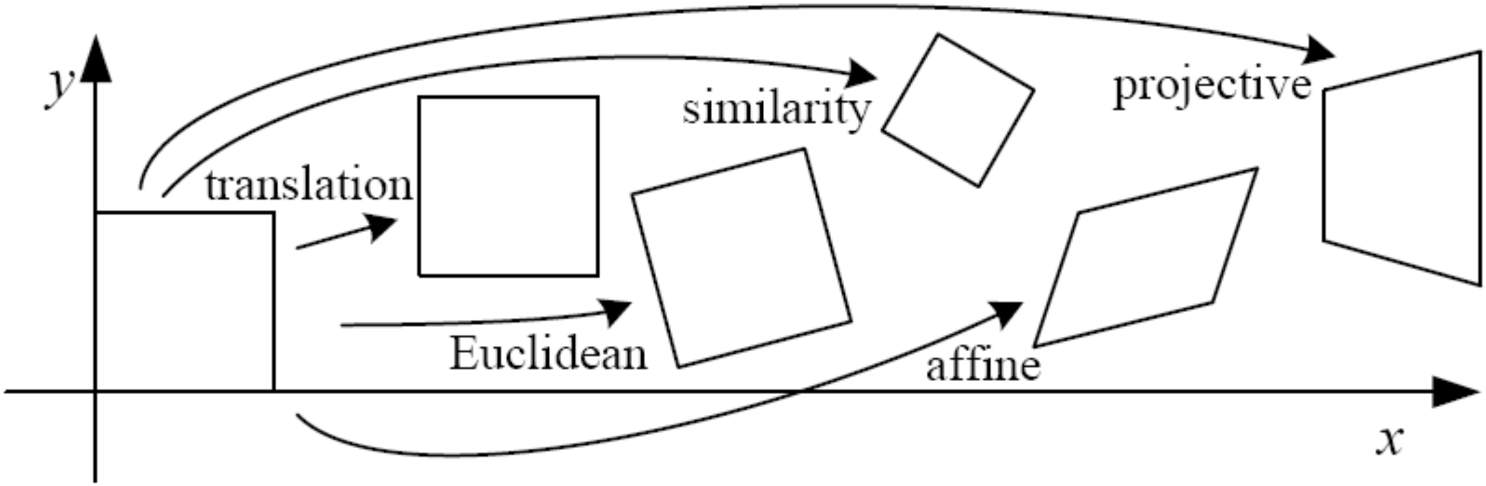
CSCI 497P/597P: Computer Vision



Lecture 16

Fitting Transformations with Outliers:

RANdom **SA**mple **C**onsensus (RANSAC)

Announcements

- P2 is out
- Do you want the option to work in pairs?

Goals

- Understand the Random Sample Consensus (RANSAC) algorithm.

- Be prepared to implement RANSAC to fitting image coordinate transforms using matches that may contain outliers.

Warping

We've found correspondence.

How do we fit a transformation to a given set of matches?

Warping

We've found correspondence.

How do we fit a transformation to a given set of matches?

Analogy: fit a line to a given set of points?

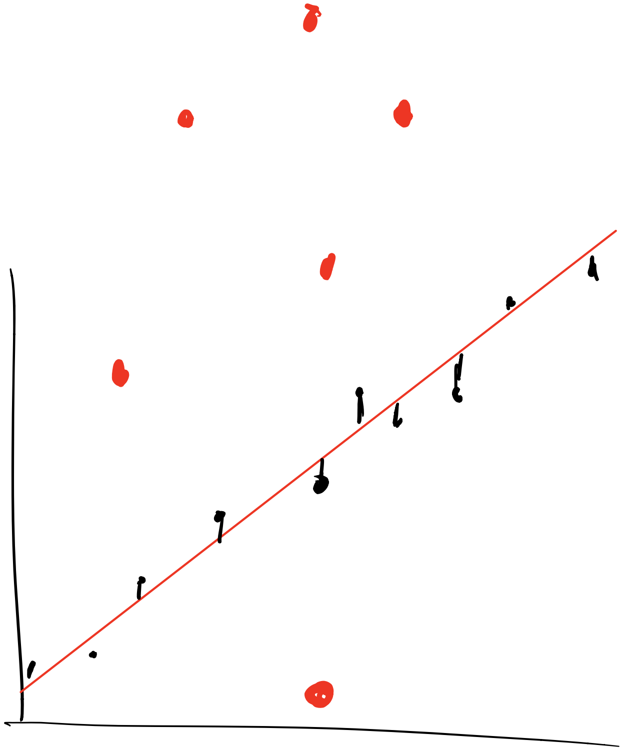
Warping

We've found correspondence.

