Model Fitting

Introduction to Computer Vision
CSE 152
Lecture 11

What to do with edges?
• Segment linked edge chains into curve features (e.g., line segments).
• Group unlinked or unrelated edges into lines (or curves in general).
• Accurately fitting parametric curves (e.g., lines) to grouped edge points.

Announcements
• Homework 3 is due May 10, 11:59 PM
• Reading:
  – Chapter 10: Grouping and Model Fitting

Hough Transform
[ Patented 1962 ]

Finding lines in an image

Connection between image (x,y) and Hough (m,b) spaces
• A line in the image corresponds to a point in Hough space

Finding lines in an image

Connection between image (x,y) and Hough (m,b) spaces
• A line in the image corresponds to a point in Hough space
• What does a point (x₀, y₀) in the image space map to?
The equation of any line passing through (x₀, y₀) has form
y = mx₀ + b.
This is a line in Hough space: b = -x₀m + y₀.
Hough Transform Algorithm

- Typically use a different parameterization
  \[ d = x \cos \theta + y \sin \theta \]
  - \( d \) is the perpendicular distance from the line to the origin
  - \( \theta \) is the angle this perpendicular makes with the \( x \) axis

- Basic Hough transform algorithm
  1. Initialize \( H[d, \theta]=0 \); \( H \) is called accumulator array
  2. For each edge point \( I[x,y] \) in the image
     - For \( \theta = 0 \) to 180
       - \( d = x \cos \theta + y \sin \theta \)
       - \( H(d, \theta) \) += 1
  3. Find the value(s) of \((d, \theta)\) where \( H[d, \theta] \) is the global maximum
  4. The detected line in the image is given by \( d = x \cos \theta + y \sin \theta \)

What’s the running time (measured in \# votes)?

Hough Transform: 20 colinear points

Hough Transform: Random points

Hough Transform: “Noisy line”

Extension: Oriented Edges

**procedure Hough\((\{x,y,\theta\})\):**

1. Clear the accumulator array.
2. For each detected edge at location \((x,y)\) and orientation \( \theta = \tan^{-1} \frac{m_y}{m_x} \)
   - Compute the value of \( d = x \cos \theta + y \sin \theta \)
   - and increment the accumulator corresponding to \((\theta, d)\).
3. Find the peaks in the accumulator corresponding to lines.
4. Optionally re-fit the lines to the constituent edges.

Algorithm 4.2 Outline of a Hough transform algorithm based on oriented edge segments.

Hough Transform for Curves

**Generalized Hough Transform**

- The Hough transform can be generalized to detect any curve that can be expressed in parametric form:
  \[ y = f(x, a_1, a_2, \ldots, a_p) \]
  or
  \[ g(x,y,a_1,a_2,\ldots,a_p) = 0 \]
  - \( a_1, a_2, \ldots, a_p \) are the parameters
  - The parameter space is \( p \)-dimensional
  - The accumulating array is \( \text{large} \)
Example: Finding circles

Equation for circle is

\[(x - x_c)^2 + (y - y_c)^2 = r^2\]

where the parameters are the circle’s center \((x_c, y_c)\) and radius \(r\).

Three dimensional generalized Hough space.

Given an edge point \((x, y)\),
1. Loop over all values of \((x_c, y_c)\),
2. Compute \(r\)
3. Increment \(H(x_c, y_c, r)\)

Transmission Electron Microscopy (TEM) Image of Keyhole Limpet Hemocyanin (KLH) with detected particles

3D Maps of KLH

Three-dimensional maps of KLH at a resolution of 23.5 Å reconstructed using particles extracted either manually or automatically as described in the text. (a), (b) The side- and top-view of a 3D map reconstructed from a set of 1042 manually selected particle images.

Processing in Stage 1 for KLH

- Canny edge detection.
- A sequence of ordered Hough transforms (HTs) is applied in order from the computationally simplest one to the most complex one.
- Edges covered by the detected shapes are removed immediately from edge images following the application of the last HT.

Picking KLH Particles in Stage 1

Zhu et al., IEEE Transactions on Medical Imaging, 2003

Line Fitting
Line Fitting

Given \( n \) points \((x_i, y_i)\), estimate parameters of line
\[ ax_i + by_i - d = 0 \]
subject to the constraint that
\[ a^2 + b^2 = 1 \]
Note: \( ax_i + by_i - d \) is distance from \((x_i, y_i)\) to line.

