ECE 4973: Lecture 15
Hough Transform

Samuel Cheng
Slide credits: S. Narasimhan

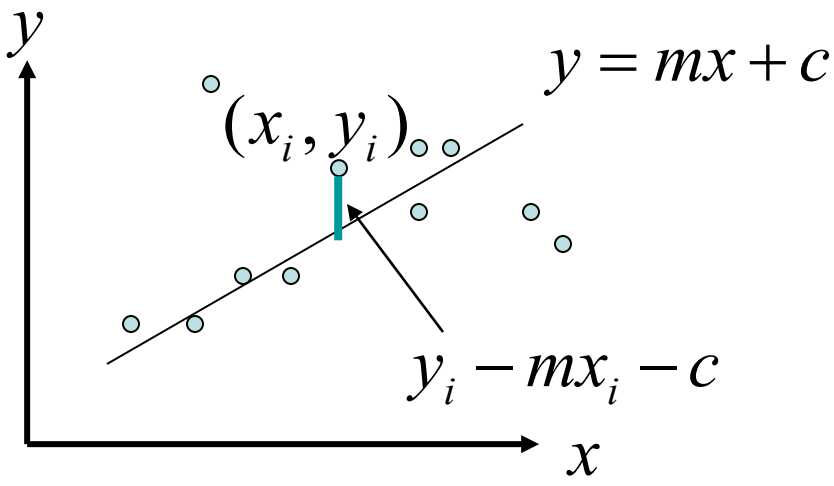
Hough transform

- What?
  - Find parametrizable outlined shapes from scene
    - Such as lines, circles
- How?
  - Outline detection preprocessing (e.g., Canny)
  - Transform to parameter space (more later)
  - Find maximum response

Motivation: Finding Lines from Edges/Points

Given: Many (x_i, y_i) pairs

Find: Parameters (m, c)



Minimize: Average square distance:

$$E = \sum_i \frac{(y_i - mx_i - c)^2}{N}$$

Using:

$$\frac{\partial E}{\partial m} = 0 \quad \& \quad \frac{\partial E}{\partial c} = 0$$

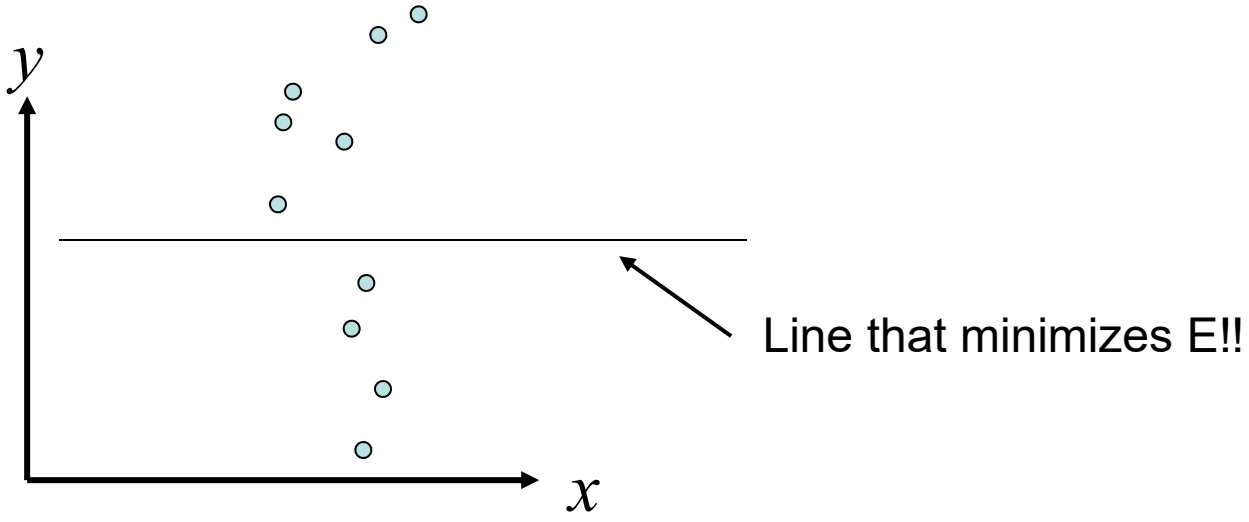
Note:

$$\bar{y} = \frac{\sum_i y_i}{N} \qquad \bar{x} = \frac{\sum_i x_i}{N}$$

$$c = \bar{y} - m\,\bar{x}$$

$$m = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sum_i (x_i - \bar{x})^2}$$