How do we fit a transformation to a given set of matches?

Analogy: fit a line to a given set of points?

This is a model-fitting problem.



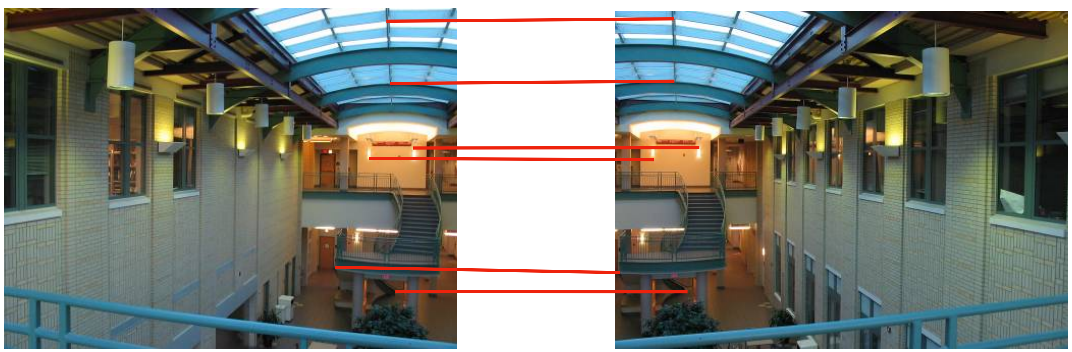
Problem Statement: Last time

(imperfect)*

Given a set of feature matches, how do I **find** the **transformation** that relates the two images?

translation?
affine?
homography?

* all points may be off by a little



Fitting a Homography: TL;DM

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1 x_1 & -x'_1 y_1 & -x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1 x_1 & -y'_1 y_1 & -y'_1 \\ & & & & & \vdots & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & -x'_n x_n & -x'_n y_n & -x'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -y'_n x_n & -y'_n y_n & -y'_n \end{bmatrix} \begin{bmatrix} h_{00} \\ h_{01} \\ h_{02} \\ h_{10} \\ h_{11} \\ h_{12} \\ h_{20} \\ h_{21} \\ h_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$\underbrace{\mathbf{A}}_{2n \times 9} \; \underbrace{\mathbf{h}}_{9} = \underbrace{\mathbf{0}}_{2n}$$

Fitting a Homography: TL;DM

im 1 im 2

- For each feature match (x_i, y_i) --> (x_i', y_i'), fill in 2 rows of A as in:

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1x_1 & -x'_1y_1 & -x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1x_1 & -y'_1y_1 & -y'_1 \\ & & & & \vdots & & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & -x'_nx_n & -x'_ny_n & -x'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -y'_nx_n & -y'_ny_n & -y'_n \end{bmatrix}$$

$\sigma_1 \; \sigma_2 \cdots \sigma_9$ V^T: $v_1, v_2, v_3, \ldots, v_9$

- Solve the homogeneous least squares problem: $\min_h \|Ah\|^2$

  $S \to 9\times1$

  Σ

  - Take the SVD of A to get U, S, and V
  - ~~Let h = the right singular vector of A whose singular value is smallest.~~
  - Let h = the column of V (row of V^T) whose column (row) index is the same as that of the smallest diagonal entry of S.

- Reshape h into H_{3x3} and divide by the bottom-right entry.

