CS 664
Image Matching and Robust Fitting

Daniel Huttenlocher
Matching and Fitting

- Recognition and matching are closely related to fitting problems
- Parametric fitting can serve as more restricted domain for investigating questions of noise and outlier
  - Methods robust in presence of noise
- Two widely used techniques
  - RANSAC
  - Hough transform
- Generalized to matching and recognition
How Many “Good” Linear Fits?
RANSAC

- RANdom SAmple Consensus
  - Fischler and Bolles, 1981
- Select small number of data points and use to generate instance of model
  - E.g., fit to a line
- Check number of data points consistent with this fit
- Iterate until “good enough” consistent set
- Generate new fit from this set
RANSAC

**Objective**
- Robust fit of model to data set S which contains outliers

**Algorithm**
1. Randomly select a sample of \( s \) data points from S and instantiate the model from this subset.
2. 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 samples and defines the inliers of S.
3. If the subset of \( S_i \) is greater than some threshold \( T \), re-estimate the model using all the points in \( S_i \) and terminate.
4. If the size of \( S_i \) is less than \( T \), select a new subset and repeat the above.
5. After \( N \) trials the largest consensus set \( S_i \) is selected, and the model is re-estimated using all the points in the subset \( S_i \).
RANSAC Line Fitting Example

Task:
Estimate best line
RANSAC Line Fitting Example

Sample two points
RANSAC Line Fitting Example

Fit line
RANSAC Line Fitting Example

Total number of points within a threshold of line
RANSAC Line Fitting Example

Repeat, until get a good result
RANSAC Line Fitting Example

Repeat, until get a good result
RANSAC Line Fitting Example

Repeat, until get a good result
Choosing Number of Samples

- Choose $N$ samples so that, with probability $p$, at least one random sample is free from outliers
  - E.g. $p=0.99$

- Let $e$ denote proportion of outliers
  - Data points that do not fit the model within the distance threshold $t$

- Probability of selecting all inliers
  - Sampling without replacement, not independent
    - E.g., $D$ data points and $I$ inliers
Choosing Number of Samples

- Probability of $s$ samples all being inliers
  \[ \prod_{i=0}^{s-1} \frac{I-i}{D-i} \]
- For $s<<D$ approximate by $(I/D)^s$ or $(1-e)^s$
- Now want to choose $N$ so that, with probability $p$, at least one random sample is free from outliers
  \[ \left(1 - (1-e)^s\right)^N = 1 - p \]
Choosing Number of Samples

\[ N = \log(1 - p) / \log(1 - (1 - e)^s) \]

<table>
<thead>
<tr>
<th>s</th>
<th>5%</th>
<th>10%</th>
<th>20%</th>
<th>25%</th>
<th>30%</th>
<th>40%</th>
<th>50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>11</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>7</td>
<td>9</td>
<td>11</td>
<td>19</td>
<td>35</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>5</td>
<td>9</td>
<td>13</td>
<td>17</td>
<td>34</td>
<td>72</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>6</td>
<td>12</td>
<td>17</td>
<td>26</td>
<td>57</td>
<td>146</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>7</td>
<td>16</td>
<td>24</td>
<td>37</td>
<td>97</td>
<td>293</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>8</td>
<td>20</td>
<td>33</td>
<td>54</td>
<td>163</td>
<td>588</td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>9</td>
<td>26</td>
<td>44</td>
<td>78</td>
<td>272</td>
<td>1177</td>
</tr>
</tbody>
</table>
Adaptively Choosing $N$

- Fraction of outliers is often unknown a priori
  - Pick “worst” case, e.g. 50%, and adapt if more inliers are found

- $N=\infty$, $sample\_count = 0$
- While $N > sample\_count$ repeat
  - Choose a sample and count the number of inliers
  - Set $e = 1 - (\text{number of inliers})/(\text{total number of points})$
  - Recompute $N$ from $e$
  - Increment the $sample\_count$ by 1
- Terminate
Number of Samples II

- Make take more samples than one would think due to degenerate point sets
Number of Samples II

- These two points are inliers
And yet the estimate yielded is poor
Determine Potential Correspondences

- Compare interest points
  - E.g., similarity measure: SAD, SSD on small neighborhood

- Note: can use correlation score to bias the selection of the samples selecting matches with a better correlation score more often

- Note multiple matches for each point can be RANSAC’ed on (although this increases the proportion of outliers)
Example: Robust Computation

- Interest points (500/image)
- Putative correspondences (268)
- Outliers (117)
- Inliers (151)
- Final inliers (262)
Example: 2D Similarity Transformation

Set 1

Set 2
Example: 2D Similarity Transformation

Set 1

Set 2

Set of matches from some correlation function, lighter ones incorrect
Example: 2D Similarity Transformation

Two matches, used to infer transform,
Here: Top match correct, bottom incorrect
Example: 2D Similarity Transformation

Features mapped under transform do not align well
Example; 2D Similarity Transformation

On the other hand, if we pick two correct matches (modulo noise)
Example: 2D Similarity Transformation

Set 1

Set 2

Alignment is good!
Cost Function

- RANSAC can be vulnerable to the correct choice of the threshold
  - Too large all hypotheses are ranked equally
  - Too small leads to an unstable fit

- Same strategy can be followed with any modification of the cost function
Threshold too high
Threshold too high

This solution...
Threshold too high

Is as good as this solution
Threshold too low-no support
Cost Function

- Examples of other cost functions
  - Least Median Squares; i.e. take the sample that minimized the median of the residuals
  - MAPSAC/MLESAC use the posterior or likelihood of the data
  - MINPRAN (Stewart), makes assumptions about randomness of data
LMS

