Homography

$$\begin{bmatrix} x_h \\ y_h \\ w_h \end{bmatrix} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

make this h_{22} then rewrite

$$x' = \frac{h_{00}x + h_{01}y + h_{02}}{h_{20}x + h_{21}y + 1} \quad \frac{x_h}{w_h}$$

$$y' = \frac{h_{10}x + h_{11}y + h_{12}}{w_h}$$

Not linear!

$$\begin{bmatrix} x_h \\ y_h \\ w_h \end{bmatrix} = \begin{bmatrix} h_{00} & h_{01} & h_{02} \\ h_{10} & h_{11} & h_{12} \\ h_{20} & h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

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

$$h_{00}x + h_{01}y + h_{02} = x'(h_{20}x + h_{21}y + h_{22})$$

✗ Residual: $(h_{00}x + h_{01}y + h_{02}) - x'(h_{20}x + h_{21}y + h_{22})$

$$= (h_{00}x + h_{01}y + h_{02}) - (h_{20}xx' + h_{21}yx' + h_{22}x')$$

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

y residual: $h_{10}x + h_{11}y + h_{12} - (h_{20}xy' + h_{21}yy' + h_{22}y')$

We had: $x' = \frac{x_h}{w_h}$ now $x_h = x'w_h$

$y' = \frac{y_h}{w_h}$ now $y_h = y'w_h$

Residuals:

$$x: \underbrace{h_{00}x + h_{01}y + h_{02}}_{x_h} - \underbrace{h_{20}xx' - h_{21}x'y - h_{22}x'}_{x' w_h}$$

$$y: h_{10}x + h_{11}y + h_{12} - h_{20}xy' - h_{21}yy' - h_{22}y'$$

$$\min \|Ah - b\|^2$$

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & x_1x_1' & x_1'y_1 & x_1' \\ 0 & 0 & 0 & x_1 & y_1 & 1 & x_1y_1' & y_1y_1' & y_1' \\ & & & & \vdots & & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & x_nx_n' & x_n'y_n & x_n' \\ 0 & 0 & 0 & x_n & y_n & 1 & x_ny_n' & y_ny_n' & y_n' \end{bmatrix} \begin{bmatrix} h_{00} \\ h_{01} \\ h_{02} \\ h_{10} \\ h_{11} \\ h_{12} \\ h_{20} \\ h_{21} \\ h_{22} \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

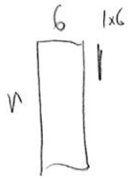
Minimizing $\|Ah - 0\|^2$

Trivial: $h = \vec{0}$

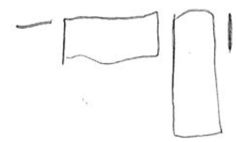
Solution: constrain $\|h\| = 1$

$$\min \|Ah - 0\|^2 \quad \text{S.T.} \quad \|h\| = 1$$

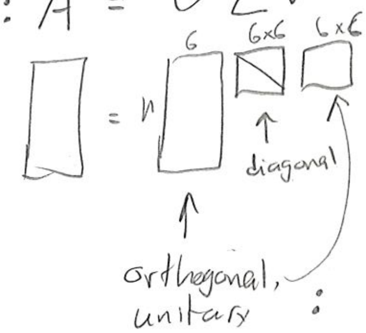
$$\min \|Ah\|^2 \quad \text{S.T.} \quad \|h\| = 1$$



$$\|Ah\|^2 = (Ah)^T (Ah) = h^T A^T A h$$



Singular value decomposition: $A = U \Sigma V^T$



$$h^T A^T A h = h^T V \Sigma U^T U \Sigma V^T h$$
$$= h^T V \Sigma \Sigma V^T h$$

orthogonal, unitary: $u_i^T \cdot u_j = 0$, $u_i^T u_i = 1$

$$\Sigma V^T h: \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \sigma_3 & \\ & & & \ddots \end{bmatrix} \begin{bmatrix} - v_1 - \\ - v_2 - \\ - v_3 - \\ \vdots \end{bmatrix} \begin{bmatrix} h \end{bmatrix}$$

$$= \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \sigma_3 & \\ & & & \ddots \end{bmatrix} \begin{pmatrix} v_1 h \\ v_2 h \\ v_3 h \\ \vdots \\ \vdots \end{pmatrix} = \begin{pmatrix} \sigma_1 v_1 h \\ \sigma_2 v_2 h \\ \sigma_3 v_3 h \\ \vdots \\ \vdots \end{pmatrix}$$

$\|h\| = 1$

$\|v_i\| = 1$

$v_i^T \cdot v_j = 1$

$h = v_3 \rightarrow$

$\sigma_1 v_1 v_3 \rightarrow 0$

$\sigma_2 v_2 v_3 \rightarrow 0$

$\sigma_3 v_3 v_3 \rightarrow \sigma_3$

$\vdots$

0

In Practice:

1. Compute SVD of A.
2. Find index of smallest σ_i in Σ
3. Take the ith column of V (ith row of V^T) as solution h.

RANSAC: The key Idea:

Points that fit the "true" model will agree.