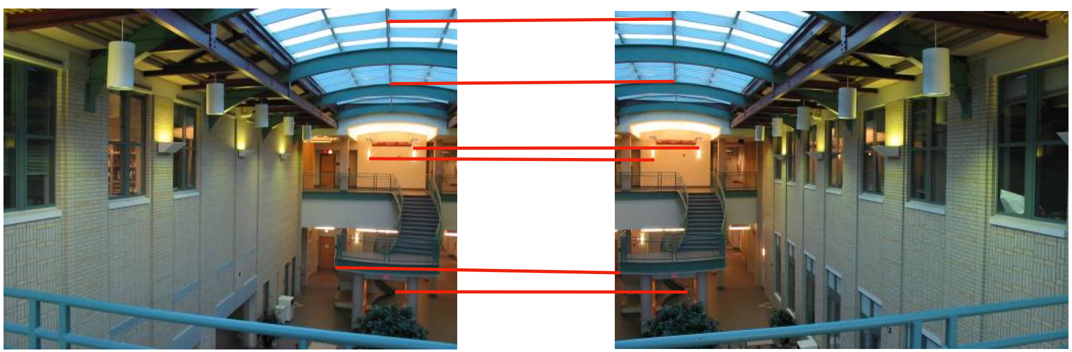
Problem Statement: Last time

(imperfect*)

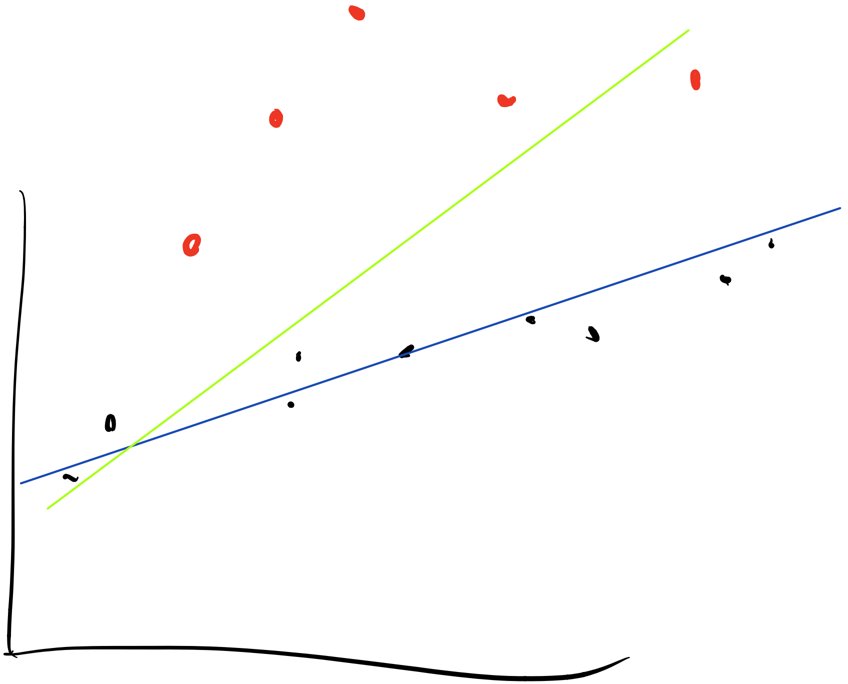
Given a set of feature matches, how do I **find** the **transformation** that relates the two images?

translation?
affine?
homography?

* all points off by a bit



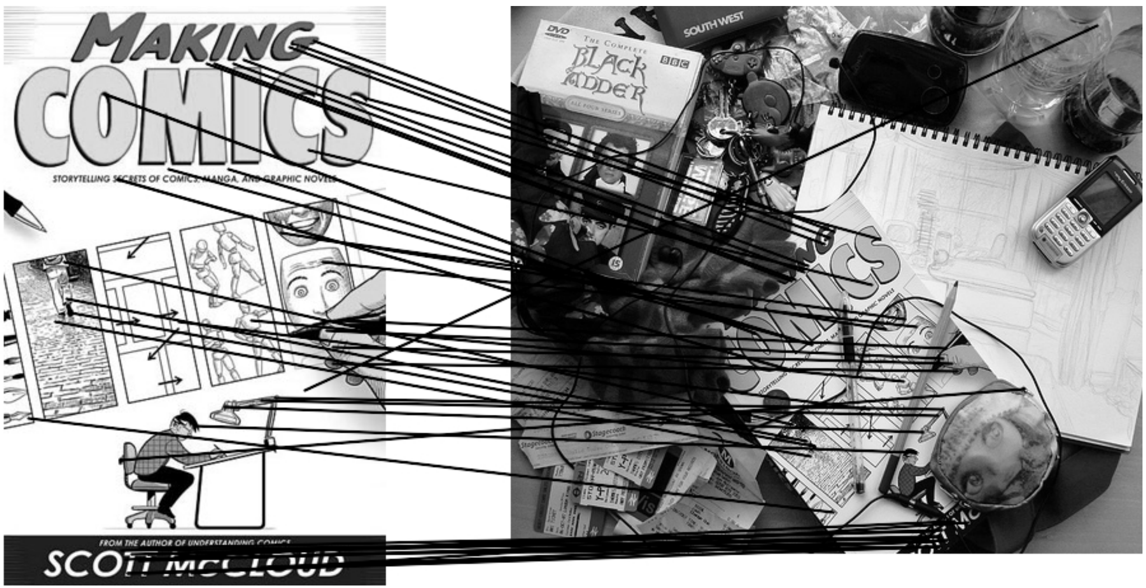
When does least squares work well?



