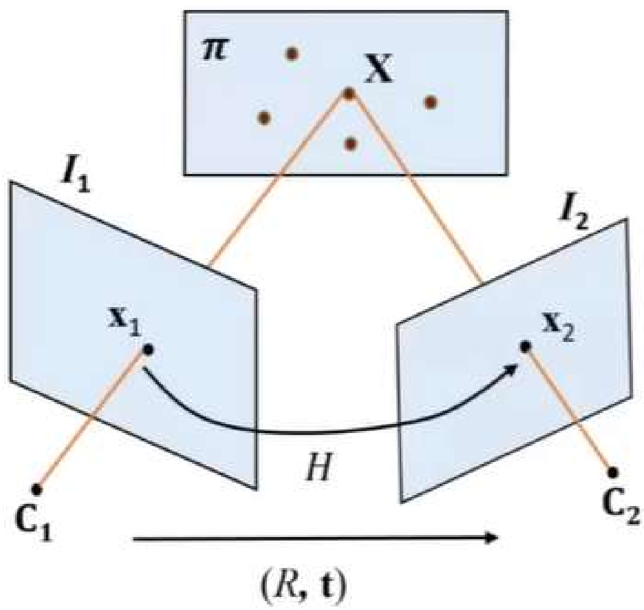


This is also known as Homography!

Image source: "Multiple View Geometry in Computer Vision", Richard Hartley and Andrew Zisserman

Existence of Projective Homography

1. Planar scene:



- $\mathbf{X}_1$ and $\mathbf{X}_2$ is the 3D point $\mathbf{X}$ expressed in $\mathbf{C}_1$ and $\mathbf{C}_2$ respectively:

$$\mathbf{X}_2 = R\mathbf{X}_1 + \mathbf{t}.$$

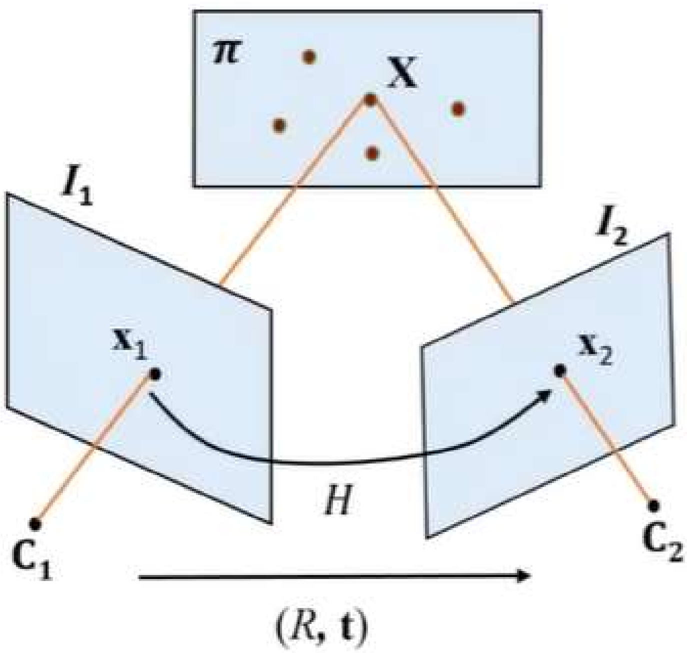
- $\mathbf{N} = [n_1, n_2, n_3]^T$ is the unit normal vector representing the plane $\boldsymbol{\pi}$ w.r.t $\mathbf{C}_1$, and d is the perpendicular distance from plane to $\mathbf{C}_1$:

$$\mathbf{N}^T\mathbf{X}_1 = n_1X + n_2Y + n_3Z = d,$$

$$\Rightarrow \frac{\mathbf{N}^T\mathbf{X}_1}{d} = 1, \ \forall \, \mathbf{X}_1 \in \boldsymbol{\pi}.$$

Existence of Projective Homography

1. Planar scene:



- Combining the two equations, we get

$$\mathbf{X}_2 = \left(R + \frac{\mathbf{tN}^T}{d}\right)\mathbf{X}_1,$$

- Since $\lambda_1 \mathbf{x}_1 = \mathbf{X}_1$ and $\lambda_2 \mathbf{x}_2 = \mathbf{X}_2$, we get

$$\lambda \mathbf{x}_2 = \underbrace{\left(R + \frac{\mathbf{tN}^T}{d}\right)}_{H}\mathbf{x}_1$$