Problem with least square method

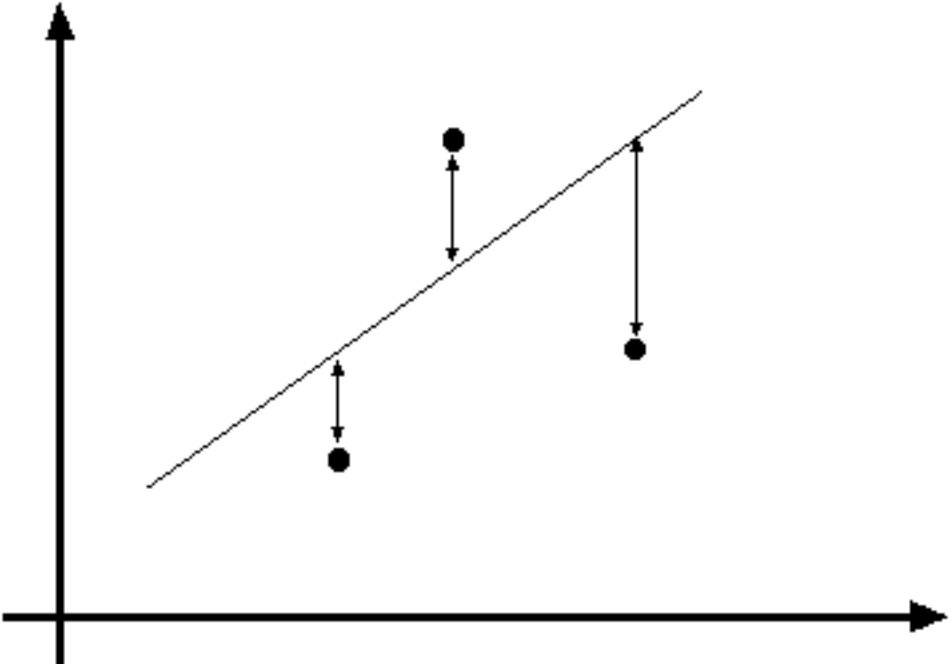


Solution: Use a different parameterization

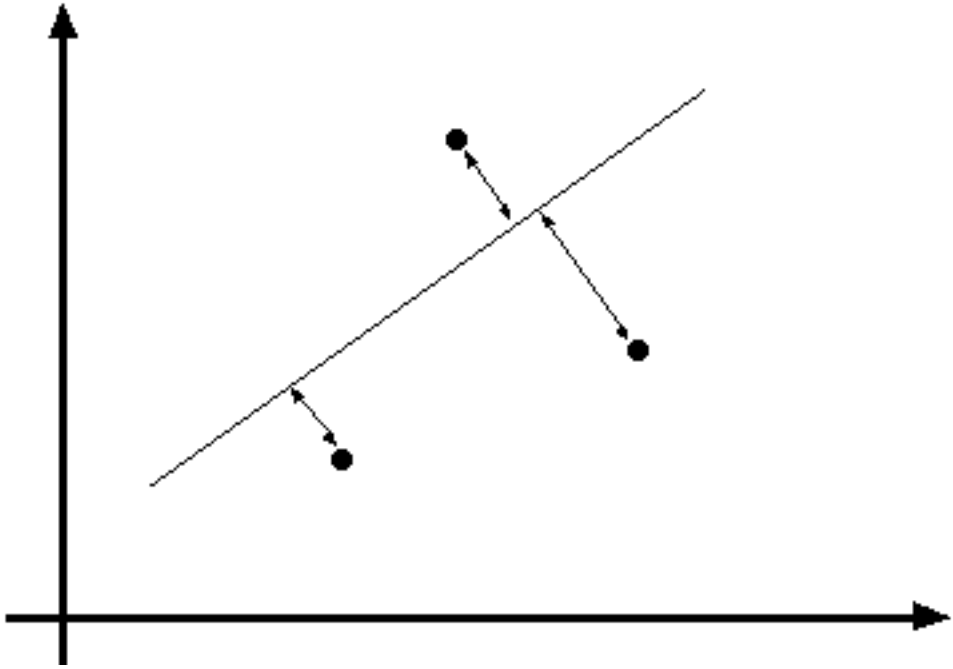
(same as the one we used in computing Minimum Moment of Inertia)