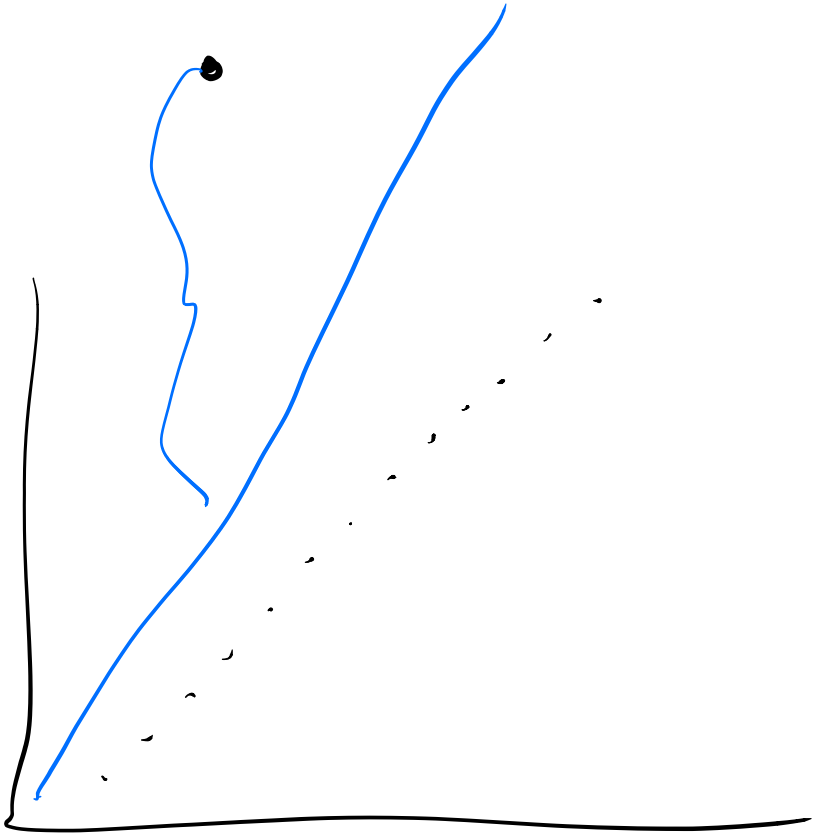
Problem Statement: Today

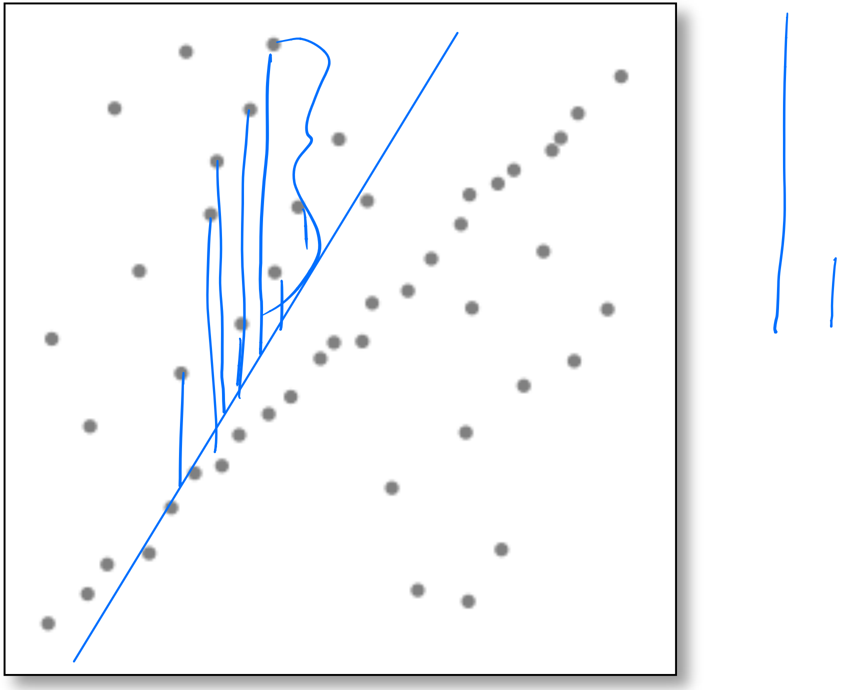
(imperfect*)