Existence of Projective Homography

2. Plane at infinity: Scene is very far away from the camera, e.g. aerial images, i.e.

$$H = \left(R + \frac{\mathbf{tN}^T}{d}\right) \Rightarrow H_\infty = \lim_{d \to \infty} \left(R + \frac{\mathbf{tN}^T}{d}\right) = R.$$

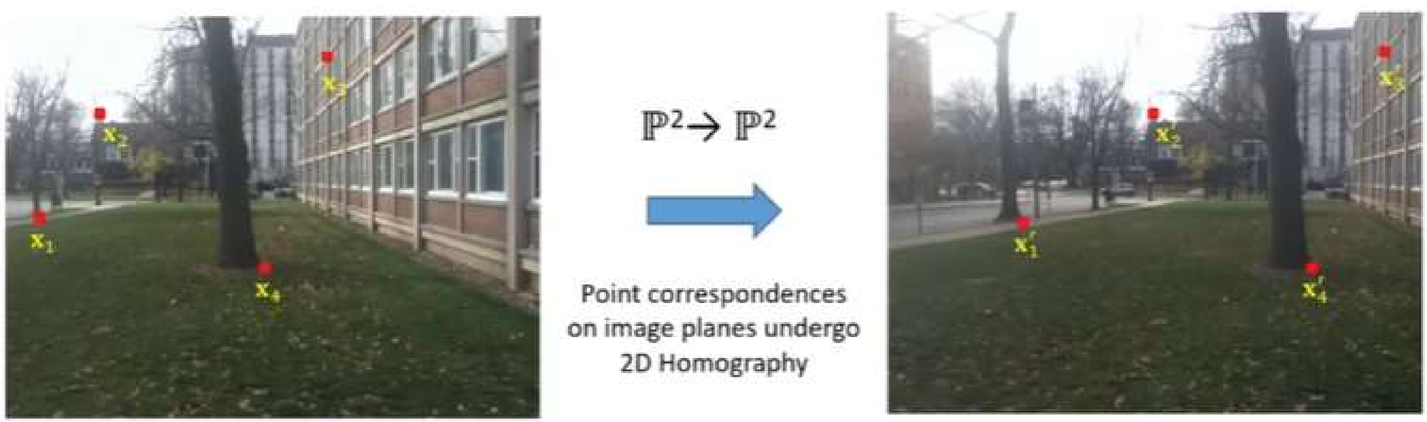
This is the same as pure rotation, i.e. $\mathbf{t} = (0,0,0)^T$:

$$H = \left(R + \frac{\mathbf{tN}^T}{d}\right) \quad \Rightarrow H = R.$$

To continue…

2D Homography

- **Given**: A set of points correspondences $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ between two images.
- **Compute**: The 2D Homography, *H* such that $H\mathbf{x}_i = \mathbf{x}'_i$ for each i.



Number of measurements required?

Question:

How many corresponding points $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ are required to compute H?

Number of measurements required?

Answer:

- The number of degrees of freedom and number of constraints give a lower bound:

1. 8 degrees of freedom for H, i.e. 9 entries less 1 for up to scale.
2. We will see that each point correspondence $\mathbf{x}_i \leftrightarrow \mathbf{x}_i'$ gives 2 constraints.

- As a consequence, it is necessary to specify four point correspondences in order to constrain H fully.

Approximate Solution

- It will be seen that if exactly four correspondences are given, then an exact solution for the matrix H is possible.