- Repeat M times:
  - Sample minimal number of matches to estimate two view relation
  - Calculate error of all data
  - Choose relation to minimize median of errors
Pros and Cons LMS

- **PRO**
  - Do not need any threshold for inliers
  - Can yield robust estimate of variance of errors

- **CON**
  - Cannot work for more than 50% outliers
Robust Maximum Likelihood Estimation

Random Sampling can optimize any function:

Better, robust cost function, MLESAC

Probability of data given instantiation of model
Maximum likelihood or MAP estimation
MLESAC/MAPSAC

This solution...
MLESAC/MAPSAC

Is better than this solution
MAPSAC

- Add in prior to get to MAP solution

- With MAPSAC one could sample less than the minimal number of points to make an estimate (using prior as extra information)

- Any posterior can be optimized; random sampling good for matching and function optimization
  - E.g. MAPSAC is a way to optimize objective functions regardless of outliers or not
Underlying Assumptions

- LMS criterion
  - Minimum fraction of inliers is known
- RANSAC criterion
  - Inlier bound is known
Not Necessarily Desirable

- Structures may be “seen” in data despite unknown scale and large outlier fractions
- Potential unknown properties:
  - Sensor characteristics
  - Scene complexity
  - Performance of low-level operations
- Problems:
  - Handling unknown scale
  - Handling varying scale

45% random outliers
44% from 1st structure
11% from 2nd structure
Goal

- A robust objective function, suitable for use in random-sampling algorithm, that is
  - Invariant to scale,
  - Does not require a prior lower bound on the fraction of inliers
Approaches

- MINPRAN (Stewart, IEEE T-PAMI Oct 1995)
  - Discussed briefly today

- MUSE (Stewart, IEEE CVPR 1996)
  - Based on order statistics of residuals
  - Focus of today’s presentation
  - Code available in VXL and on the web

- Other order-statistics based methods:
  - Lee, Meer and Park, PAMI 1998
  - Bab-Hadiashar and Suter, Robotica 1999

- Kernel-density techniques
  - Chen-Meer ECCV 2002
  - Wang and Suter, PAMI 2004
  - Subbarao and Meer, RANSAC-25 2006
MINPRAN: Minimize Probability of Randomness

65% outliers

26 inliers within +/- 8 units of random-sample-generated line

72 inliers within +/- 7 units of random-sample-generated line

55 inliers within +/- 2 units of random-sample-generated line
MINPRAN: Probability Measure

- Probability of having at least $k$ points within error distance +/- $r$ if all errors follow a uniform distribution within distance +/- $Z_0$:

$$F(r, k, N) = \sum_{i=k}^{N} \binom{N}{i} \left( \frac{r}{Z_0} \right)^i \left( 1 - \frac{r}{Z_0} \right)^{N-i}$$

- Lower values imply it is less likely that the residuals are uniform

- Good estimates, with appropriately chosen values of $r$ (inlier bound) and $k$ (number of inliers), have extremely low probability values
MINPRAN: Discussion

- O(S N log N + N^2) algorithm
- Good results for single structure
- Limitations
  - Requires a background distribution
  - Tends to “bridge” discontinuities
  - Quadratic running time
MUSE: Ordered Residuals of Good and Bad Estimates

Objective function should capture:
- Ordered residuals are lower for inliers to “good” estimate than for “bad” estimate
- Transition from inliers to outliers in “good” estimate
Voting Based Schemes

- In the Hough transform each feature in image “votes” for those instantiations of model that are consistent with it.
- Classic case is fitting lines to point features
  - A given feature point defines a pencil of possible lines through it
  - Several collinear (or nearly collinear) feature points will agree on one (or a few similar) lines
  - Conventional to use $r, \theta$ parameterization of line
Hough Space

- Each \((x,y)\) point in Cartesian plane defines constraint on possible lines

\[ x \cos \theta + y \sin \theta = r \]

  - Sinusoidal curve in \(r,\theta\) plane

- Analogous for finding circles

\[ (x - a)^2 + (y - b)^2 = r \]

  - But space now three-dimensional
Accumulator Array

- Discretize Hough parameter space
  - Problematic for higher dimensions
- Increment counts in all buckets that are intersected by the parameter values
  - Analogous to line or curve-drawing on pixel or voxel grid
- To allow for uncertainty in the measured values may make cells larger or increment values of neighboring cells
  - Fractional increments
    - Analogous to anti-aliasing
Classical Line Example

- Each edge point votes for sinusoidal curve of buckets in Hough accumulator array
  - Peaks corresponding to 8 lines defined by these edges
Finding the Lines

- Threshold peaks in accumulator array
  - Common to use relative threshold, fraction of biggest peak
- Some form of non-maximum suppression or “thinning”
Uncertainty in Detected Lines

- Bin size affects uncertainty, causing multiple similar lines to be found

- Noise in data causes incorrect estimates

- Problem: when does random noise produce peaks in the accumulator array?
  - Random points in plane each generating curve in Hough space
    - Get peaks of some size at random
Small Amounts of Noise
Small Amounts of Missing Data
Real Image Example

- 70% relative threshold on peak size
Real Image Example

- 40% relative threshold on peak size
Generalized Hough Transform

- Model as template and each image point votes for all possible models at that point
  - For instance binary model under translation vote for all any way of placing model that has this image point an edge

- Pre-compute a “lookup table” for incrementing Hough array
  - For translation, offsets of each point to some fixed origin

- Often more votes per image pixel, with more chance of randomly occurring peaks