Image Alignment

Samuel Cheng

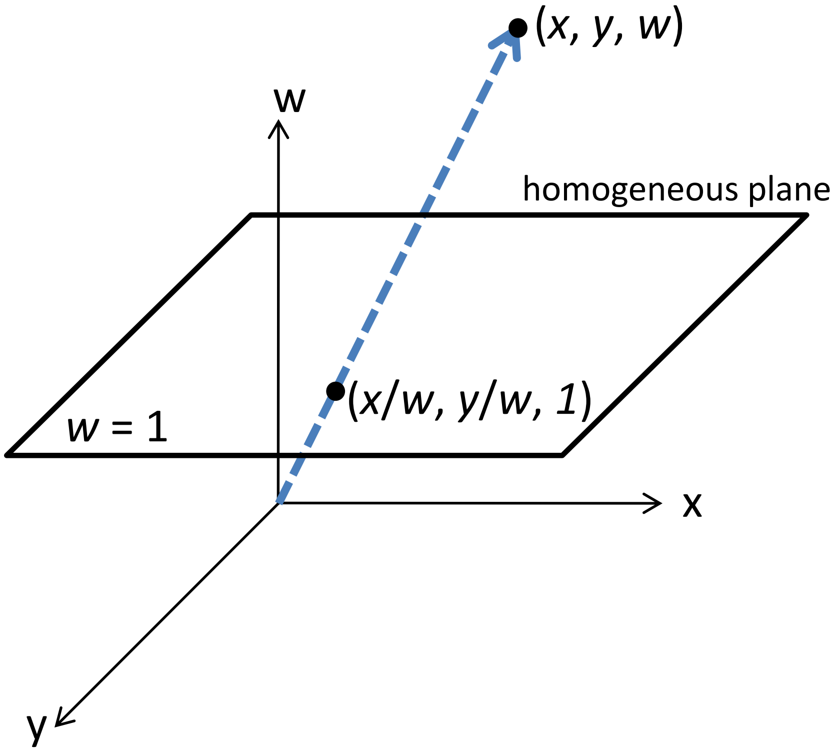
Slide credit: Noah Snavely, James Thompkin

Homogeneous coordinates

Trick: add one more coordinate:

$$(x, y) \Rightarrow \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

homogeneous image coordinates



Converting *from* homogeneous coordinates

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \Rightarrow (x/w, y/w)$$

Basic affine transformations

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Translate

$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{s}_x & 0 & 0 \\ 0 & \boldsymbol{s}_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

Scale

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

2D *in-plane* rotation

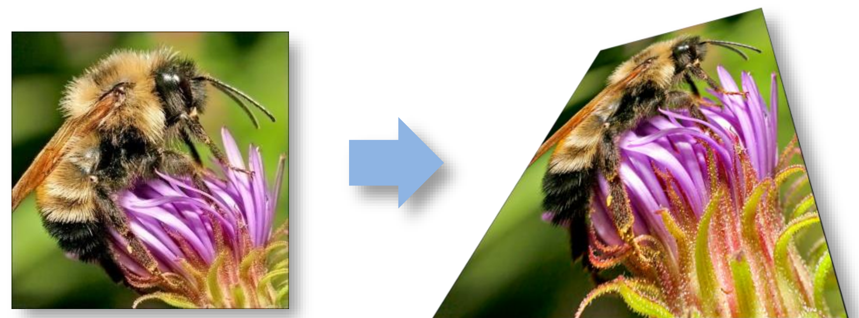
$$\begin{bmatrix} \boldsymbol{x}' \\ \boldsymbol{y}' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{sh}_x & 0 \\ \boldsymbol{sh}_y & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{y} \\ 1 \end{bmatrix}$$

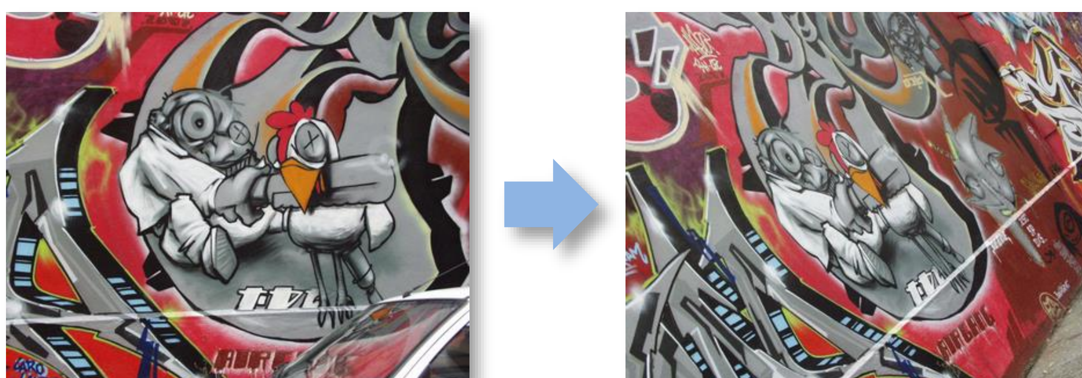
Shear

Projective Transformations aka Homographies aka Planar Perspective Maps

$$\mathbf{H} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & 1 \end{bmatrix}$$

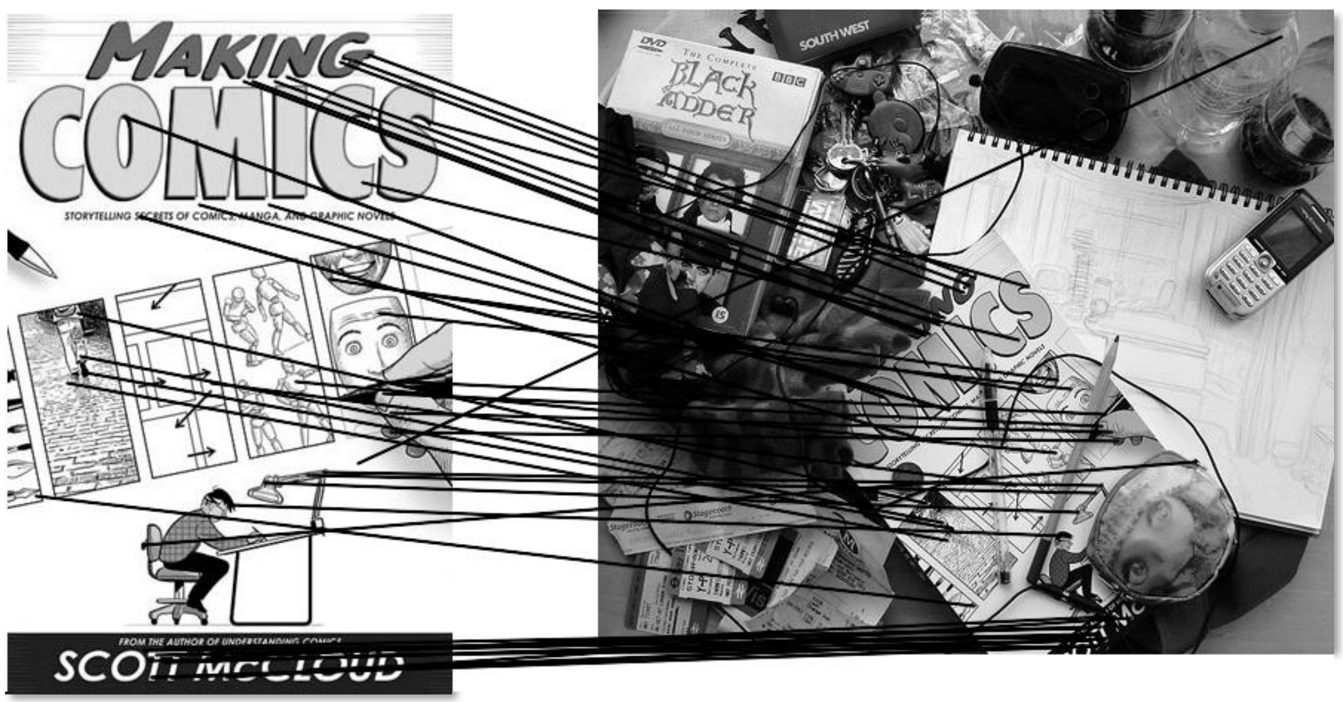
Called a *homography*
(or *planar perspective map*)