Points that don't will not agree on some other, wrong model.

An idea: Generate all possible lines, see which one has the most agreeing points.

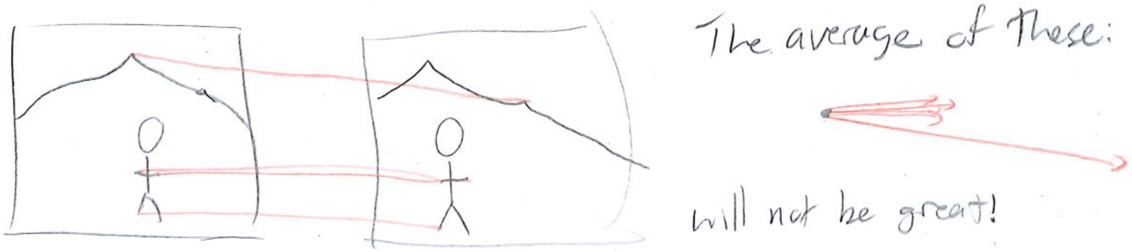
Runtime: $O(\infty)$

A better idea: all possible lines through 2 points:

for $p_1, p_2 \in P$:

RANSAC - RAndom SAmple Consensus

<u>Motivation</u> Fitting geometric transformations with least squares works well <u>if</u> there are no <u>outliers</u>.

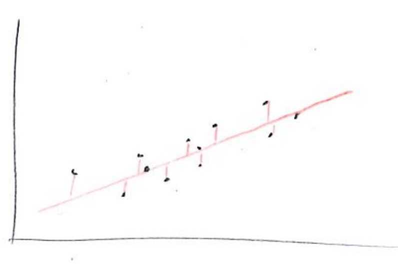


Fitting a transformation is a model fitting problem:

Given matches, find transformation

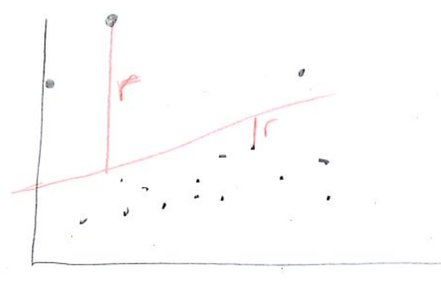
analogy: Given points, find a line.

When does least squares work well?



When errors are small and random.

(math answer): When errors are i.i.d Gaussian w/ zero mean.



Outliers have an outsized effect because of the square in the objective

$$\min_x \|Ax-b\| = \min_x \|Ax-b\|^T\|Ax-b\|$$

RANSAC: Algorithm

for $i = 0..K$:

  $d_i \in S$ random data points

  $M_i \leftarrow$ fit_model(d_i)

  inlier_count $\leftarrow \sum_{x_i, y_i \in D} \mathbb{1}(|M(x_i) - y_i| < \delta)$ #data points in agreement w/ M_i

  if inlier_count < best_count:

    best_count $\leftarrow$ inlier_count

    best_M $\leftarrow M_i$

    best_data $\leftarrow \{(x_i, y_i) : |M(x_i) - y_i| < \delta\}$ ← all data points in agreement w/ M_i

M_final = fit_model(best_data)

Parameters

K - # iterations (hypotheses)

S - # data points needed to fit a model

δ - inlier threshold

Choosing parameter values:

δ: based on expected inlier noise. Common case:

  Assume Gaussian w/ variance σ^2

  Let $\delta \approx 1$ or $2 \cdot \sigma$

S: based on specific problem:

- Linear regression - 2 x, y points
- Translation fitting - one match $(x, y) \leftrightarrow (x', y')$
- Affine - 3 matches
- Homography - 4 matches
- Ellipse (why not?) - 3 (x, y) points

Choosing Parameter Values (cont.):

K (# iterations) - suppose we want to find a set of s inliers with probability $\geq P$

Assume we can estimate the <u>inlier ratio</u>

$$r = \frac{\# \text{ inliers}}{\# \text{ data points}}$$

In one hypothesis,

$$P(\text{choose all inliers}) = r^s$$

$$P(\text{at least one outlier}) = 1 - r^s$$

↖ Bad Thing (at least one outlier)

