Hough Transform - Overview

Key Concepts to Master

1. Objective of Hough Transform

Purpose:

- Detect parametric shapes (lines, circles, ellipses) in images
- Robust to noise and gaps in edge data
- Group individual edge pixels into higher-level features

Why it works:

- Transforms detection problem into peak-finding problem
- Collinear points vote for same parameters
- Outliers don't form peaks

2. Core Principle

Image Space → Parameter Space Transformation

For lines:

- Image space: Points (x, y)
- Parameter space: Line parameters (m, b) or (ρ, θ)

Key insight:

- One point in image space → Curve in parameter space
- Collinear points → Curves intersect at one point in parameter space
- Intersection point = line parameters

3. Line Representations

Slope-Intercept Form

```
y = mx + b
```

- m = slope
- b = y-intercept

Problems:

- Cannot represent vertical lines (m = ∞)
- Unbounded parameter space for steep lines
- Not used in Hough Transform

Normal (ρ-θ) Form ★ PREFERRED

```
ρ = x·cos(θ) + y·sin(θ)
```

Parameters:

- ρ (rho): Perpendicular distance from origin to line
- θ (theta): Angle of perpendicular (0° to 180°)

Advantages:

- Can represent ALL lines (including vertical)
- Bounded parameter space
- $\rho \in [-D, D]$ where D = diagonal of image
- $\theta \in [0°, 180°]$

Conversion between forms:

```
Given: y = mx + b
ρ = b / √(1 + m²)
θ = arctan(-1/m)
```

4. Hough Transform Algorithm

For each edge pixel (xi, yi):

```
For θ = 0° to 180° (in small increments):
    ρ = xi·cos(θ) + yi·sin(θ)
    Accumulator[ρ, θ] += 1  // Vote for this line
```

After processing all pixels:

```
Find peaks in accumulator array
Each peak (ρ*, θ*) represents a detected line
```

5. Accumulator Array

2D array in parameter space:

- Dimensions: $[\rho\_bins \times \theta\_bins]$
- Each cell = **accumulator cell** = one bin in (ρ, θ) space
- Value = number of edge pixels voting for that line

Example dimensions:

- θ: 180 bins (1° resolution)
- ρ: 2D bins where D = image diagonal

6. Parameter Space Mapping

Key transformations:

One point (x_0, y_0) →

$$\rho = x_0 \cdot \cos(\theta) + y_0 \cdot \sin(\theta)$$

- For varying θ, traces sinusoidal curve in (ρ, θ) space

Two collinear points →

- Both curves intersect at same (ρ, θ)
- Intersection = line containing both points

Three collinear points →

- All three curves intersect at same (ρ, θ)

- Strong peak in accumulator!

Hand Calculation Examples

Example 1: Simple point

Given: Point (2, 3), find ρ for θ = 0°, 45°, 90°

θ = 0°:

```
ρ = 2·cos(0°) + 3·sin(0°)
ρ = 2·(1) + 3·(0) = 2
```

θ = 45°:

```
ρ = 2·cos(45°) + 3·sin(45°)
ρ = 2·(√2/2) + 3·(√2/2)
ρ = √2 + 1.5√2 = 2.5√2 ≈ 3.54
```

θ = 90°:

```
ρ = 2·cos(90°) + 3·sin(90°)
ρ = 2·(0) + 3·(1) = 3
```

Example 2: Collinear points

Given: Points (0,0), (1,1), (2,2) - clearly on line y=x

Expected line parameters:

- Line y = x has slope m = 1
- In normal form: 45° line through origin
- Should get θ ≈ 45°, ρ ≈ 0

Check point (1,1) at θ = 45°:

```
ρ = 1·cos(45°) + 1·sin(45°)
ρ = √2/2 + √2/2 = √2 ≈ 1.41
```

Note: ρ ≠ 0 because our origin is at (0,0), not at perpendicular

Example 3: Vertical line

Given: Points (5,0), (5,1), (5,2) - vertical line at x=5

Expected: θ = 0°, ρ = 5

Check point (5,1) at θ = 0°:

```
ρ = 5·cos(0°) + 1·sin(0°)
ρ = 5·(1) + 1·(0) = 5  +
```

Example 4: Horizontal line

Given: Points (0,3), (1,3), (2,3) - horizontal line at y=3

Expected: θ = 90°, ρ = 3

Check point (1,3) at θ = 90°:

```
ρ = 1·cos(90°) + 3·sin(90°)
ρ = 1·(0) + 3·(1) = 3  +
```

Visualization

Image Space vs Parameter Space

```
Image Space (x, y):          Parameter Space (ρ, θ):

·   ·   ·                    \    /
    ·                         \  /
        ·                      \/  ← intersection = line
            ·                  /\
                ·             /  \

Single line                  Sinusoids from each point
                             intersect at line params
```