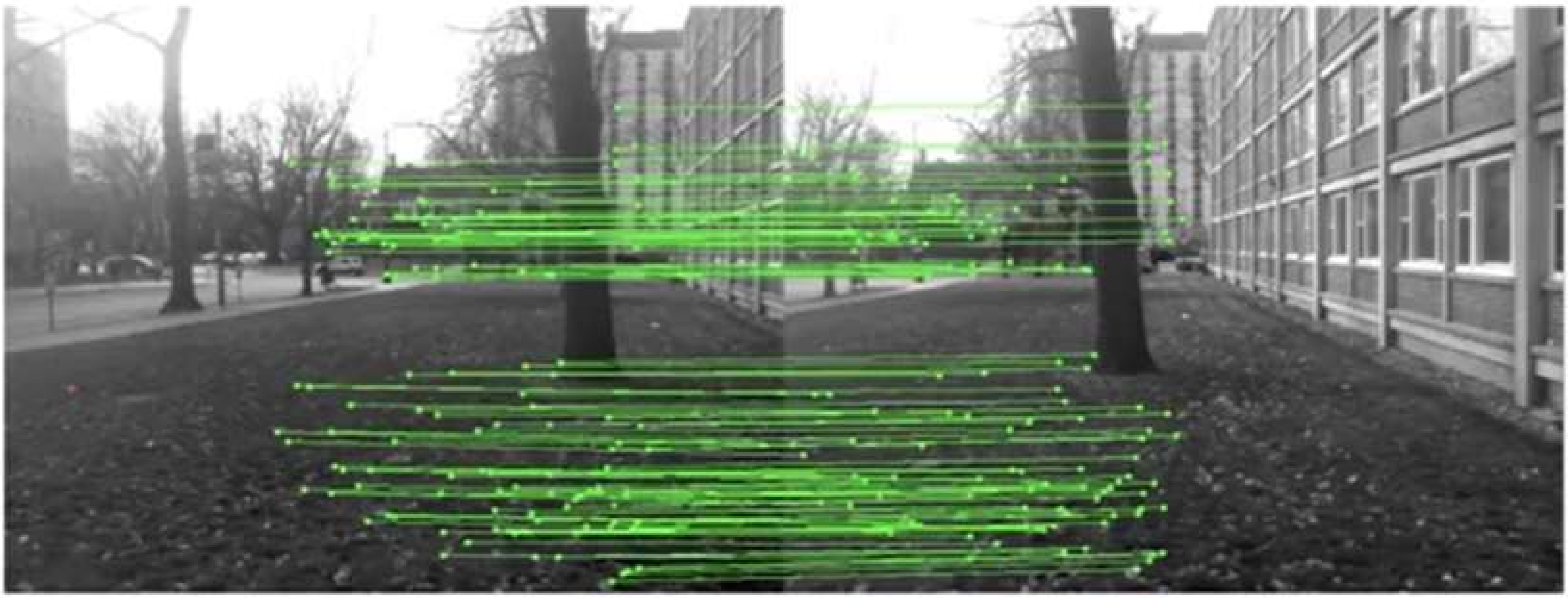
- This is the minimal solution, which is important for the number of RANSAC loops for robust estimation (details later).

- Since points are measured inexactly (“noise”), more than four correspondences are usually used to obtain a least-squares solution (details later).

Possible ways

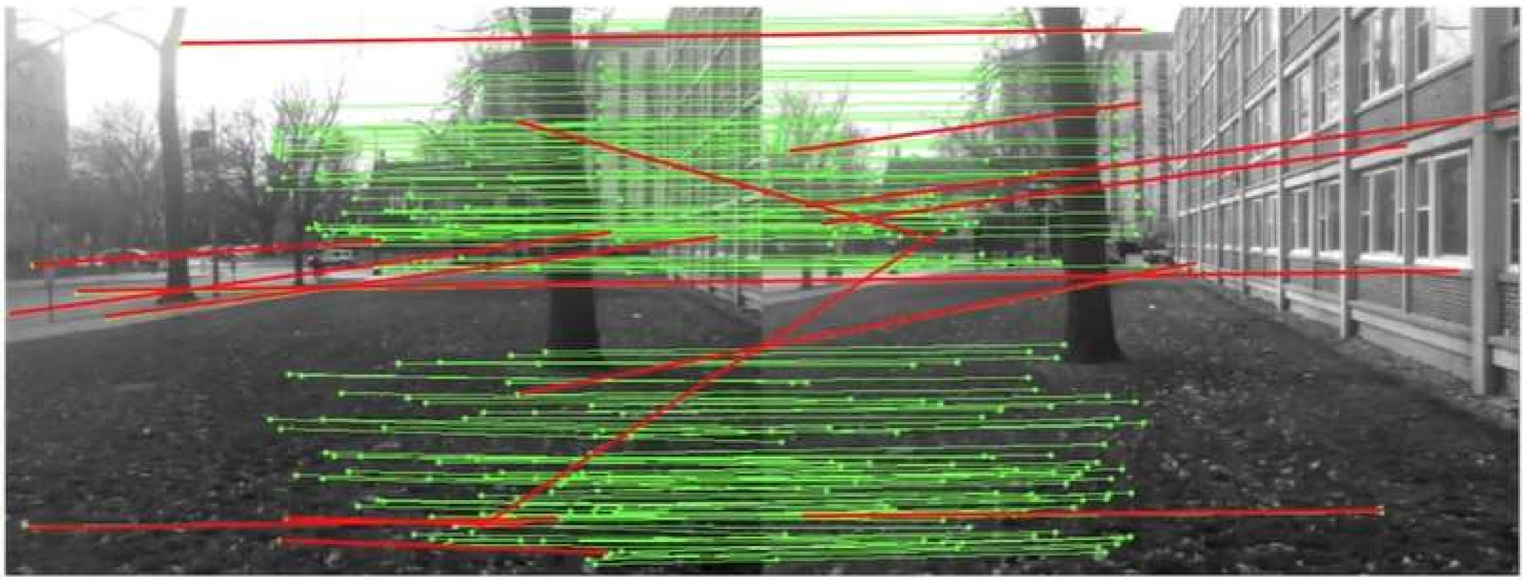
- Direct Linear Transformation (DLT) Algorithm
  - Least square method
  - Iterative method
- RANSAC Algorithm