Over K trials,

Bad Thing happens K times in a row

$$P(\text{at least one outlier all K trials}) = (1 - r^s)^K$$

Bad Thing doesn't happen K times in a row

$$P = P(\text{no outliers in at least one trial}) = 1 - (1 - r^s)^K$$

(no outliers in at least one trial) ≡ Success

What K do I need to make $P(\text{success}) \geq P$?

$$P \geq 1 - (1 - r^s)^K$$

$$1 - P \geq (1 - r^s)^K$$

$$\log(1 - P) \geq K \log(1 - r^s)$$

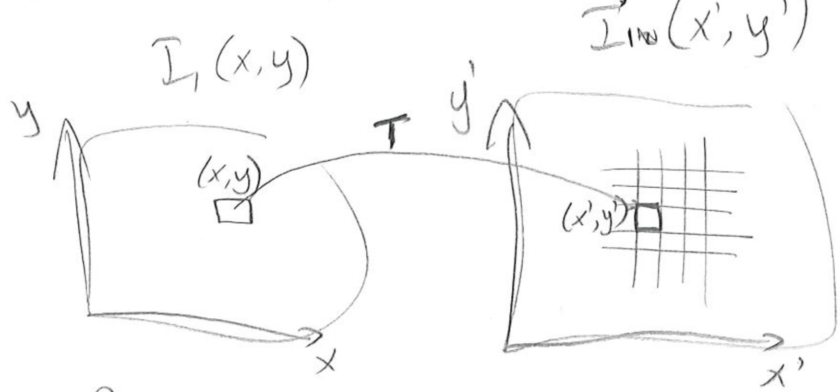
$$\boxed{\frac{\log(1-P)}{\log(1-r^s)} \geq K}$$

Sanity checks:

Lower Prob. of success → fewer iterations

More points to fit a model (s) → more iterations

Forward Warping



for x, y in I_1:

$x', y' = T \begin{bmatrix} x \\ y \end{bmatrix}$

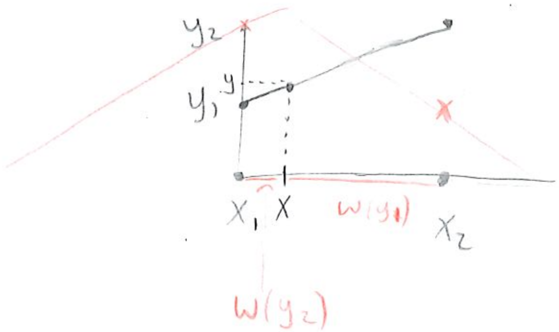
$I'_{1w}[x', y'] = I[x, y]$

Problem: what if x', y' are floats?

Possible answer: "splat" $I[x,y]$ to multiple pixels in $I'[x', y']$

Issues with:
- scale (e.g. T is a 16x uniform scale)
- holes remaining after splatting

Linear Interpolation

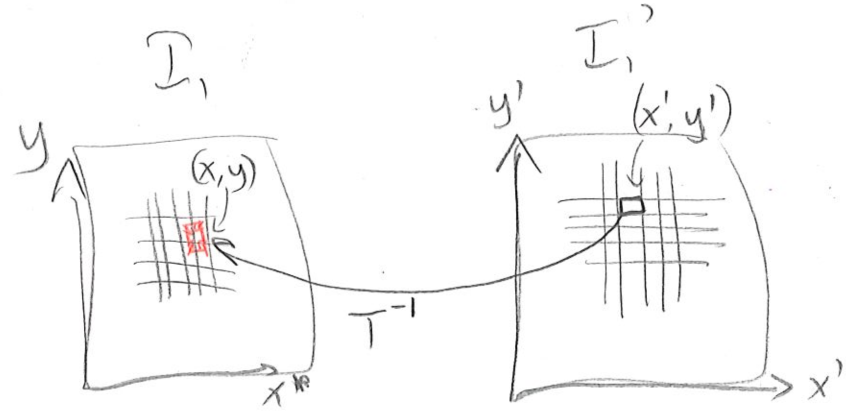


$$y = y_1(x_2 - x) + y_2(x - x_1)$$

If $x_1 = 0$, $x_2 = 1$,

$$y_1(1-x) + y_2(x)$$

Inverse Warping



for each (x', y') in I_1'

$x, y = T^{-1}\begin{bmatrix} x' \\ y' \end{bmatrix}$

$I_1'[x', y'] = \text{interpolate}\left(I_1, T^{-1}\begin{bmatrix} x' \\ y' \end{bmatrix}\right)$

Bilinear Interpolation - placing a tent fiter at non-integer coordinates!

Interpretations:
- a tent filter at non-integer coords
- weights determined by areas of rectangles at opposite corner
- interpolate linearly on two sides, then interpolate linearly between the two

$I'[x', y'] =$

$(x_2 - x)(y_1 - y)\, I[x_1, y_2]$

$+$