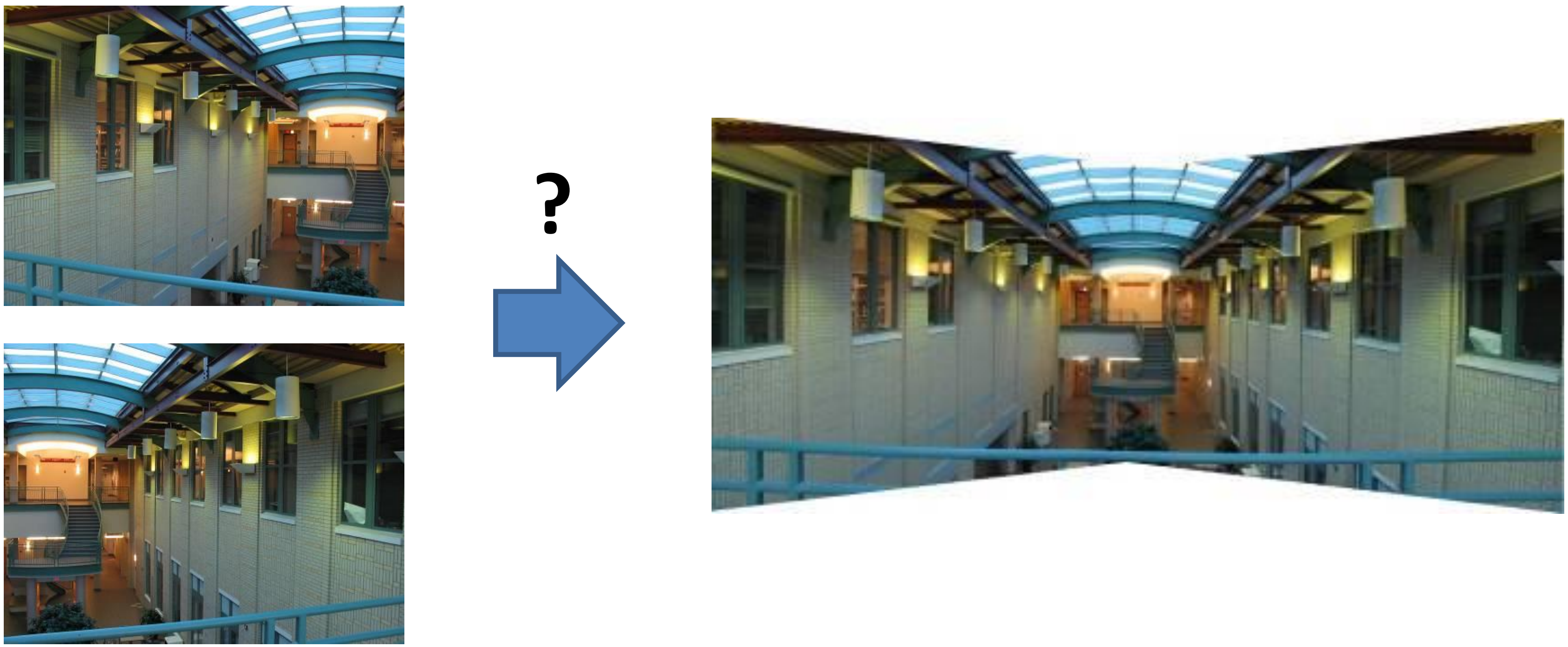
Computing transformations

- Given a set of matches between images A and B
  - How can we compute the transform T from A to B?

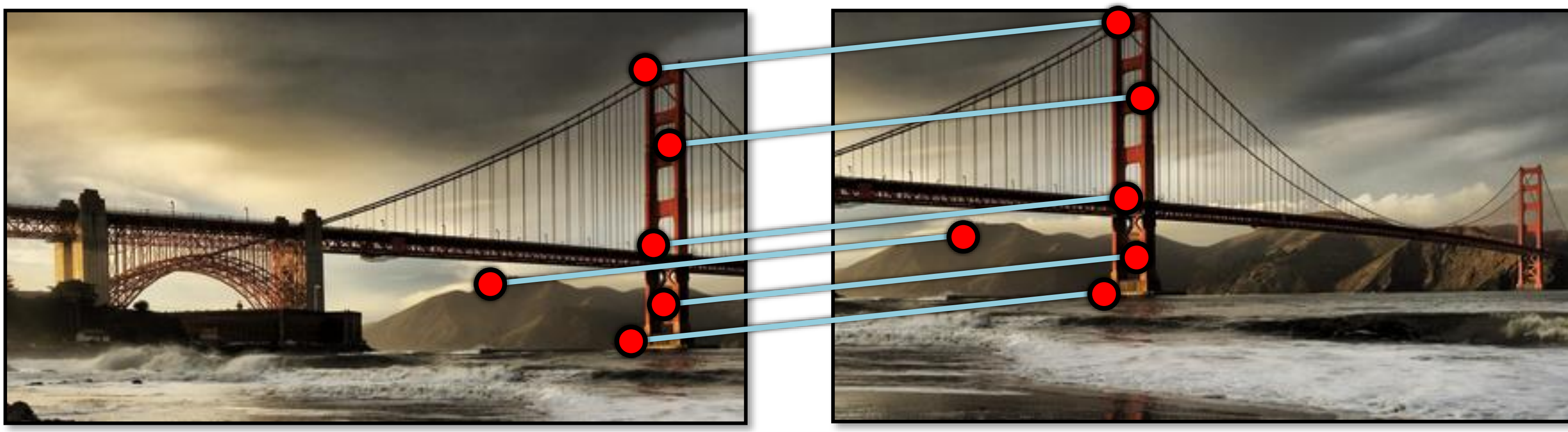


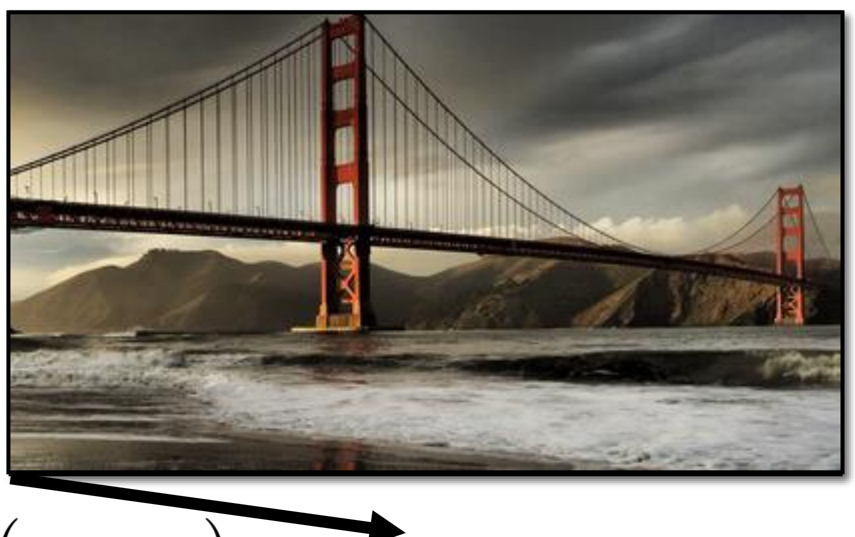
  - Find transform T that best “agrees” with the matches