Given a set of feature matches, how do I **find** the **transformation** that relates the two images?

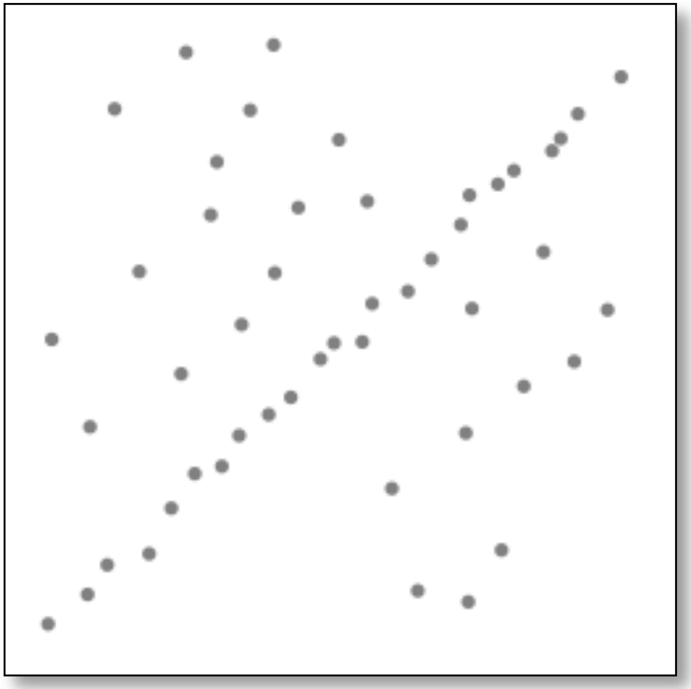


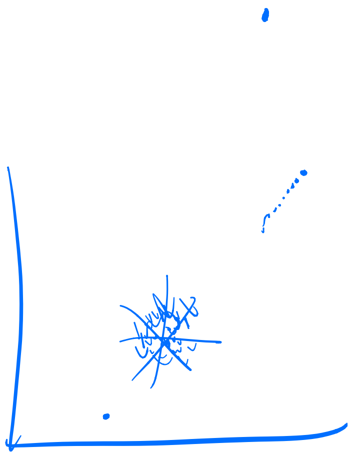
* a few points are outliers





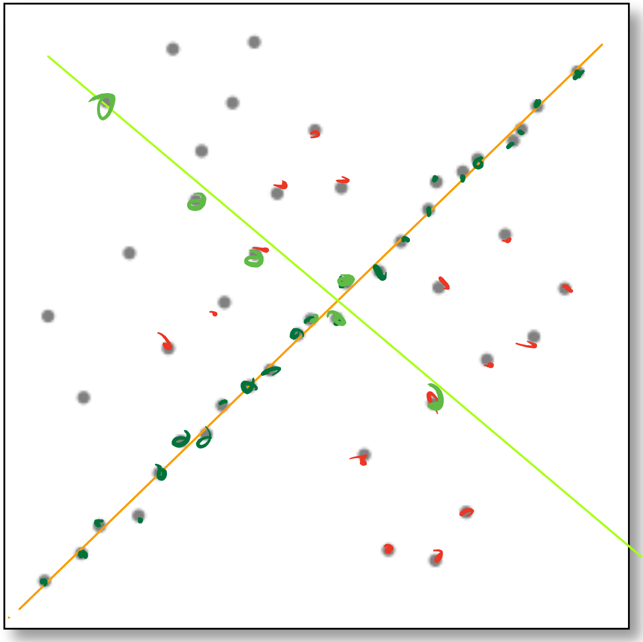
How could we fit a line to this data?