$$E = \frac{1}{N}\sum_i (\rho - x_i \cos\theta + y_i \sin\theta)^2$$

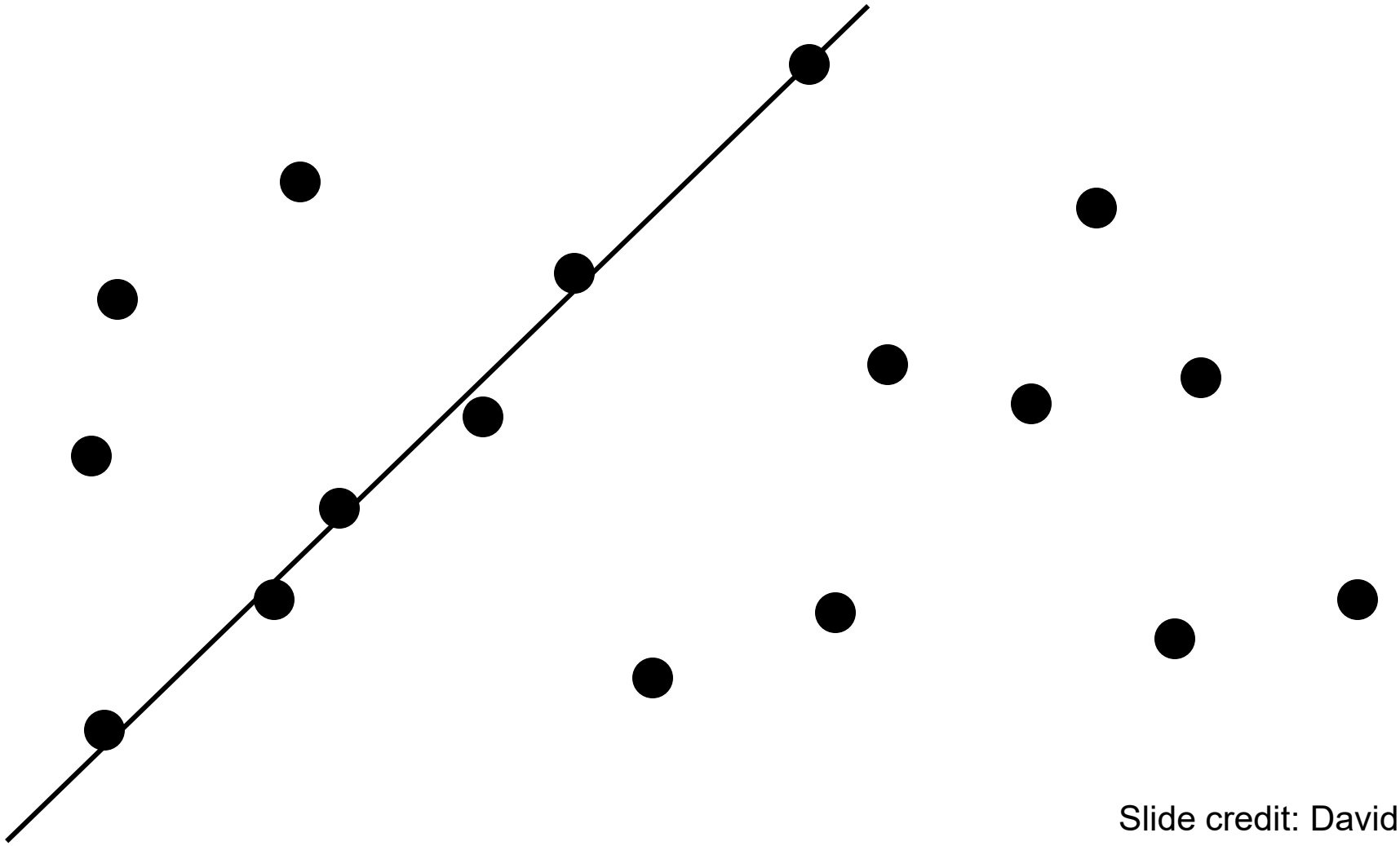
Note: Error E must be formulated carefully!



Line fitting can be max. likelihood - but choice of model is important



Line Grouping Problem



Slide credit: David Jacobs

This is difficult because of:

- Extraneous data: clutter or multiple models
  - We do not know what is part of the model?
  - Can we pull out models with a few parts from much larger amounts of background clutter?
- Missing data: only some parts of model are present
- Noise

Hough Transform

- Edges need not be connected
- Complete object need not be visible
- Key Idea: Edges VOTE for the possible model

Image and Parameter Spaces

Equation of Line: $y = mx + c$

Find: (m, c)

Consider point: (x_i, y_i)

$$y_i = mx_i + c \quad or \quad c = -x_i m + y_i$$

Parameter space also called Hough Space

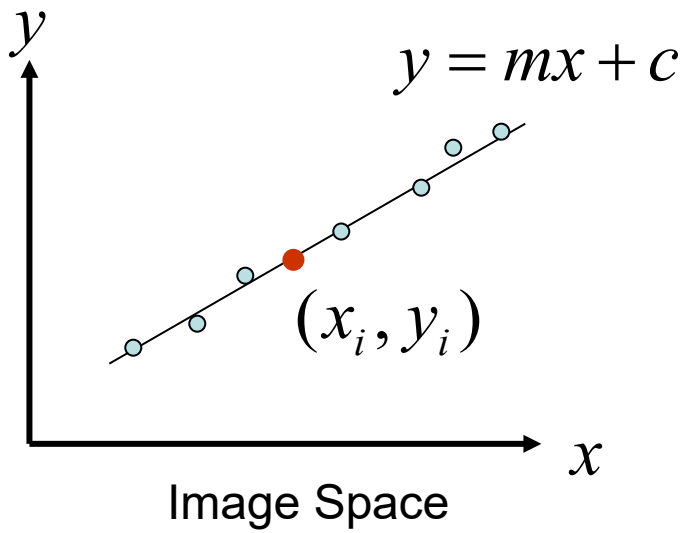
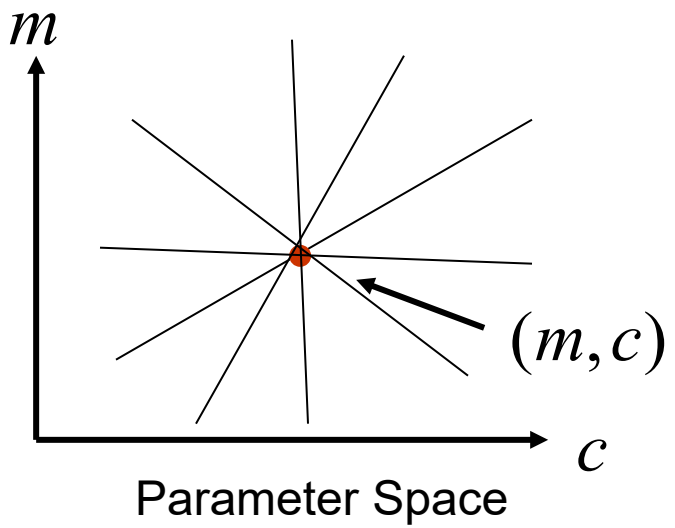