Computing transformations



Simple case: translations



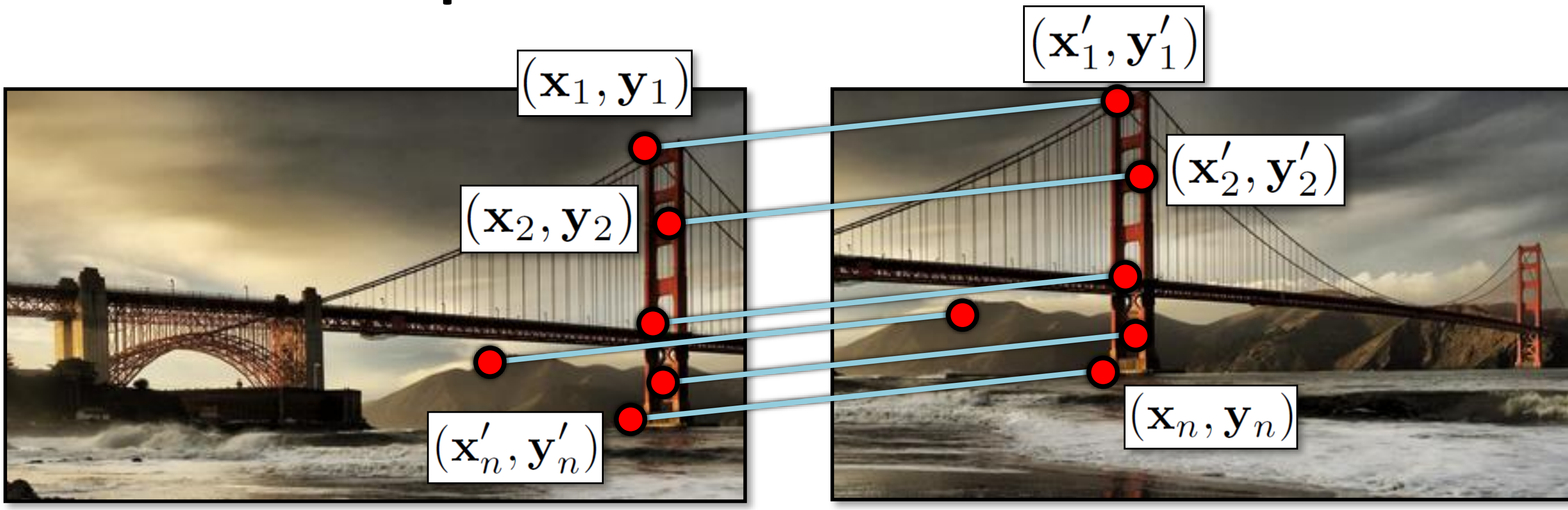


$(\mathbf{x}_t, \mathbf{y}_t)$

$$T = \begin{bmatrix} 1 & 0 & \mathbf{x}_t \\ 0 & 1 & \mathbf{y}_t \\ 0 & 0 & 1 \end{bmatrix}$$

How do we solve for $(\mathbf{x}_t, \mathbf{y}_t)$?

Simple case: translations

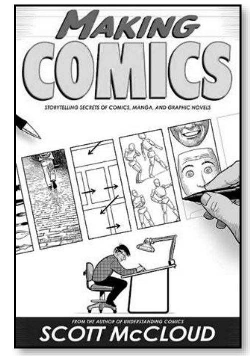


Displacement of match *i* = $(\mathbf{x}'_i - \mathbf{x}_i, \mathbf{y}'_i - \mathbf{y}_i)$

$$(\mathbf{x}_t, \mathbf{y}_t) = \left(\frac{1}{n} \sum_{i=1}^{n} \mathbf{x}'_i - \mathbf{x}_i, \frac{1}{n} \sum_{i=1}^{n} \mathbf{y}'_i - \mathbf{y}_i \right)$$

Affine transformations

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$



- How many unknowns?
- How many equations per match?
- How many matches do we need?

Affine transformations

- Matrix form

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_2 & y_2 & 1 \\ & & & \vdots & & \\ x_n & y_n & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ x'_2 \\ y'_2 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix}$$

$$\underset{2n \times 6}{\mathbf{A}} \quad \underset{6 \times 1}{\mathbf{t}} = \underset{2n \times 1}{\mathbf{b}}$$

Least squares

- Find **t** that minimizes $||\mathbf{At} - \mathbf{b}||^2$

$$f(\mathbf{t}) = \|\mathbf{At} - \mathbf{b}\|^2 = (\mathbf{At} - \mathbf{b})^T (\mathbf{At} - \mathbf{b})$$

$$= \mathbf{t}^T \mathbf{A}^T \mathbf{A} \mathbf{t} - \mathbf{t}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{A} \mathbf{t} + \mathbf{b}^T \mathbf{b}$$

$$(\nabla_{\mathbf{t}}(\mathbf{t}^T \mathbf{C} \mathbf{t}))_i = \frac{\partial \mathbf{t}^T \mathbf{C} \mathbf{t}}{\partial t_i} = \frac{\partial \sum_j \sum_k t_j C_{j,k} t_k}{\partial t_i}$$

$$\delta_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

$$= \sum_j \sum_k \delta_{i,j} C_{j,k} t_k + t_j C_{j,k} \delta_{i,k} = \sum_k C_{i,k} t_k + \sum_j t_j C_{j,i}$$

$$= \sum_k C_{i,k} t_k + \sum_j C^T_{i,j} t_j = \sum_k (C_{i,k} + C^T_{i,k}) t_k = ((\mathbf{C} + \mathbf{C}^T)\mathbf{t})_i$$

$$\therefore \ \nabla_{\mathbf{t}}(\mathbf{t}^T \mathbf{C} \mathbf{t}) = (\mathbf{C} + \mathbf{C}^T)\mathbf{t}$$

Least squares

- Find **t** that minimizes $||\mathbf{At} - \mathbf{b}||^2$

$$f(\mathbf{t}) = \|\mathbf{At} - \mathbf{b}\|^2 = (\mathbf{At} - \mathbf{b})^T (\mathbf{At} - \mathbf{b})$$

$$= \mathbf{t}^T \mathbf{A}^T \mathbf{At} - \mathbf{t}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{At} + \mathbf{b}^T \mathbf{b}$$

$$\nabla_{\mathbf{t}} (\mathbf{t}^T \mathbf{Ct}) = (\mathbf{C} + \mathbf{C}^T) \mathbf{t}$$

$$\text{Similarly, } (\nabla_{\mathbf{t}} (\mathbf{c}^T \mathbf{t}))_i = \frac{\partial \mathbf{c}^T \mathbf{t}}{\partial t_i} = \frac{\partial \sum_j c_j t_j}{\partial t_i} = \sum_j c_j \delta_{i,j} = c_i$$

$$\therefore \quad \nabla_{\mathbf{t}} (\mathbf{c}^T \mathbf{t}) = \mathbf{c}$$

Least squares

- Find **t** that minimizes $||\mathbf{At} - \mathbf{b}||^2$

$$f(\mathbf{t}) = \|\mathbf{At} - \mathbf{b}\|^2 = (\mathbf{At} - \mathbf{b})^T (\mathbf{At} - \mathbf{b})$$

$$= \mathbf{t}^T \mathbf{A}^T \mathbf{At} - \mathbf{t}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{At} + \mathbf{b}^T \mathbf{b}$$

$$\nabla_{\mathbf{t}} (\mathbf{t}^T \mathbf{Ct}) = (\mathbf{C} + \mathbf{C}^T)\mathbf{t}$$

$$\nabla_{\mathbf{t}} (\mathbf{c}^T \mathbf{t}) = \mathbf{c}$$

$$\nabla_{\mathbf{t}} f(\mathbf{t}) = \nabla_{\mathbf{t}} \left(\mathbf{t}^T \mathbf{A}^T \mathbf{At} - \mathbf{t}^T \mathbf{A}^T \mathbf{b} - \mathbf{b}^T \mathbf{At} + \mathbf{b}^T \mathbf{b} \right)$$

$$= \nabla_{\mathbf{t}} \left(\mathbf{t}^T \mathbf{A}^T \mathbf{At} - (\mathbf{A}^T \mathbf{b})^T \mathbf{t} - \mathbf{b}^T \mathbf{At} + \mathbf{b}^T \mathbf{b} \right)$$