Cost Function:
Sum of squared distances between each point and the line
with respect to \((a, b, d)\).

1. Minimize \( E \) with respect to \( d \):
   \[
   \frac{\partial E}{\partial d} = 0 \Rightarrow d = \frac{1}{n} \sum_{i=1}^{n} ax_i + by_i = \bar{x} + b \bar{y}
   \]
   Where \((\bar{x}, \bar{y})\) is the mean of the data points

2. Substitute \( d \) back into \( E \)
   where \( n = (a, b)^T \).

3. Minimize \( E = \| n - \bar{u} \|^2 U n^T S n \) with respect to \( a, b \)
subject to the constraint \( n^T n = 1 \). Note that \( S \) is given by
   \[
   S = \begin{pmatrix}
   \sum x_i^2 - n \bar{x}^2 & \sum x_i y_i - n \bar{x} \bar{y} \\
   \sum x_i y_i - n \bar{x} \bar{y} & \sum y_i^2 - n \bar{y}^2
   \end{pmatrix}
   \]
   which is real, symmetric, and positive definite

4. This is a constrained optimization problem in \( n \). Solve with Lagrange multiplier
   \[
   L(n) = n^T S n - \lambda (n^T n - 1)
   \]
   Take partial derivative (gradient) w.r.t. \( n \) and set to 0.
   \[
   \nabla L = 2S n - 2 \lambda n = 0
   \]
or
   \[
   S n = \lambda n
   \]
   \( n = (a, b) \) is an Eigenvector of the symmetric matrix \( S \)
   (the one corresponding to the smallest Eigenvalue).

5. \( d \) is computed from Step 1.

RANSAC

Slides shamelessly taken from
Frank Dellaert and Marc Pollefeys and modified

Simpler Example

• Fitting a straight line

Discard Outliers

• No point with \( d > t \)
• RANSAC:
  – RANdom SAmple Consensus
  – Fischler & Bolles 1981
  – Copes with a large proportion of outliers
Main Idea

• Select 2 points at random
• Fit a line
• “Support” = number of inliers
• Line with most inliers wins

Why will this work?

• Best line has most support
  – More support -> better fit

RANSAC

Objective

Robust fit of model to 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 Si which are within a distance threshold t of the model. The set Si is the consensus set of samples and defines the inliers of S.
(iii) If the size of Si is greater than some threshold T, re-estimate the model using all the points in Si and terminate
(iv) If the size of Si is less than T, select a new subset and repeat the above.
(v) After N trials the largest consensus set Si is selected, and the model is re-estimated using all the points in the subset Si

Number of trials

Choose N (number of trials) so that, with probability p, at least one random sample is free from outliers. e.g., p=0.99

\[
\left(1 - (1 - e)^s\right)^N = 1 - p
\]

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

where:

- e: proportion of outliers
- s: Number of points needed for the model

Number of inliers threshold

• Typically, terminate when inlier ratio reaches expected ratio of inliers

\[
T = (1 - e)N
\]

Distance threshold

Choose threshold t so probability for inlier is α (e.g., 0.95)

• Often empirically
• Zero-mean Gaussian noise \( \sigma \) then \( d^2 \) follows \( \chi^2_m \) distribution with \( m=\text{codimension of model} \)

\[
\text{Codimension} \quad \text{Model} \quad \text{t}^2
\]

<table>
<thead>
<tr>
<th>Codimension</th>
<th>Model</th>
<th>\text{t}^2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E, F, 2D line</td>
<td>3.84(\mu^2)</td>
</tr>
<tr>
<td>2</td>
<td>P</td>
<td>5.99(\mu^2)</td>
</tr>
</tbody>
</table>
Using RANSAC to estimate the Fundamental Matrix

• What is the model?
• What is the sample size and where do the samples come from?
• What distance do we use to compute the consensus set?
• How often do outliers occur?

Other models

• 2D motion models
• Typically: points in two images
• Candidates:
  – Translation
  – Euclidean
  – Similarity
  – Affine
  – Projective

Feature Detection and Matching

Input Images

Detected Corners

Simple Matching

Simple Matching Including Outlier Rejection
Mosaicing: Homography Estimate with RANSAC

www.cs.cmu.edu/~dellaert/mosaicking

Next Lecture

- Motion
- Reading:
  - Section 10.6.1: Optical Flow and Motion
  - Section 10.6.2: Flow Models
  - Introductory Techniques for 3-D Computer Vision, Trucco and Verri
    - Chapter 8: Motion