Image Space



Parameter Space

Line Detection by Hough Transform

Algorithm:

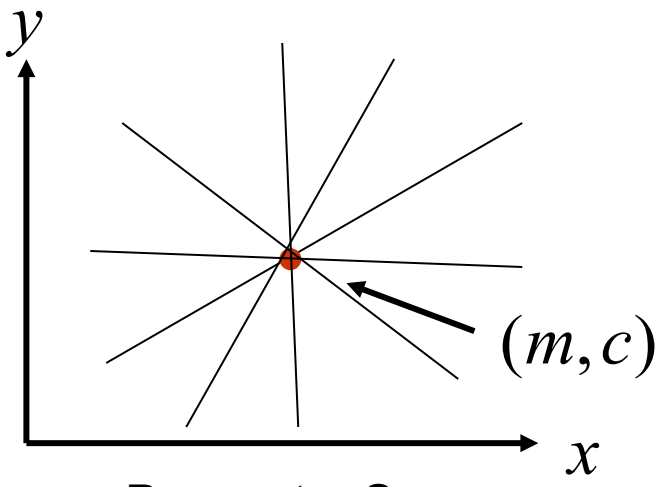
- Quantize Parameter Space (m,c)
- Create Accumulator Array $A(m,c)$
- Set $A(m,c)=0 \quad \forall m,c$
- For each image edge (x_i, y_i) increment:

$$A(m,c) = A(m,c)+1$$

- If (m,c) lies on the line:

$$c = -x_i m + y_i$$

- Find local maxima in $A(m,c)$



Parameter Space

$A(m,c)$

	1						1		
		1				1			
			1		1				
				2					
			1		1				
		1				1			
	1						1		

Better Parameterization

NOTE: $-\infty \le m \le \infty$

Large Accumulator

More memory and computations

Improvement: (Finite Accumulator Array Size)

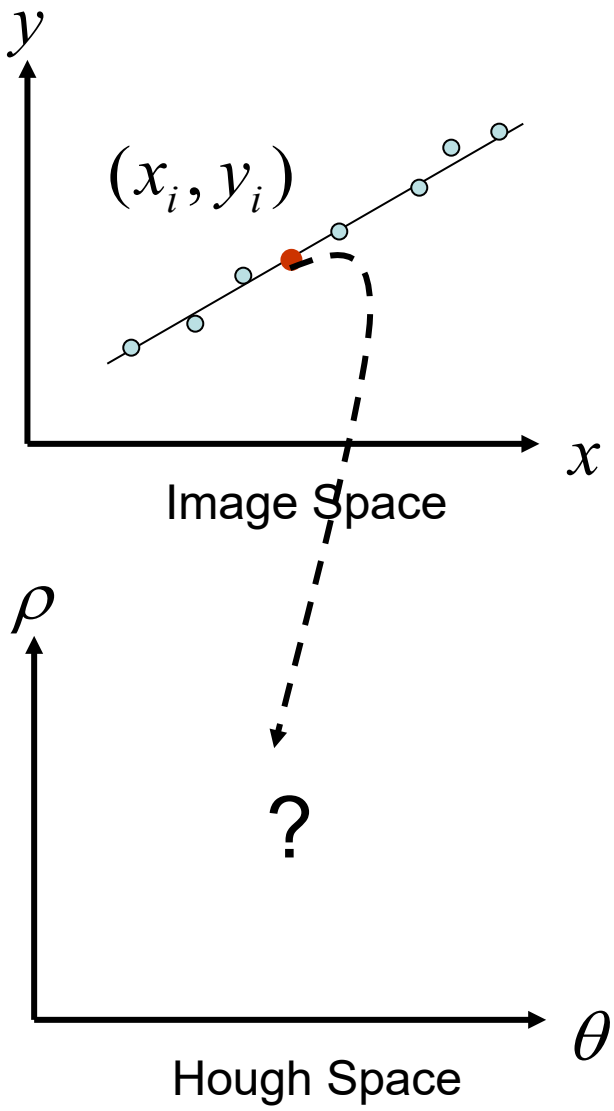
Line equation: $\rho = x\cos\theta + y\sin\theta$

Here $0 \le \theta \le 2\pi$

$0 \le \rho \le \rho_{\max}$

Given points (x_i, y_i) find (ρ, θ)