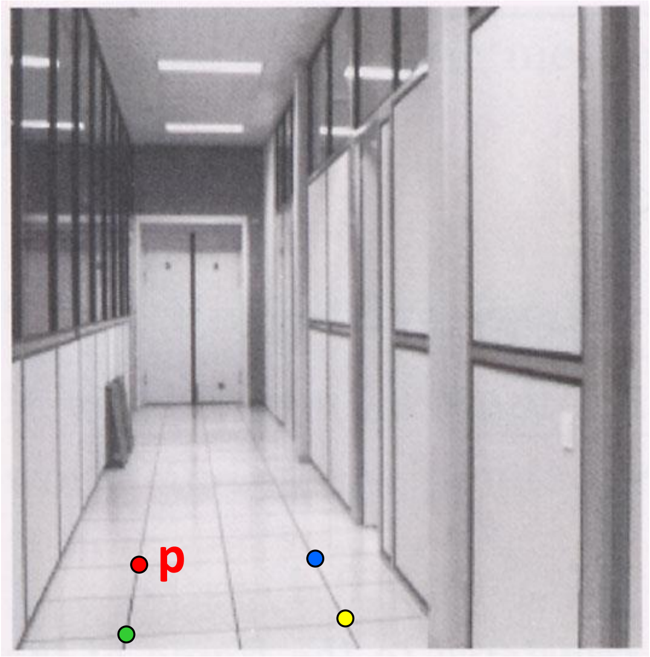
$$= (\mathbf{A}^T \mathbf{A} + (\mathbf{A}^T \mathbf{A})^T)\mathbf{t} - \mathbf{A}^T \mathbf{b} - (\mathbf{b}^T \mathbf{A})^T)$$

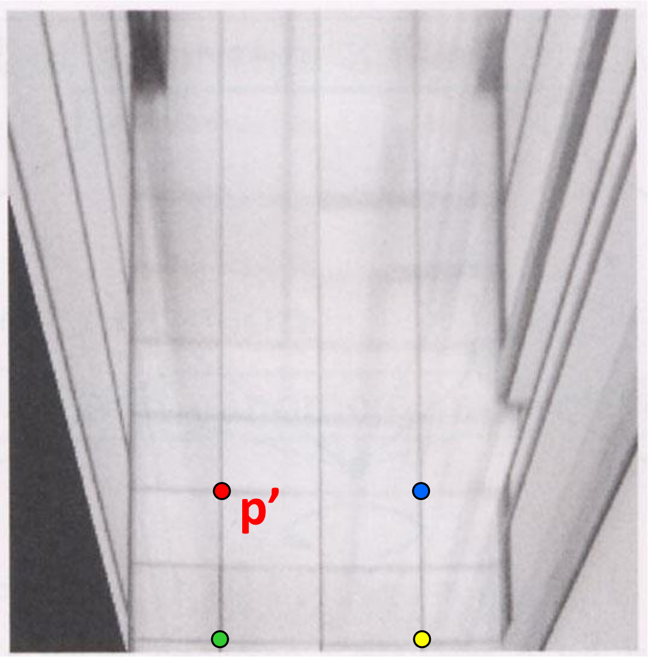
$$= 2((\mathbf{A}^T \mathbf{A})\mathbf{t} - \mathbf{A}^T \mathbf{b})$$

$$\nabla_{\mathbf{t}} f(\mathbf{t}) = \mathbf{0} \Rightarrow (\mathbf{A}^T \mathbf{A})\mathbf{t} - \mathbf{A}^T \mathbf{b} = \mathbf{0}$$

$$\therefore \mathbf{t} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$$

Homographies





To unwarp (rectify) an image

- solve for homography **H** given **p** and **p’**
- solve equations of the form: w**p’** = **Hp**
  - linear in unknowns: w and coefficients of **H**
  - H is defined up to an arbitrary scale factor
  - how many points are necessary to solve for **H**?

Solving for homographies

$$\begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} \cong \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

$$x'_i = \frac{h_{00}x_i + h_{01}y_i + h_{02}}{h_{20}x_i + h_{21}y_i + h_{22}}$$

$$y'_i = \frac{h_{10}x_i + h_{11}y_i + h_{12}}{h_{20}x_i + h_{21}y_i + h_{22}}$$

Not linear!

$$x'_i(h_{20}x_i + h_{21}y_i + h_{22}) = h_{00}x_i + h_{01}y_i + h_{02}$$
$$y'_i(h_{20}x_i + h_{21}y_i + h_{22}) = h_{10}x_i + h_{11}y_i + h_{12}$$

Solving for homographies

$$x'_i(h_{20}x_i + h_{21}y_i + h_{22}) = h_{00}x_i + h_{01}y_i + h_{02}$$
$$y'_i(h_{20}x_i + h_{21}y_i + h_{22}) = h_{10}x_i + h_{11}y_i + h_{12}$$

$$\begin{bmatrix} x_i & y_i & 1 & 0 & 0 & 0 & -x'_i x_i & -x'_i y_i & -x'_i \\ 0 & 0 & 0 & x_i & y_i & 1 & -y'_i x_i & -y'_i y_i & -y'_i \end{bmatrix} \begin{bmatrix} h_{00} \\ h_{01} \\ h_{02} \\ h_{10} \\ h_{11} \\ h_{12} \\ h_{20} \\ h_{21} \\ h_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Solving for homographies

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

$$\underset{2n \times 9}{\mathbf{A}} \qquad \underset{9}{\mathbf{h}} \qquad \underset{2n}{\mathbf{0}}$$

Defines a least squares problem: minimize $\|\mathbf{A}\mathbf{h} - \mathbf{0}\|^2$

$\mathbf{h} = \mathbf{0}$ is a trivial solution. Consider instead

$\min \|\mathbf{A}\mathbf{h}\|^2$ subject to $\|\mathbf{h}\| = 1$

Solving for homographies

$$\min_{\mathbf{h}} \|\mathbf{Ah}\|^2 \quad \text{subject to } \|\mathbf{h}\| = 1 \quad \equiv \quad \min_{\mathbf{h}} \mathbf{h}^T(\mathbf{A}^T\mathbf{A})\mathbf{h} \quad \text{subject to } \|\mathbf{h}\| = 1$$

Let $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$ be normalized eigenvectors of $\mathbf{A}^T\mathbf{A}$ with corresponding eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$

Since $\mathbf{A}^T\mathbf{A}$ is real symmetric, $\mathbf{x}_1, \cdots, \mathbf{x}_N$ form a complete orthonormal basis. Thus, we can write $\mathbf{h} = \sum_i a_i \mathbf{x}_i$,

and $\|\mathbf{h}\| = \mathbf{h}^T\mathbf{h} = \left(\sum_i a_i \mathbf{x}_i\right)^T \left(\sum_j a_j \mathbf{x}_j\right) = \sum_{i,j} a_i a_j \delta_{i,j} = \sum_i a_i^2$

$$\mathbf{h}^T(\mathbf{A}^T\mathbf{A})\mathbf{h} = \left(\sum_i a_i \mathbf{x}_i\right)^T \left(\sum_j a_j \lambda_j \mathbf{x}_j\right) = \sum_{i,j} a_i a_j \lambda_j \delta_{i,j} = \sum_i \lambda_i a_i^2$$

$$\min_{\mathbf{h}} \mathbf{h}^T(\mathbf{A}^T\mathbf{A})\mathbf{h} \quad \text{subject to } \|\mathbf{h}\| = 1 \quad \Rightarrow \mathbf{h}^* = \mathbf{x}_1$$

Questions?