An idea

- If I have a hypothesis, I can tell how "good" it is:
  - Count the number of points that are close to the line (**inliers**)



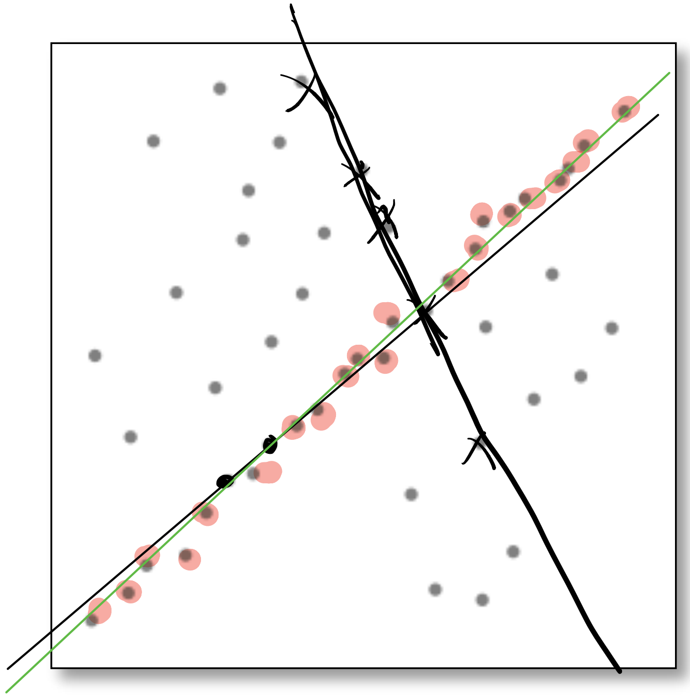
An idea

- If I have a hypothesis, I can tell how "good" it is:
  - Count the number of points that are close to the line (**inliers**)
- Algorithm: generate all possible lines and pick the one with the most inliers
  - Runtime: $O(\infty)$

Another idea

- If I have a hypothesis, I can tell how "good" it is:
  - Count the number of points that are close to the line (**inliers**)
- Algorithm: generate **many random** lines and pick the one with the most inliers.
- Questions:
  - How many lines? Which ones? How do I measure "inlierness"?

$s = 2$



The key

- Points that fit the model will agree.
- Points that don't fit the model will all be wrong in their own unique ways, and there won't be a very large set of them that agree with each other.

“All good matches are alike; every bad match is bad in its own way.”
-Tolstoy, as misquoted by Alyosha Efros

RANSAC

The Algorithm

for $i = 0..K$:

$d_i \leftarrow s$ random data points

$M_i \leftarrow \text{fit\_model}(d_i)$

observation

$\text{inlier\_count} \leftarrow \Sigma\left(\mathbb{1}\left(\overbrace{M(x_i)}^{\text{model prediction}} - y_i < \delta\right)\right)$

1 if, 0 otherwise

if inlier_count > best_count:

best_count ← inlier_count

best_M ← M_i

best_data ← $\{x_i, y_i : |M(x_i) - y_i| < \delta\}$

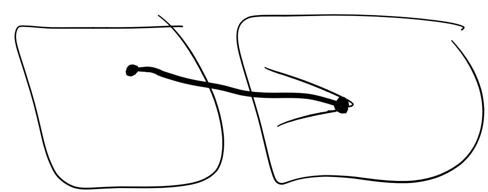
M_final ← fit_model(best_data)

Choose Parameters:

δ - inlier threshold

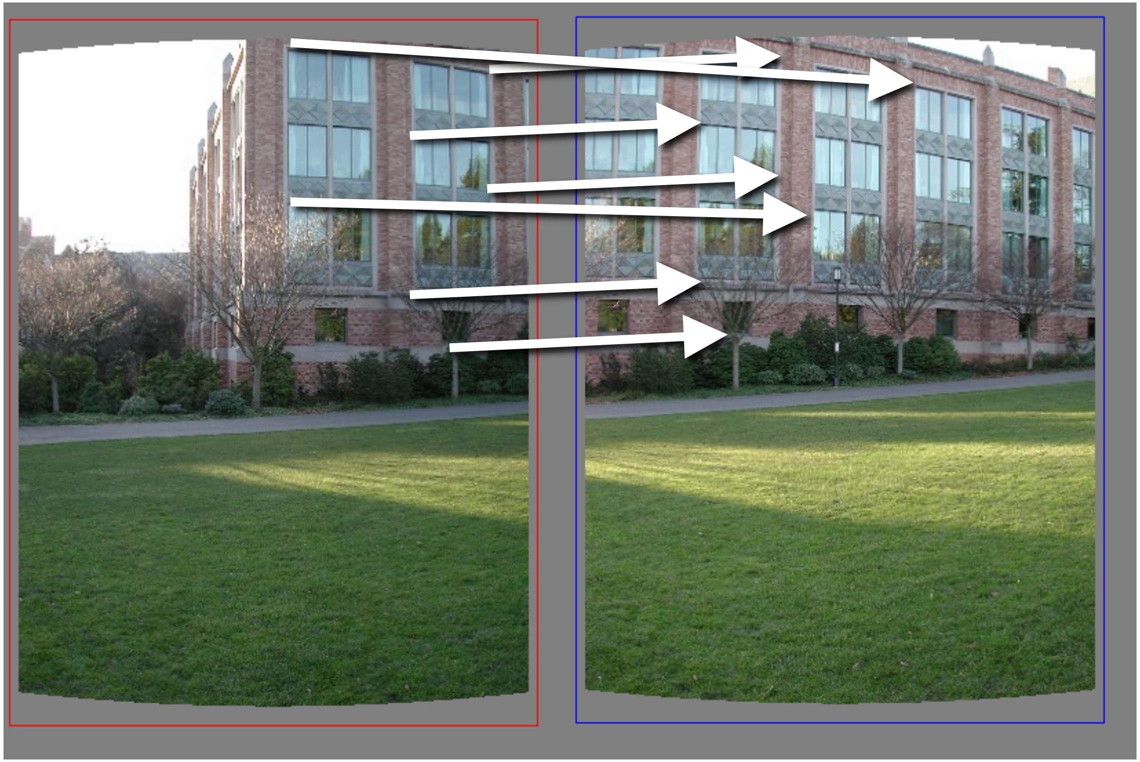
assume Gaussian noise $\delta \leftarrow \sigma, 2\sigma$

K: iterations - guess, or assume fraction of inliers and acceptable probability of picking a set of all inliers.



Questions Remain

- How do we generate a hypothesis for transformations?

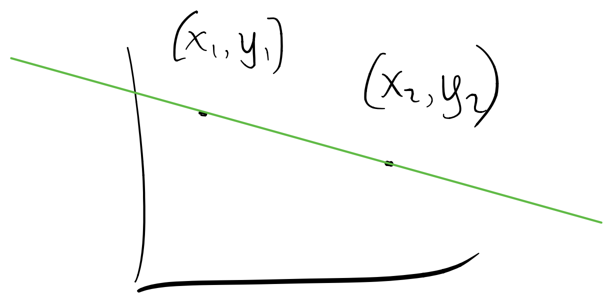


Questions Remain

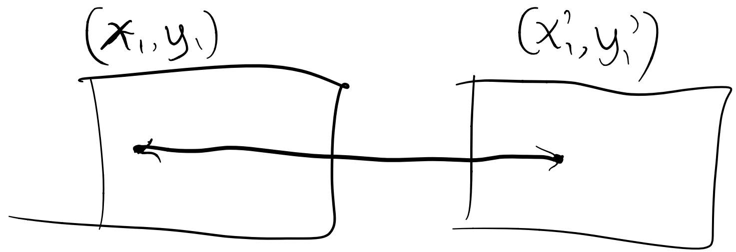
- How do we generate a hypothesis for transformations?

S: smallest number of points that can fully determine your model. Or: # of degrees of freedom.

Linear Regression? 2 points fit a line.

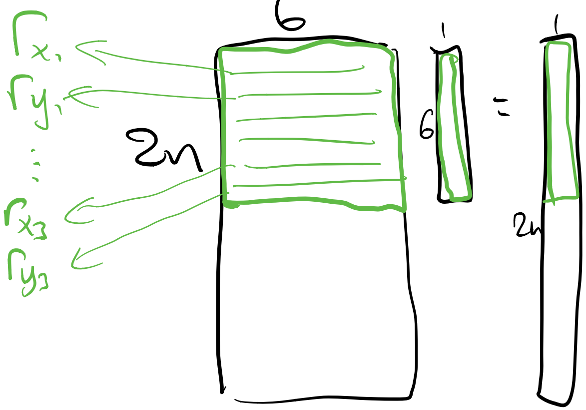


Translation? 1 match determines (t_x, t_y)