Hough Space Sinusoid



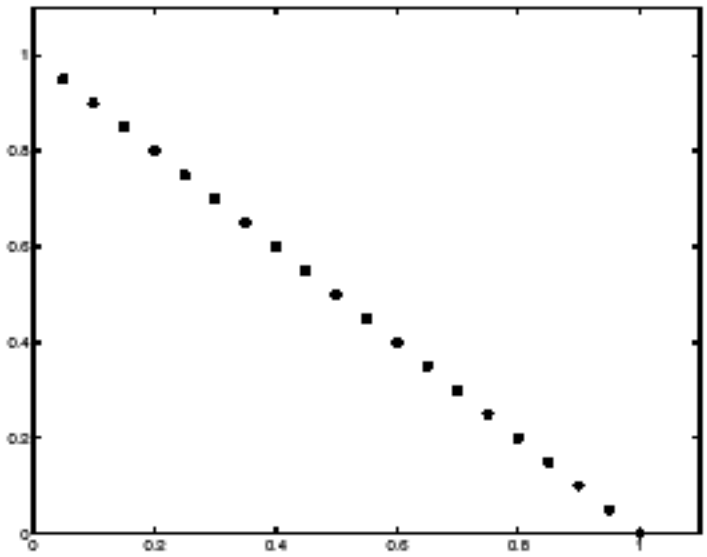
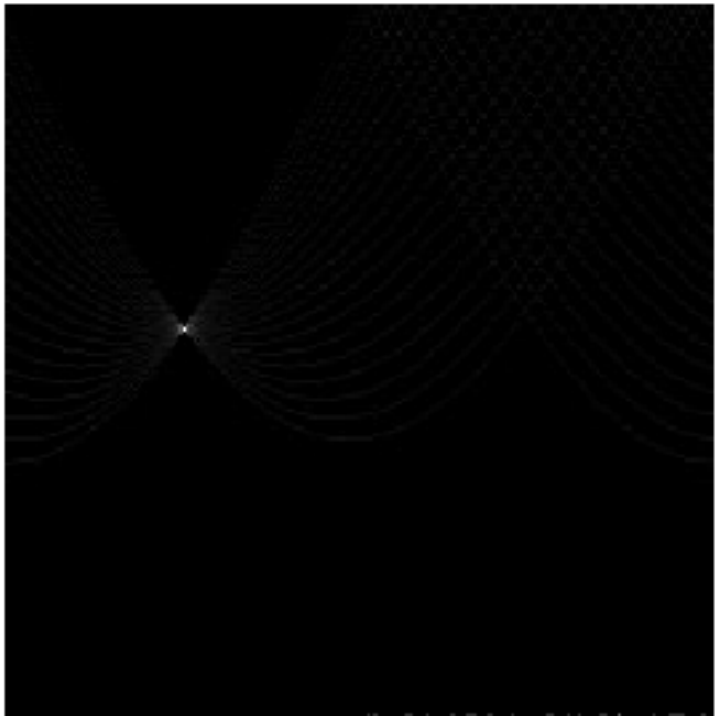
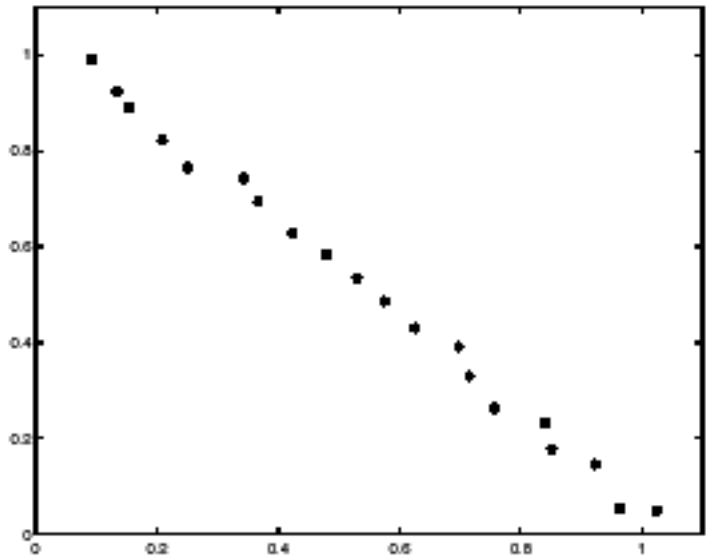


Image space



Votes

Horizontal axis is θ,
vertical is rho.