Image Alignment Algorithm

Given images A and B

1. Compute image features for A and B
2. Match features between A and B
3. Compute homography between A and B using least squares on set of matches

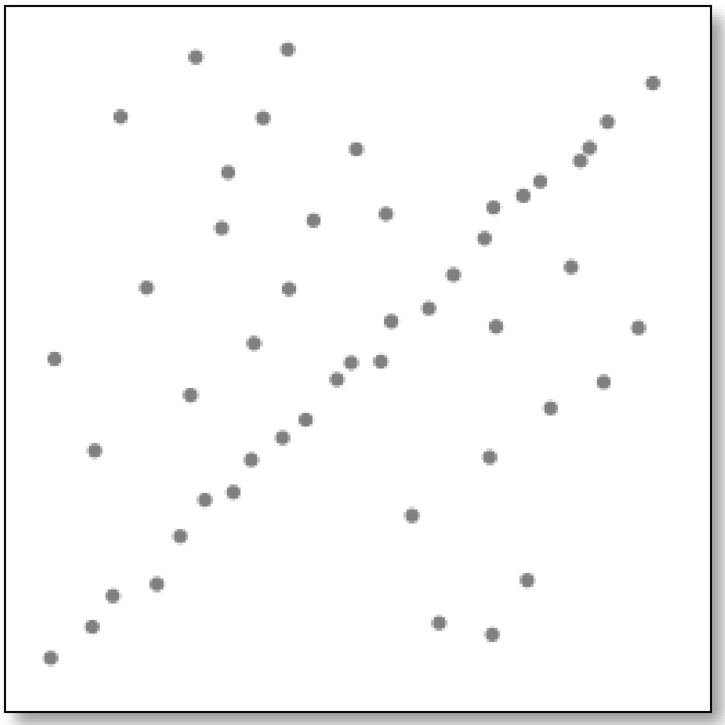
What could go wrong?

Outliers

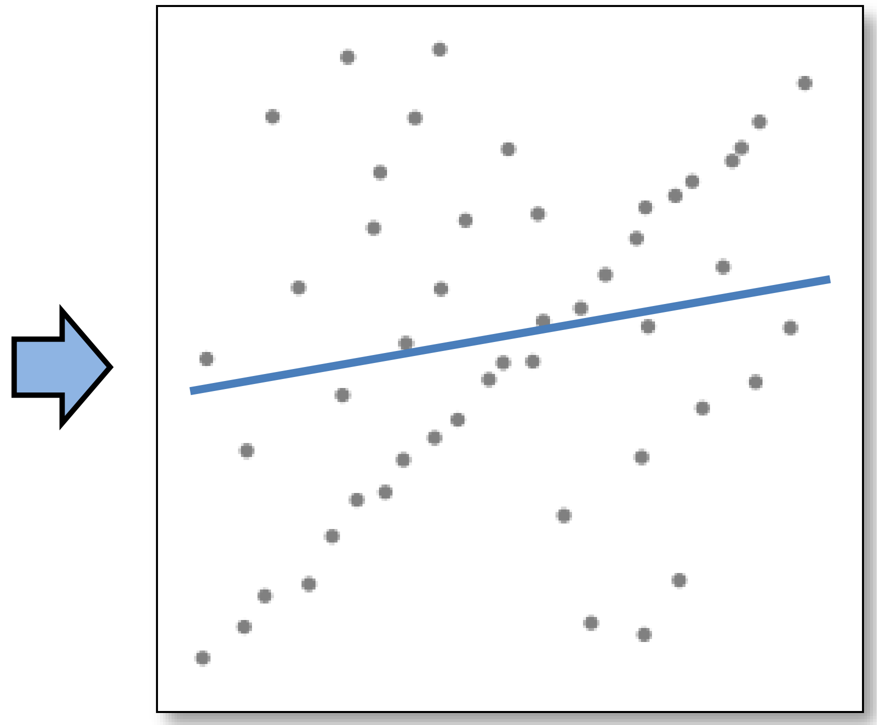

outliers
MAKING COMICS
STORYTELLING SECRETS OF COMICS, MANGA, AND GRAPHIC NOVELS
SCOTT McCLOUD
SOUTH WEST
THE COMPLETE BLACK ADDER
inliers

Robustness

- Let’s consider a simpler example… linear regression



Problem: Fit a line to these datapoints



Least squares fit

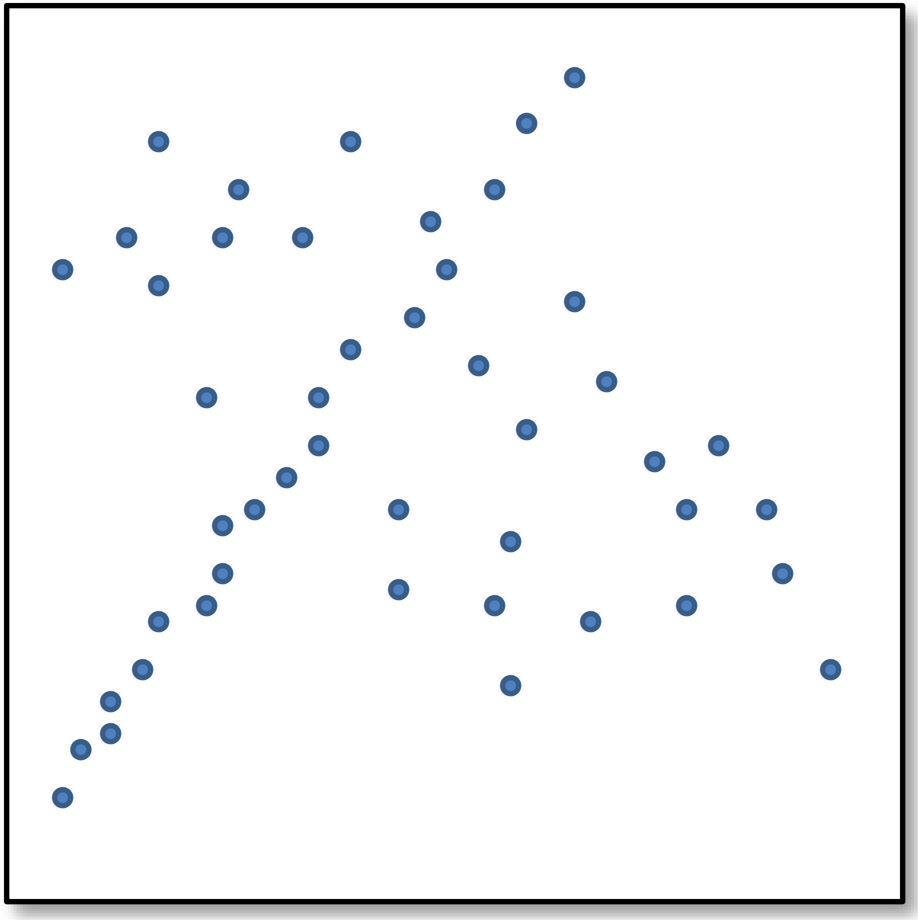
- How can we fix this?