Affine?

$$A t = b$$

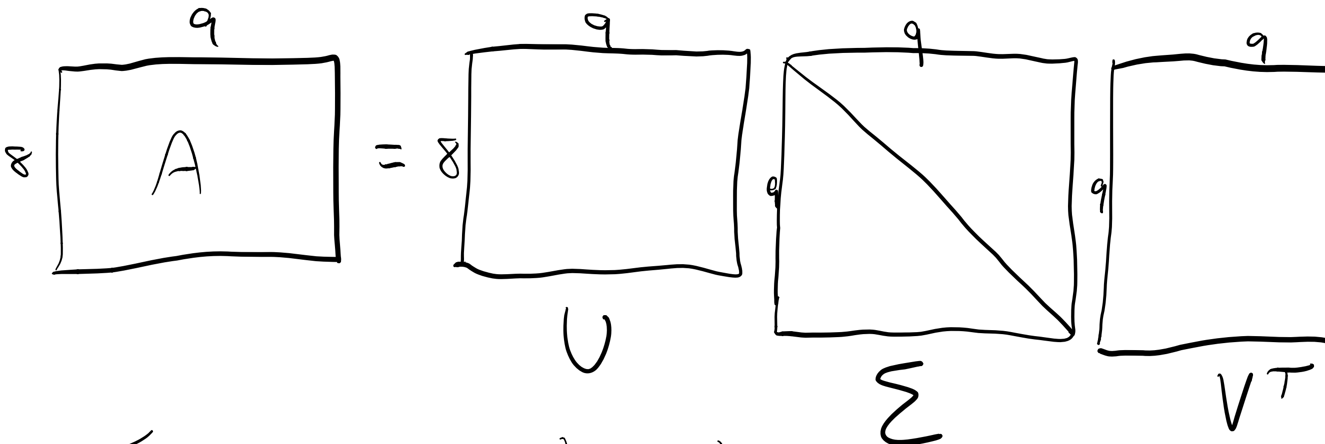


smallest n without being underdetermined: 3

Homography? Awkward: A is $2n \times \underline{9}$

but has only 8 DOF

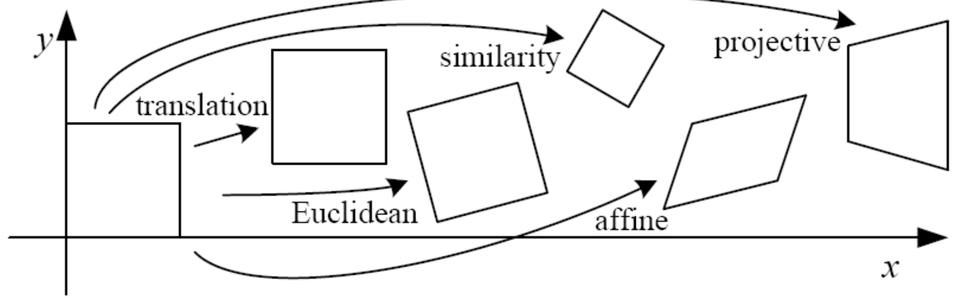
Turns out: SVD works on 8×9 A:



Same argument applies, so

A_{89} uses 4 matches, 2 residuals per match.

TL;DM: $s = \lceil \mathrm{DOF}/2 \rceil$



Name	Matrix	# D.O.F.	Preserves:	Icon
translation	$\begin{bmatrix} \boldsymbol{I} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	2	orientation $+\cdots$	
rigid (Euclidean)	$\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	3	lengths $+\cdots$	
similarity	$\begin{bmatrix} s\boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}_{2\times 3}$	4	angles $+\cdots$	
affine	$\begin{bmatrix} \boldsymbol{A} \end{bmatrix}_{2\times 3}$	6	parallelism $+\cdots$	
projective	$\begin{bmatrix} \tilde{\boldsymbol{H}} \end{bmatrix}_{3\times 3}$	8	straight lines	

Questions do not! Remain

- How do we choose the parameters?
    - k (# iterations)
    - s (# data points needed to fit a model)
    - δ (inlier threshold)

(See above - page 22 for δ, k
25-28 for s)