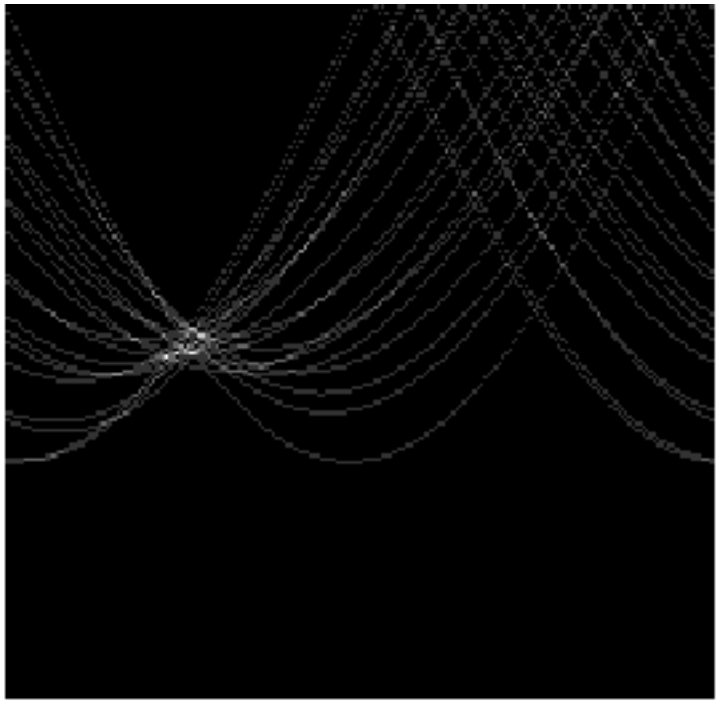
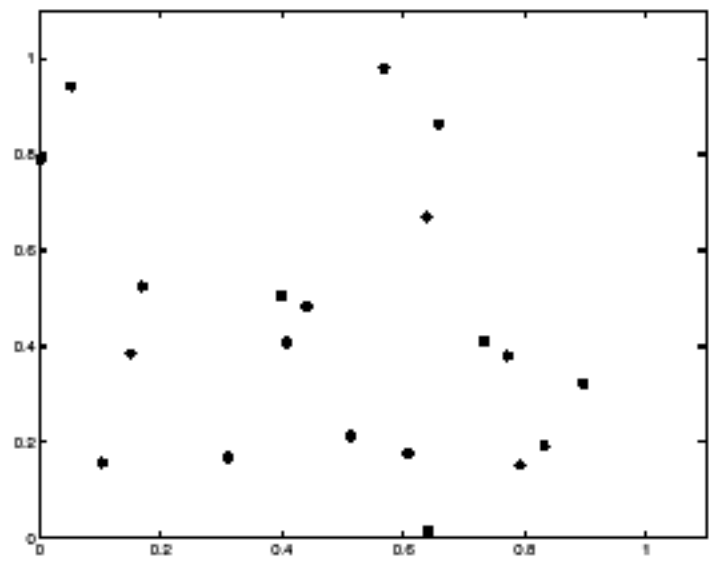
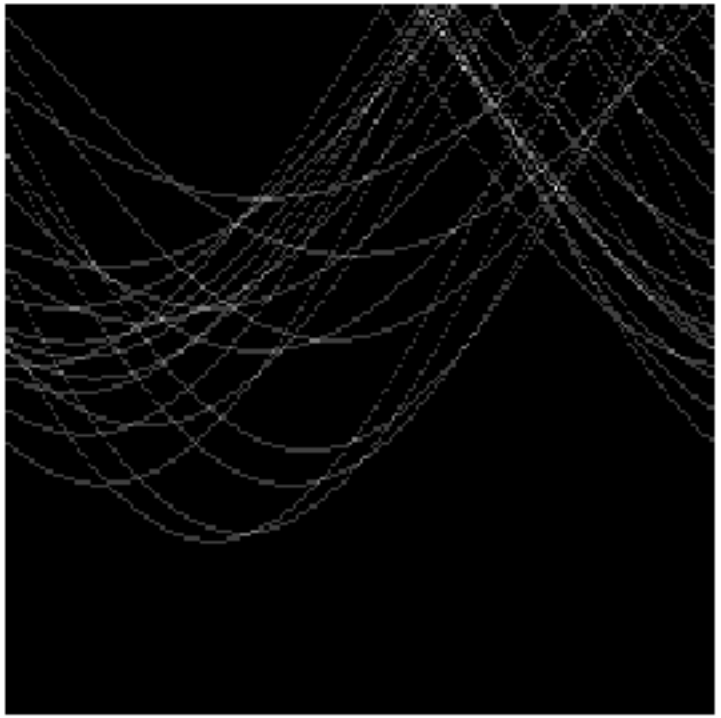