Idea

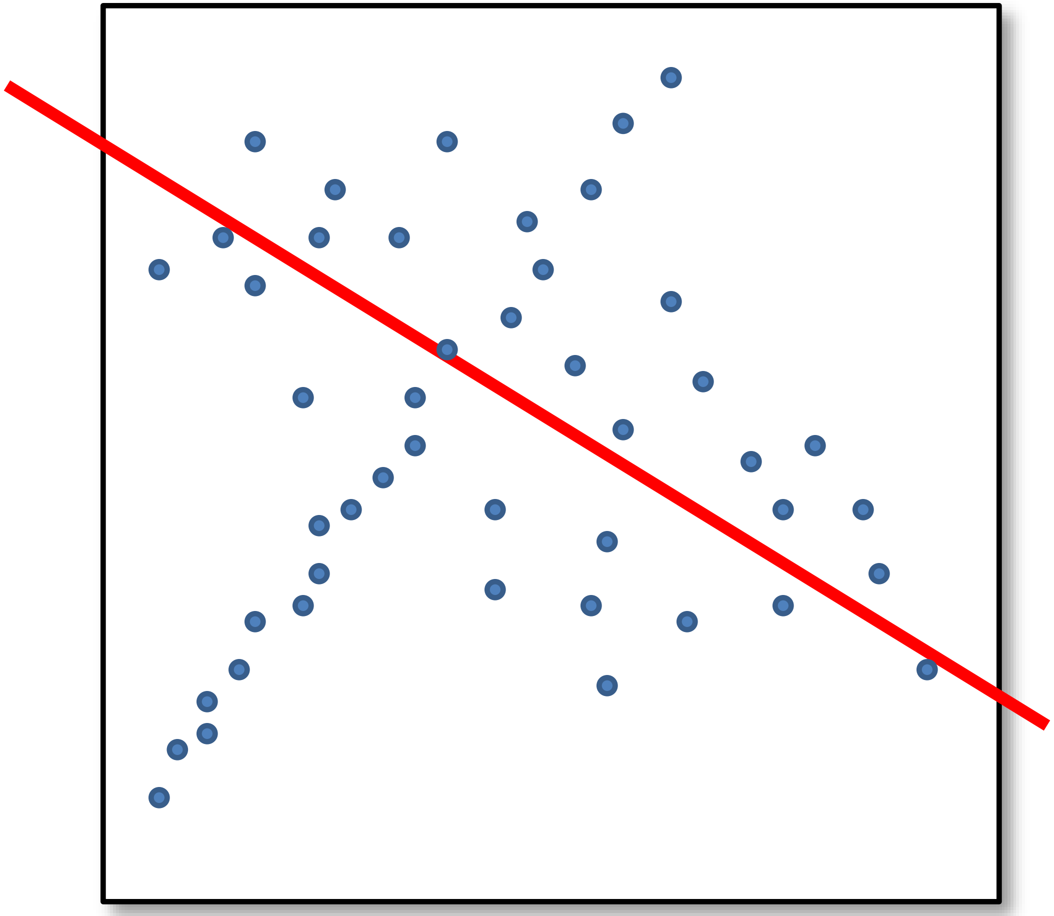
- Given a hypothesized line
- Count the number of points that “agree” with the line
  - “Agree” = within a small distance of the line
  - I.e., the **inliers** to that line