RANdom SAmple Consensus: RANSAC



RANdom SAmple Consensus: RANSAC

- In reality, keypoint matching gives us many outliers.
- Outliers can severely disturb the least-squares estimation, and should be removed.



The RANSAC Song

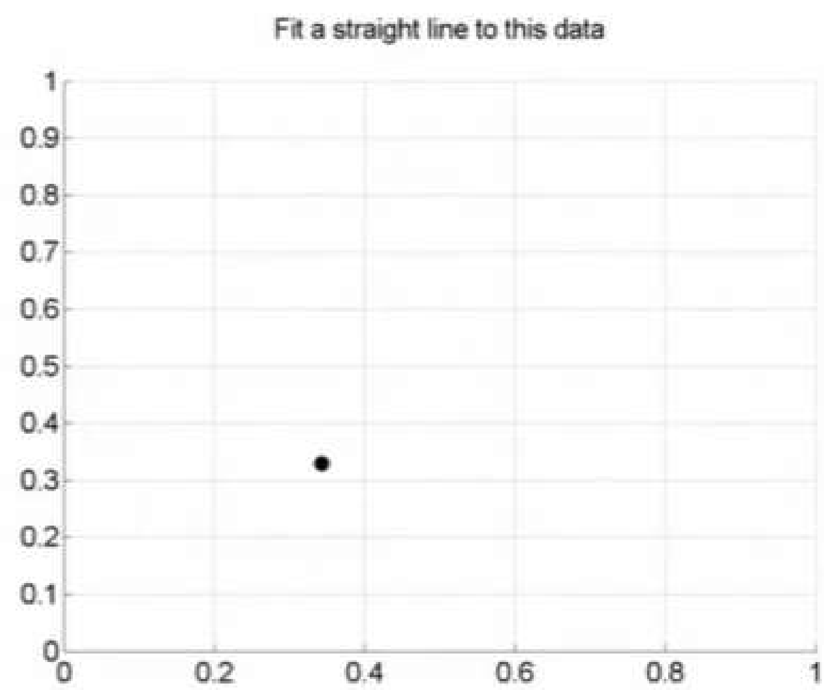
When you have outliers you may face much frustration
if you include them in a model fitting operation.
But if your model's fit to a sample set of minimal size,
the probability of the set being outlier-free will rise.
Brute force tests of all sets will cause computational constipation.

N random samples
will provide an example
of a fitted model uninfluenced by outliers. No need to test all combinations!

Each random trial should have its own unique sample set
and make sure that the sets you choose are not degenerate.
N, the number of sets, to choose is based on the probability
of a point being an outlier, and of finding a set that's outlier free.
Updating *N* as you go will minimise the time spent.

So if you gamble
that *N* samples are ample
to fit a model to your set of points, it's likely that you will win the bet.