MATLAB Quick Reference

```
% Basic Hough Transform workflow

% 1. Get edge image
edges = edge(img, 'canny');

% 2. Compute Hough Transform
[H, theta, rho] = hough(edges);
% H = accumulator array
% theta = θ values (degrees)
% rho = ρ values (pixels)

% 3. Visualize accumulator
imshow(H, [], 'XData', theta, 'YData', rho);
xlabel('\theta (degrees)');
ylabel('\rho (pixels)');

% 4. Find peaks
peaks = houghpeaks(H, numpeaks);

% 5. Extract lines
lines = houghlines(edges, theta, rho, peaks, ...
    'FillGap', 20, 'MinLength', 30);

% 6. Display detected lines
imshow(img); hold on;
```

```
for k = 1:length(lines)
    xy = [lines(k).point1; lines(k).point2];
    plot(xy(:,1), xy(:,2), 'LineWidth', 2, 'Color', 'green');
end
```

Study Checklist

- [ ] Understand objective: detect lines robustly
- [ ] Know both line representations
- [ ] Prefer ρ-θ form (can handle all lines)
- [ ] Can calculate ρ for given (x, y, θ)
- [ ] Understand voting mechanism
- [ ] Know what accumulator cell represents
- [ ] Can explain image→parameter space mapping
- [ ] Know why collinear points intersect
- [ ] Can sketch parameter space curves

Common Exam Questions

Q1: What is the objective of Hough Transform?

- Detect parametric shapes (lines) robustly despite noise/gaps

Q2: Why use ρ-θ instead of slope-intercept?

- Can represent vertical lines
- Bounded parameter space

Q3: What is an accumulator cell?

- Single bin in parameter space
- Counts votes for that particular line

Q4: Given point (x,y) and angle θ, find ρ

- $\rho = x \cdot \cos(\theta) + y \cdot \sin(\theta)$

Q5: Why do collinear points create a peak?

- Each point creates sinusoid in parameter space
- All sinusoids intersect at line's (ρ, θ)

Practice Problems

1. Calculate ρ for point (3, 4) at θ = 0°, 30°, 60°, 90°
2. Three points (0,1), (1,2), (2,3) are collinear. Find their line parameters.
3. Convert line y = 2x + 3 to ρ-θ form
4. Sketch parameter space curves for two collinear points

Answers

Common Exam Questions

Q1: What is the objective of Hough Transform? Detect parametric shapes (especially lines) in images by transforming edge pixels into a parameter space and finding peaks. It is robust to noise, gaps in edges, and partial occlusions because collinear points independently vote for the same line parameters.

Q2: Why use rho-theta instead of slope-intercept? Slope-intercept `y = mx + b` cannot represent vertical lines (m = infinity) and has unbounded parameter space. The normal form `rho = x*cos(theta) + y*sin(theta)` handles all line orientations with bounded parameters: `rho in [-D, D]`, `theta in [0, 180)`.

Q3: What is an accumulator cell? One bin in the discretized (rho, theta) parameter space. Its value counts how many edge pixels voted for that particular line. A high count (peak) means many edge pixels are collinear along that line.

Q4: Given point (x,y) and angle theta, find rho Direct substitution: `rho = x*cos(theta) + y*sin(theta)`.

Q5: Why do collinear points create a peak? Each edge pixel (x,y) traces a sinusoidal curve in (rho, theta) space. If multiple pixels lie on the same line, their sinusoids all pass through the same (rho, theta) point, creating a peak in the accumulator.

Practice Problems

1. Calculate rho for point (3, 4) at theta = 0, 30, 60, 90 degrees

- `theta=0: rho = 3*cos(0) + 4*sin(0) = 3*1 + 4*0 = 3`
- `theta=30: rho = 3*cos(30) + 4*sin(30) = 3*(0.866) + 4*(0.5) = 2.598 + 2.0 = 4.598`
- `theta=60: rho = 3*cos(60) + 4*sin(60) = 3*(0.5) + 4*(0.866) = 1.5 + 3.464 = 4.964`
- `theta=90: rho = 3*cos(90) + 4*sin(90) = 3*0 + 4*1 = 4`

2. Points (0,1), (1,2), (2,3) are collinear. Find line parameters. The line is y = x + 1 (slope 1, intercept 1). To find (rho, theta): all three points should give the same rho at the correct theta. Using `rho = x*cos(theta) + y*sin(theta)` and checking theta = 135 degrees (perpendicular to slope-1 line):

- `(0,1): rho = 0*cos(135) + 1*sin(135) = 0.707`
- `(1,2): rho = 1*(-0.707) + 2*(0.707) = -0.707 + 1.414 = 0.707`
- `(2,3): rho = 2*(-0.707) + 3*(0.707) = -1.414 + 2.121 = 0.707` All agree: **theta = 135 degrees, rho = sqrt(2)/2 = 0.707**.

3. Convert line y = 2x + 3 to rho-theta form Using formulas: `rho = b / sqrt(1 + m^2) = 3 / sqrt(1 + 4) = 3/sqrt(5) = 1.342`. `theta = arctan(-1/m) = arctan(-0.5) = 180 - 26.57 = 153.43 degrees` (adjusting to [0,180) range since the line has positive intercept).

4. Sketch parameter space curves for two collinear points Each point traces a sinusoid `rho = x*cos(theta) + y*sin(theta)` in (rho, theta) space. Two collinear points produce two sinusoids that intersect at exactly one point -- this intersection is the (rho, theta) of the line through both points. The sketch shows two sine-like curves crossing at a single peak location.

Related Topics

- Lecture 5: Hough Transform
- Edge Detection (preprocessing step)
- Feature Detection (alternative approach)