- For all possible lines, select the one with the largest number of inliers

Counting inliers

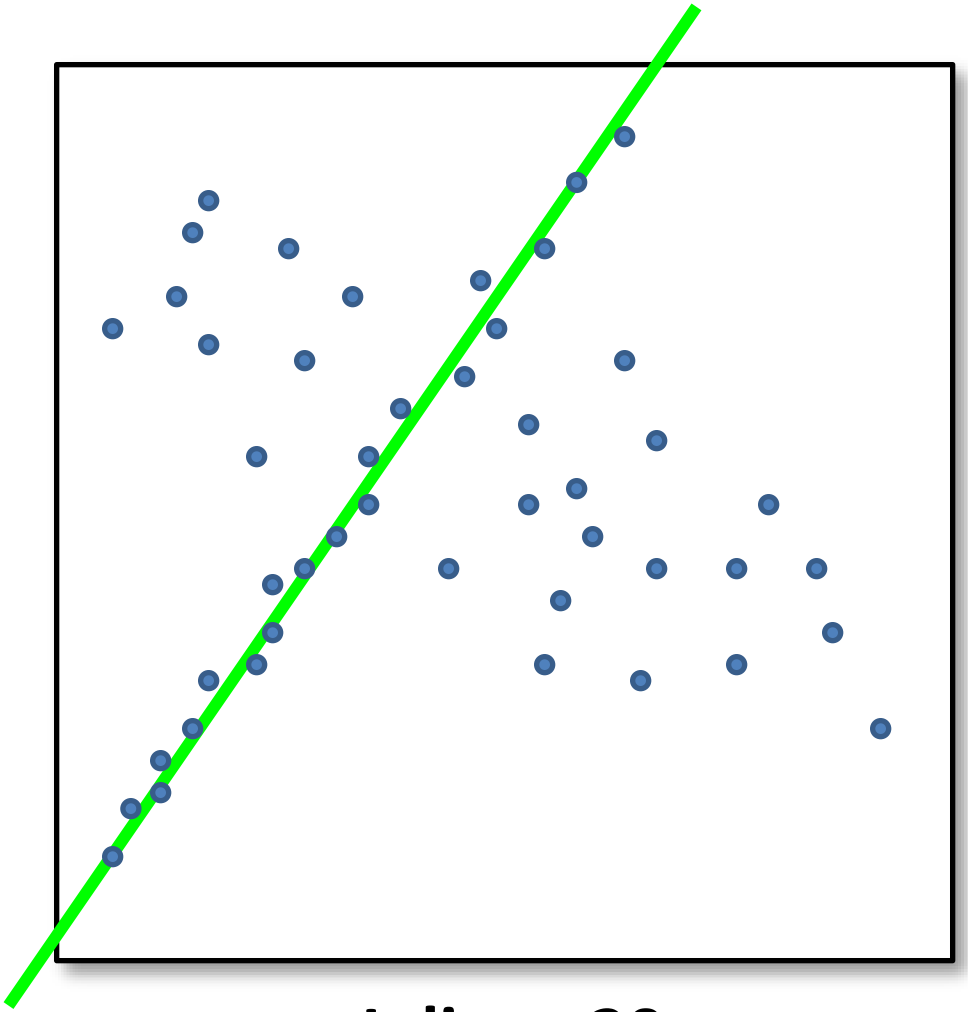


Counting inliers



Inliers: 3

Counting inliers

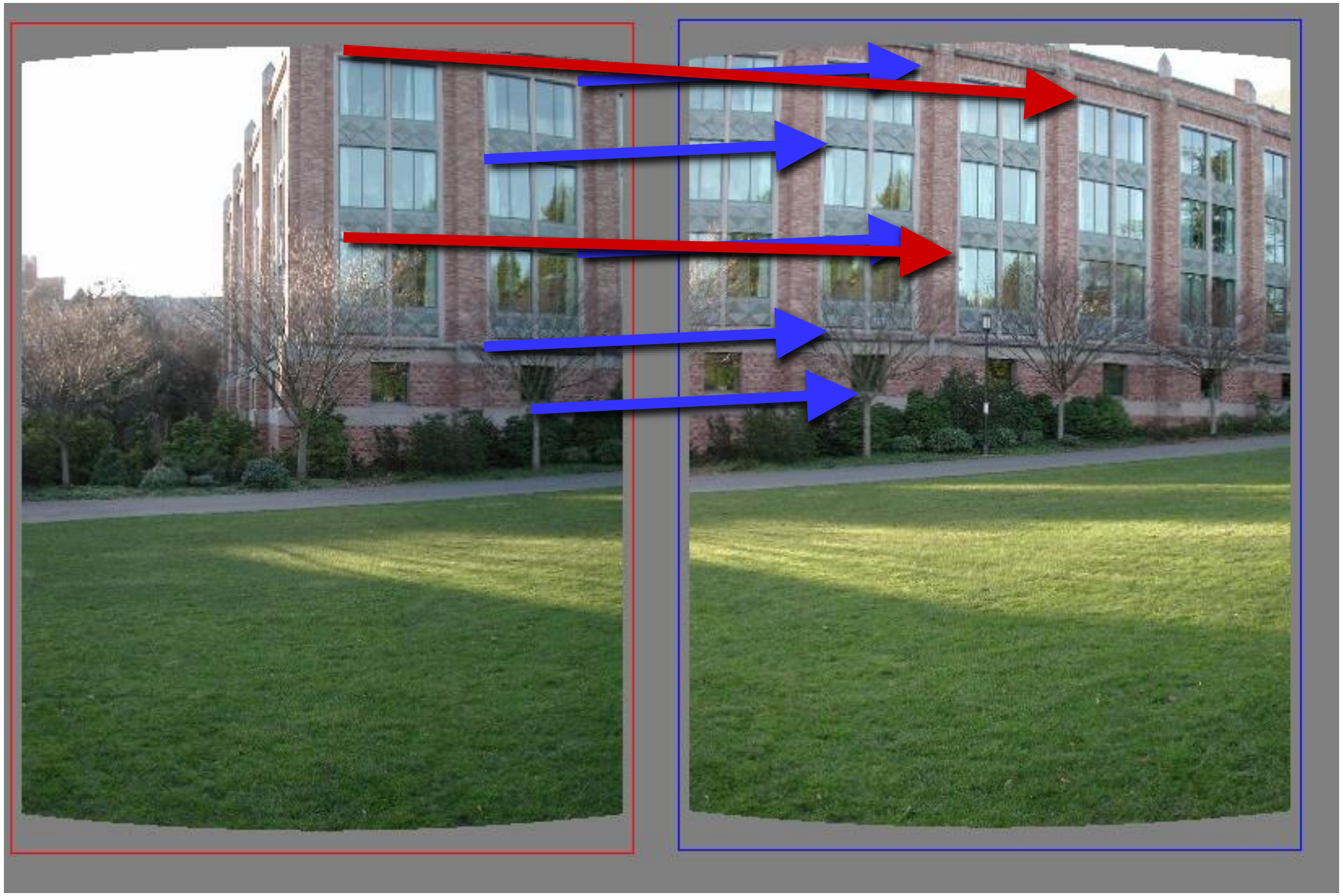


Inliers: 20

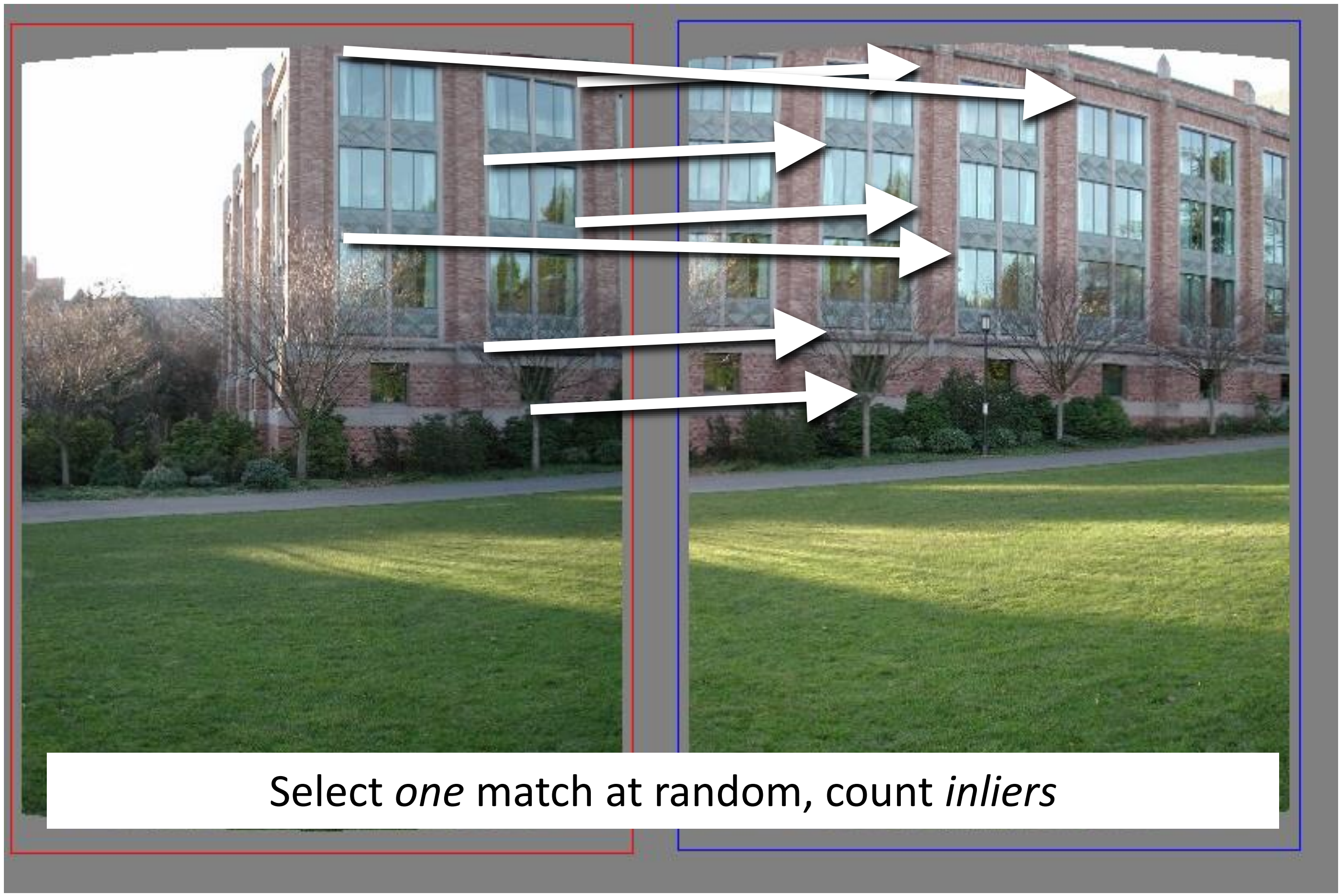
How do we find the best line?

- Unlike least-squares, no simple closed-form solution
- Hypothesize-and-test
  - Try out many lines, keep the best one
  - Which lines?