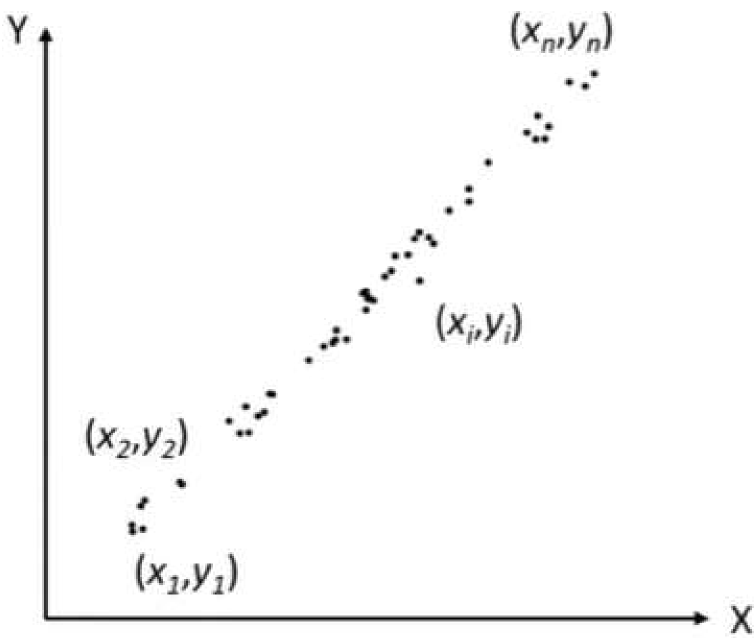
Select the set that boasts
that its number of inliers is the most (you're almost there).
Fit a new model just to those inliers and discard the rest,
an estimated model for your data is now possessed!
This marks the end point of your model fitting quest.



The RANSAC Song

RANSAC: Line fitting example

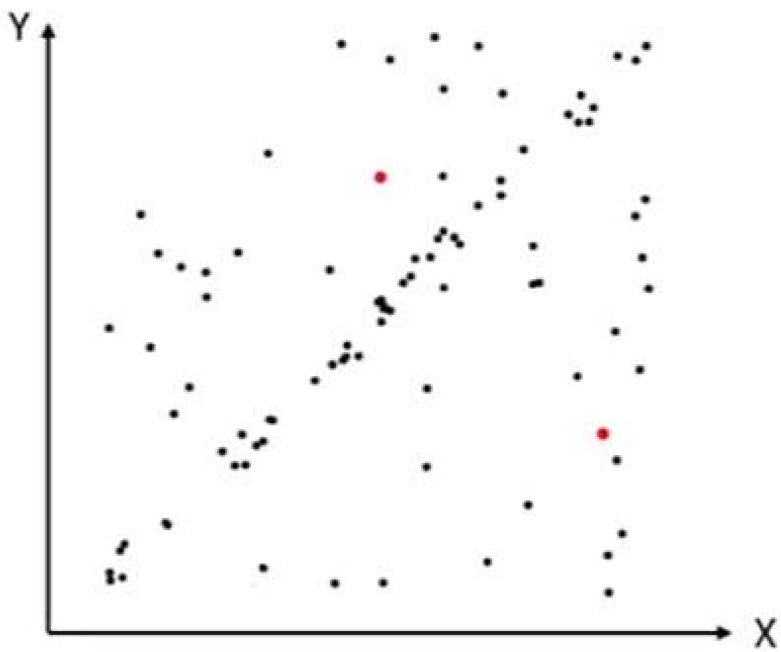
- **Given**: n data points (x_i, y_i), for $i = 1, \dots, n$
- **Find**: Best fit line, i.e. two parameters (m, c) from the line equation $y_i = mx_i + c$, for $i = 1, \dots, n$



RANSAC: Line fitting example

RANSAC Steps:

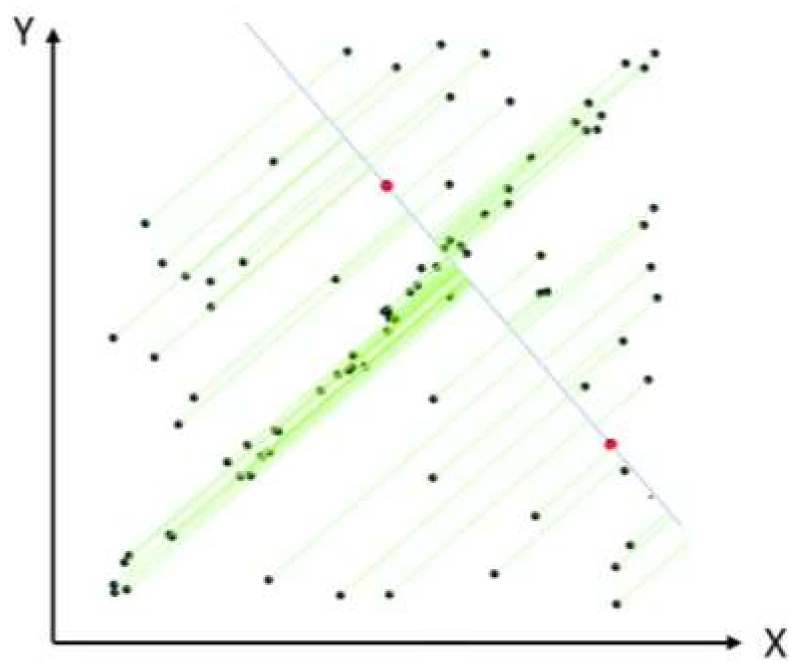
1. Randomly select minimal subset of points, i.e. 2 points



RANSAC: Line fitting example

RANSAC Steps:

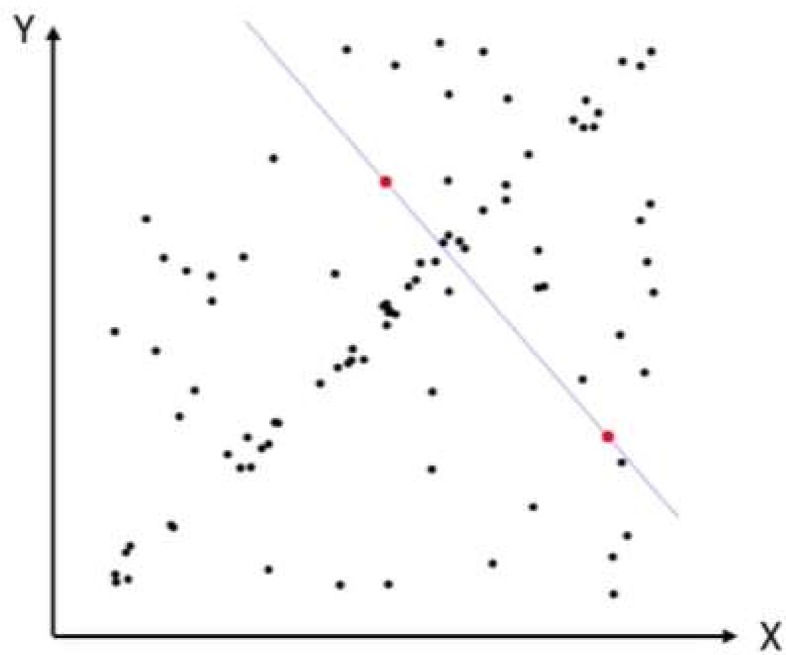
3. Compute error function, i.e. shortest point to line distance



RANSAC: Line fitting example

RANSAC Steps:

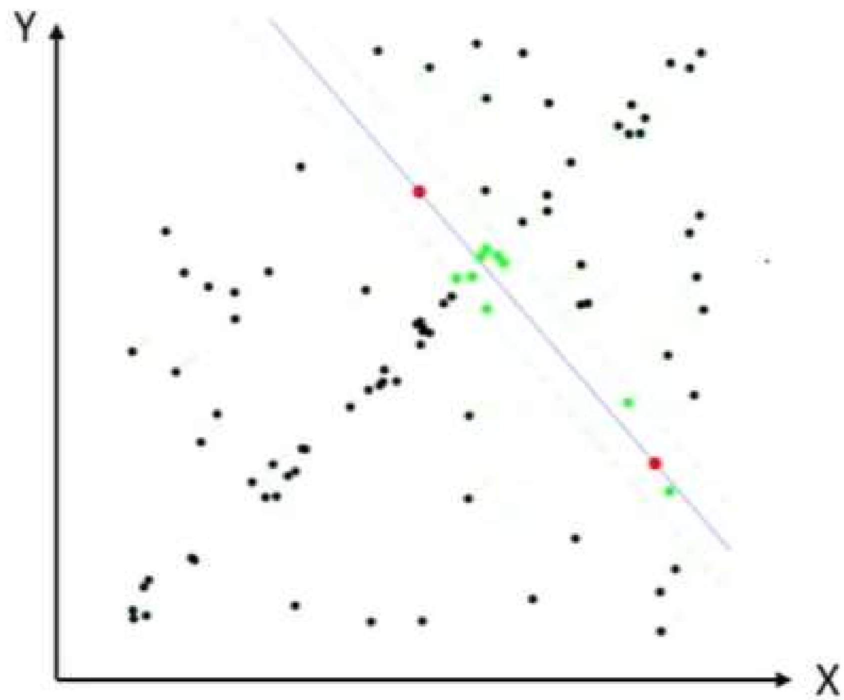
2. Hypothesize a model



RANSAC: Line fitting example

RANSAC Steps:

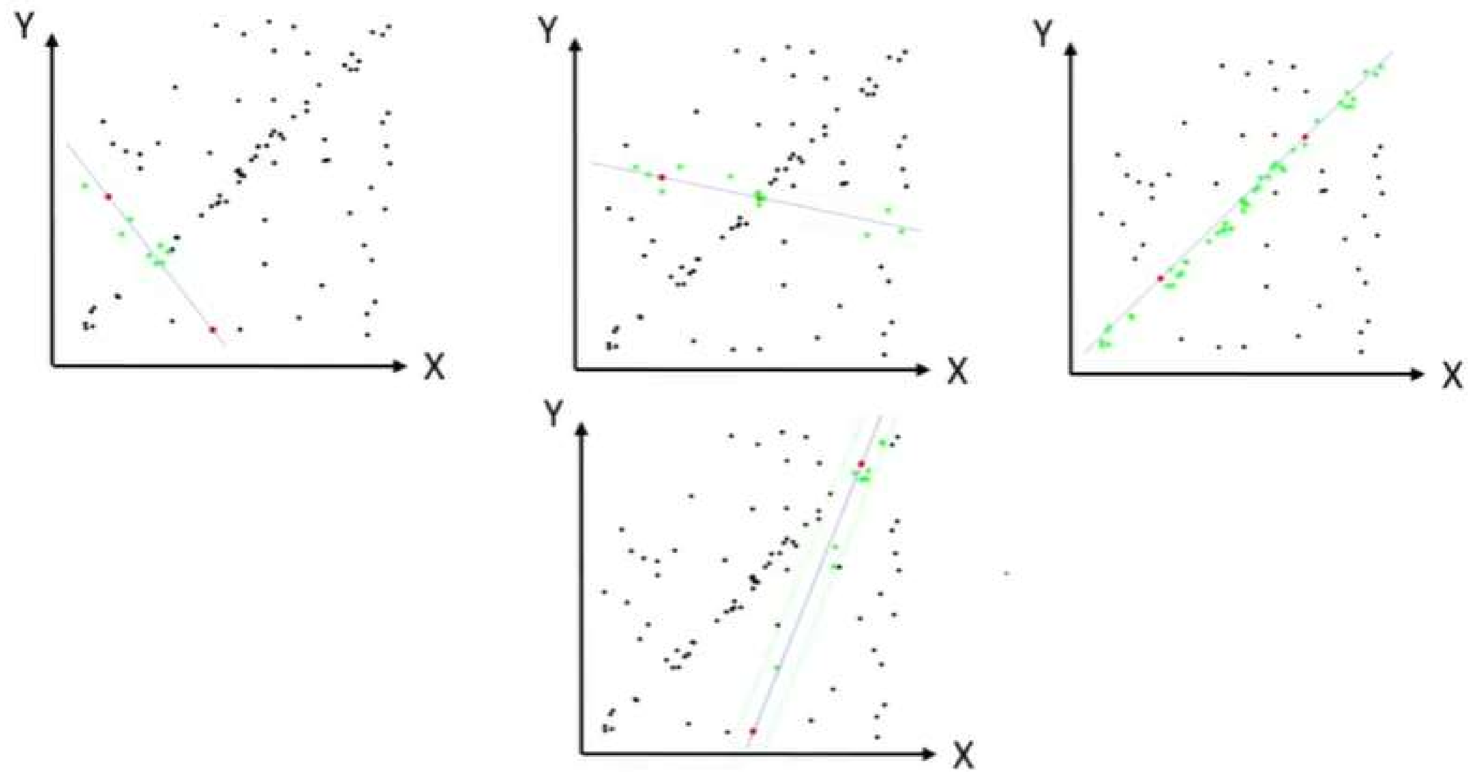
4. Select points consistent with model



RANSAC: Line fitting example

RANSAC Steps:

5. Repeat hypothesize-and-verify loop



RANSAC Algorithm

Objective

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.

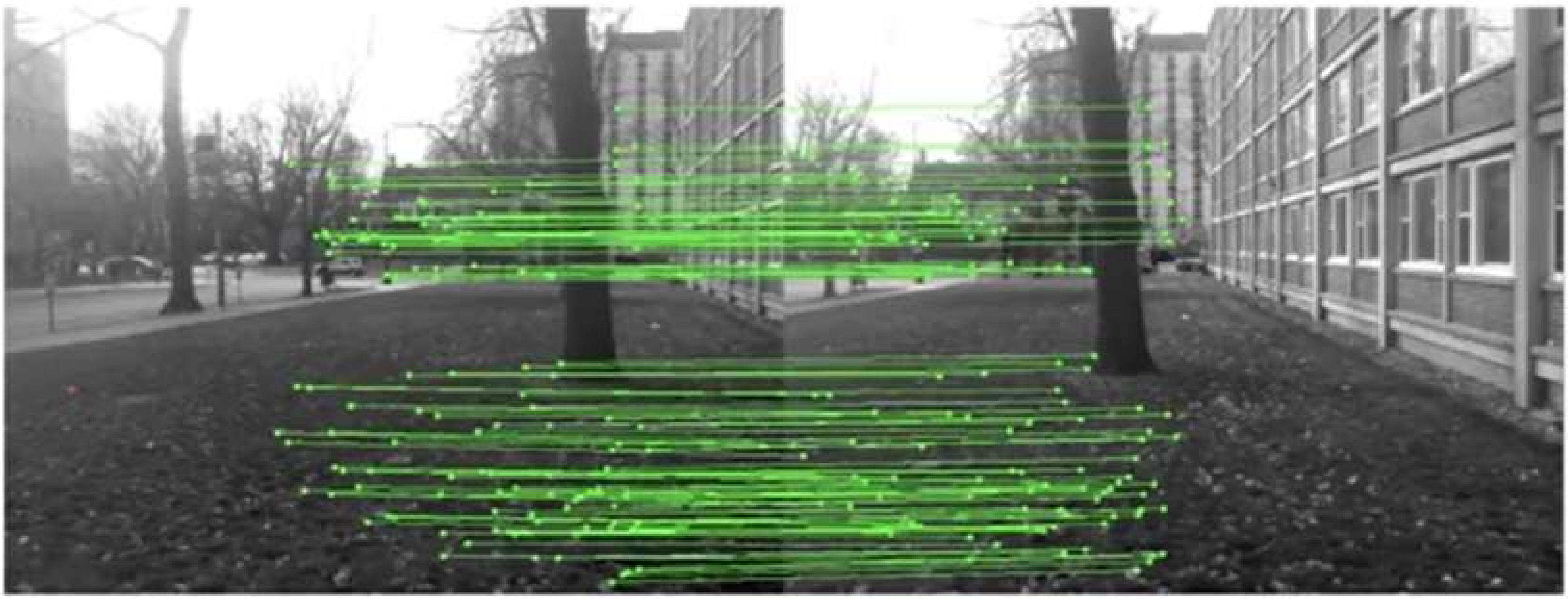
iii. After N trials, select the largest consensus set S_i. The model is re-estimated using all the points in the subset S_i.

Three parameters:

Number of points	s
Distance threshold	t
Number of Samples	N

M. Fischler, R. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", Communications ACM, 1981.

How to find the points of interest?



To continue...