Image space

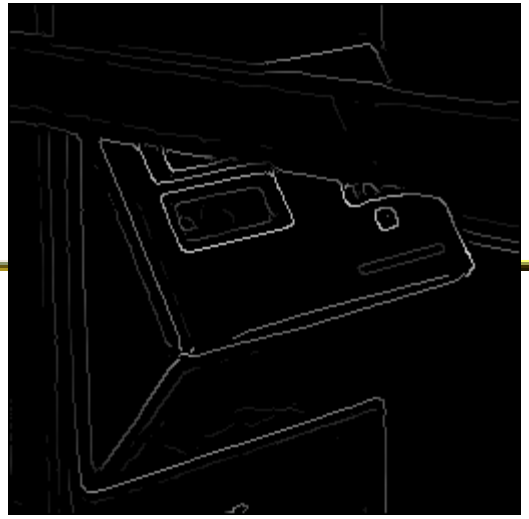
votes


0
0.2
0.4
0.6
0.8
1
0
0.2
0.4
0.6
0.8
1

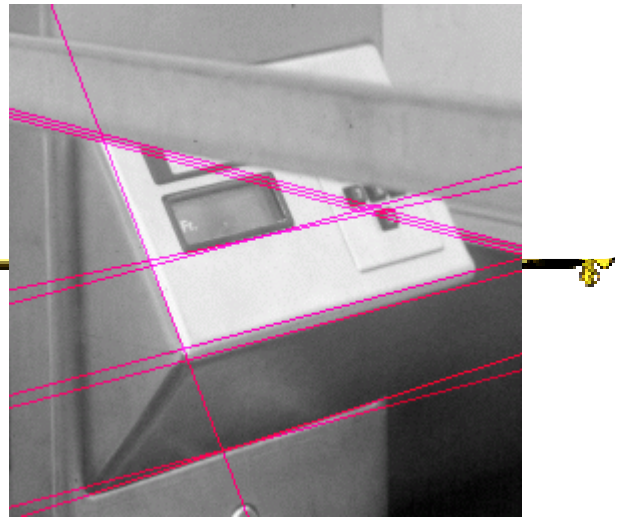


Real World Example

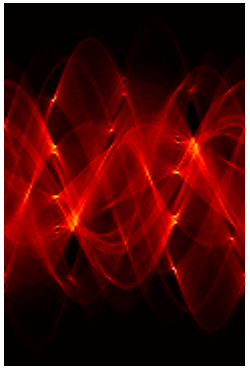
Original



Edge Detection



Found Lines

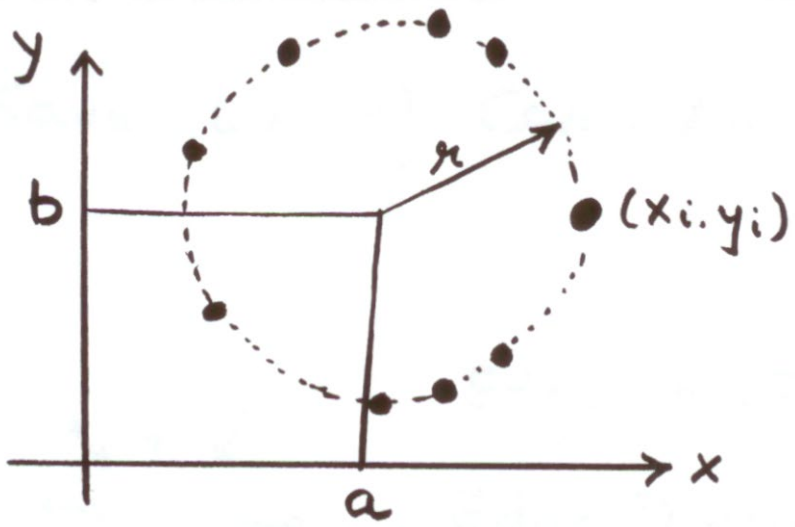


Parameter Space

Finding Circles by Hough Transform

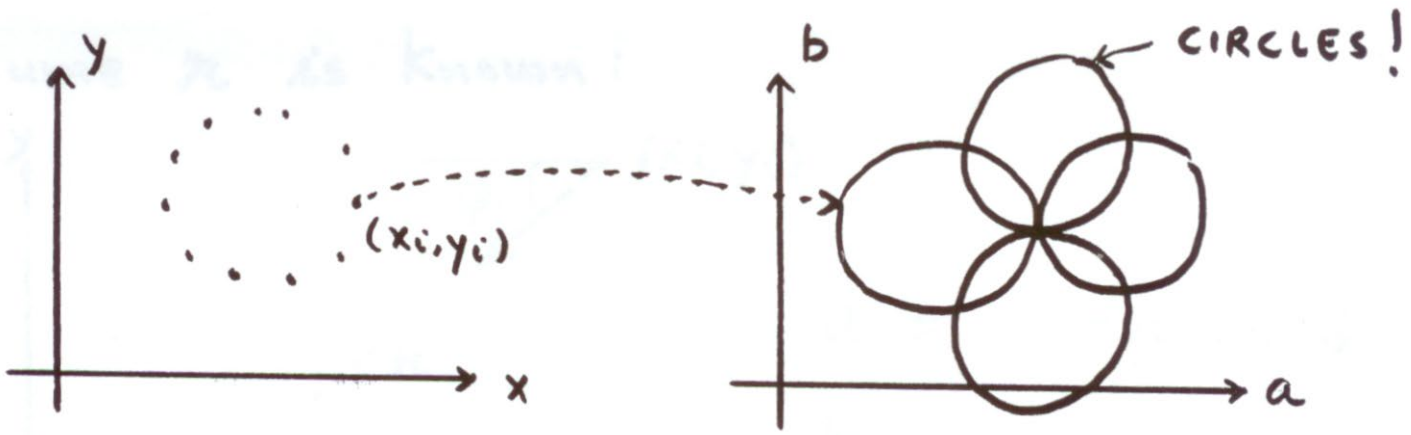
Equation of Circle:

$$(x_i - a)^2 + (y_i - b)^2 = r^2$$



If radius is known: (2D Hough Space)