Translations

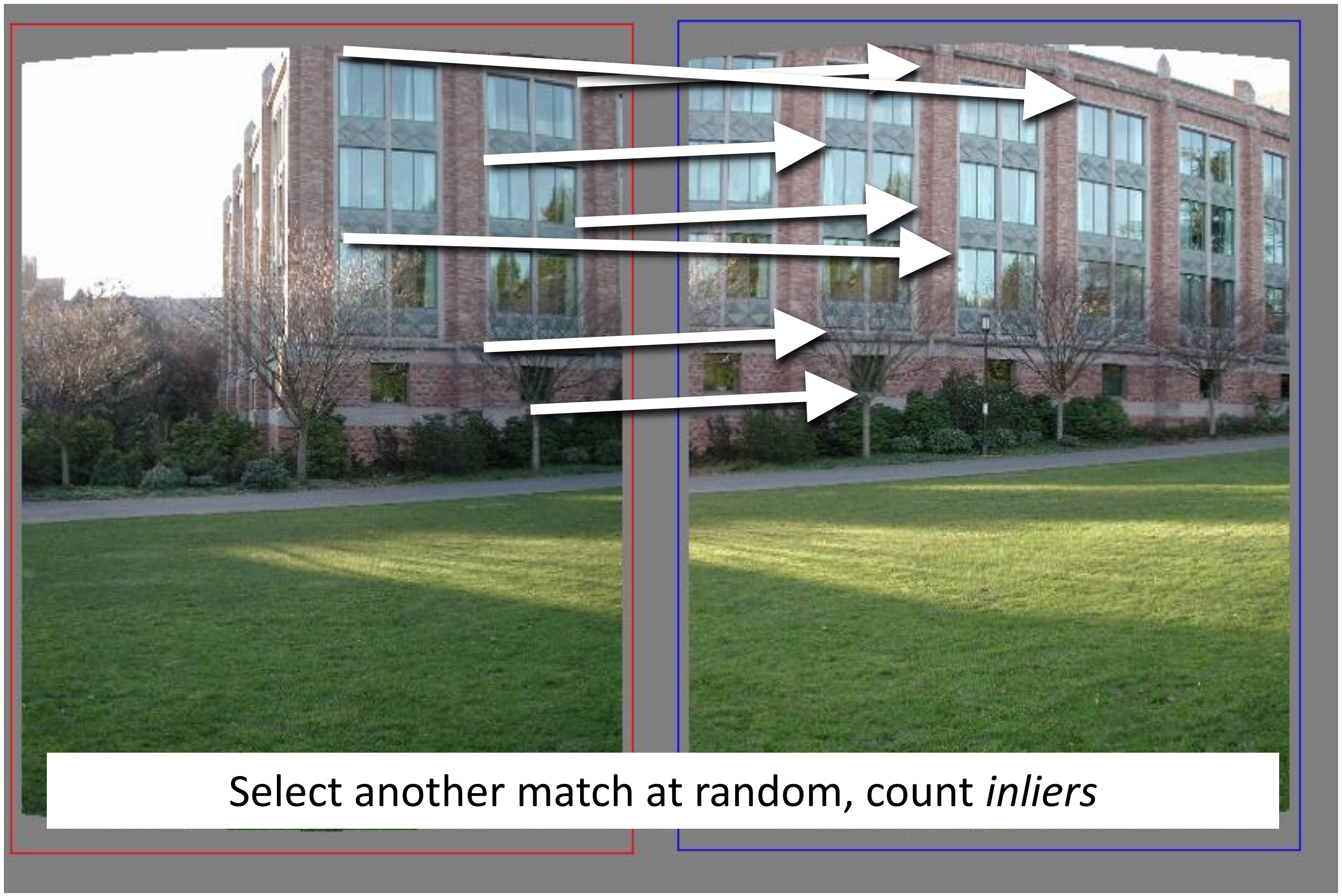


<u>RA</u>ndom <u>SA</u>mple <u>C</u>onsensus



Select *one* match at random, count *inliers*

RAndom SAmple Consensus



RANSAC

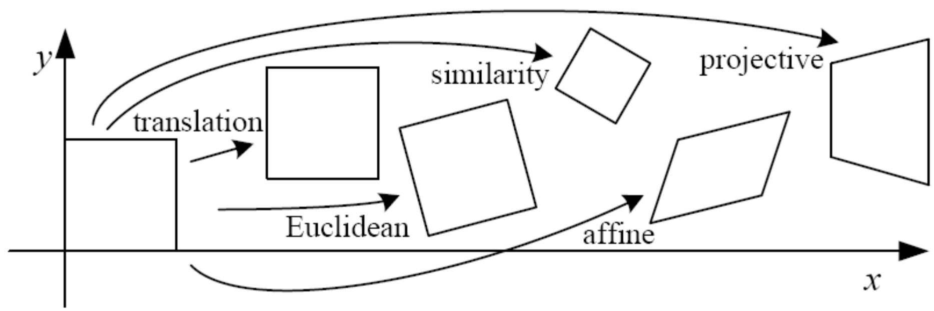
- Idea:
  - All the inliers will agree with each other on the translation vector; the (hopefully small) number of outliers will (hopefully) disagree with each other

  - “All good matches are alike; every bad match is bad in its own way.”

– Tolstoy via Alyosha Efros

RANSAC: How many samples?

- For alignment, depends on the motion model
  - Here, each sample is a correspondence (pair of matching points)



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	

RANSAC: How many samples?

- How many samples are needed?
  - Suppose w is fraction of inliers (points from line).
  - n points needed to define hypothesis (2 for lines)
  - k samples chosen.
- Prob. that a single sample of n points is correct: w^n
- Prob. that all k samples fail is: $(1-w^n)^k$

$\Rightarrow$ Choose k high enough to keep this below desired failure rate.

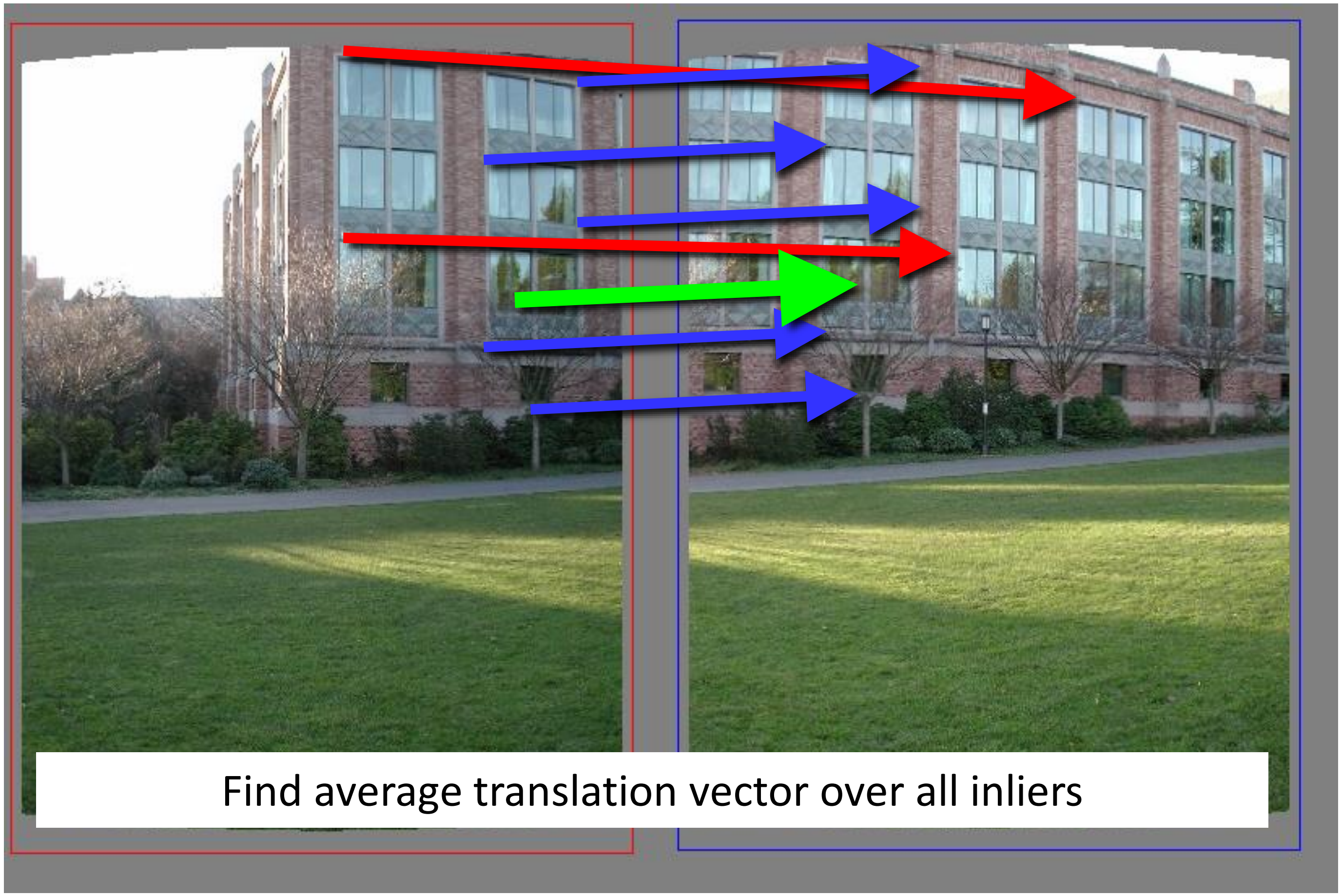
Slide credit: David Lowe

RANSAC: Computed k (p=0.99)

Sample size n	Proportion of outliers						
	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**

Slide credit: David Lowe

Final step: least squares fit



RANSAC pros and cons

- Pros
  - Simple and general
  - Applicable to many different problems
  - Often works well in practice
- Cons
  - Parameters to tune
  - Sometimes too many iterations are required
  - Can fail for extremely low inlier ratios

Summary

- Global geometric transforms
  - Homogenous coordinates
  - Linear -> affine -> homography
- Alignment (registration)
  - Least square problem
  - RANSAC