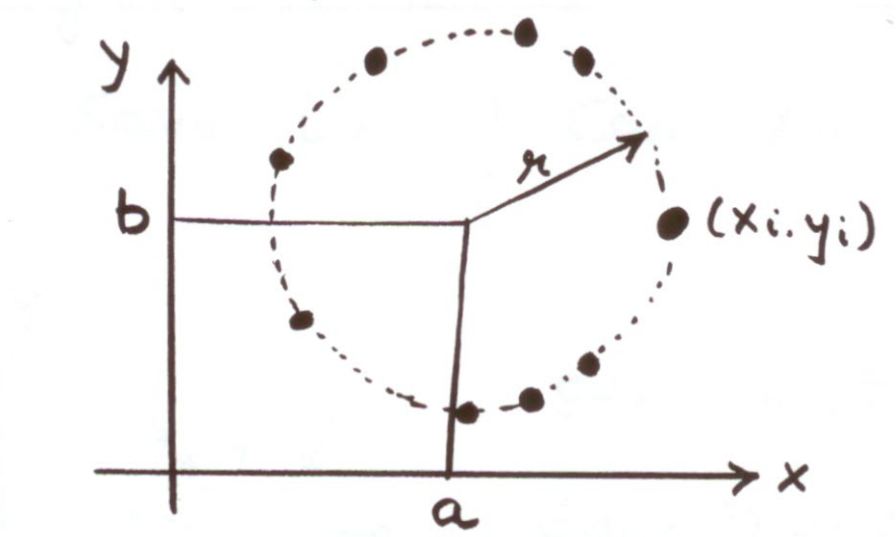
Accumulator Array $A(a, b)$



Finding Circles by Hough Transform

Equation of Circle:

$$(x_i - a)^2 + (y_i - b)^2 = r^2$$



If radius is not known: 3D Hough Space!

Use Accumulator array $A(a, b, r)$

What is the surface in the hough space?

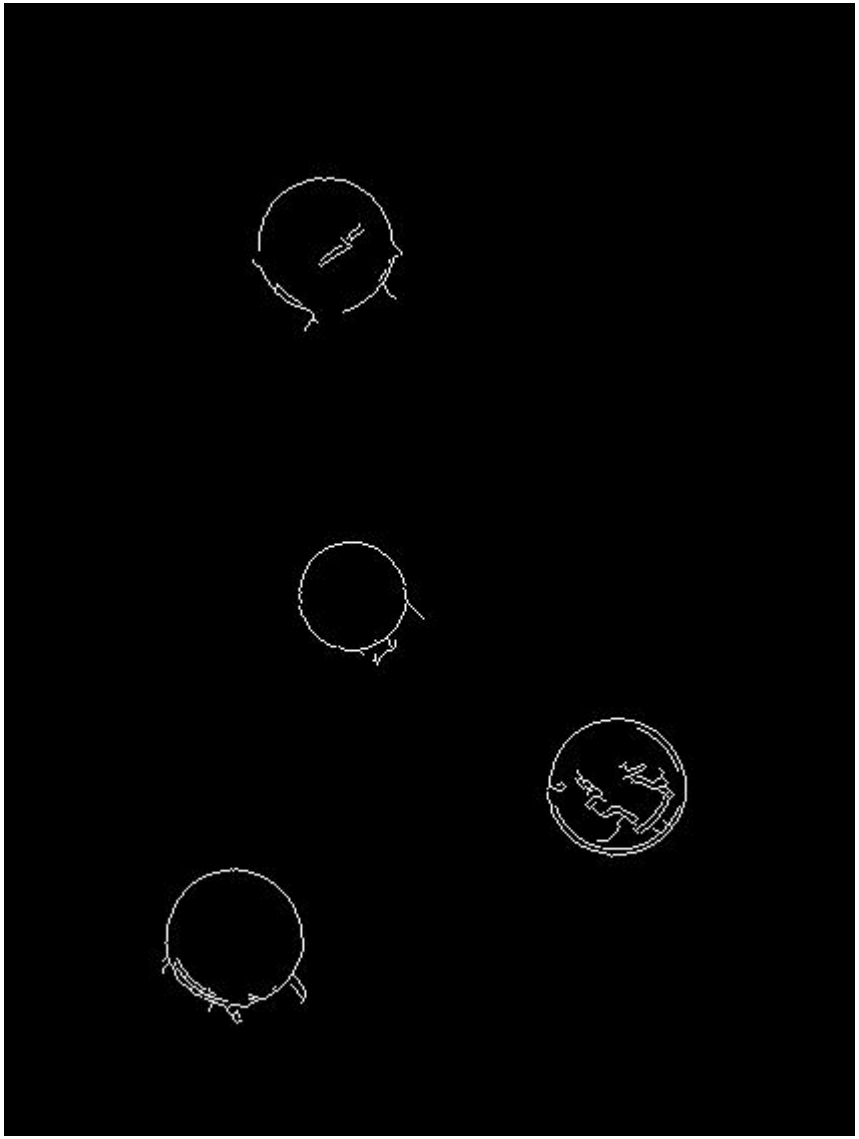
Real World Circle Examples

Crosshair indicates results of Hough transform, bounding box found via motion differencing.

Finding Coins

Original

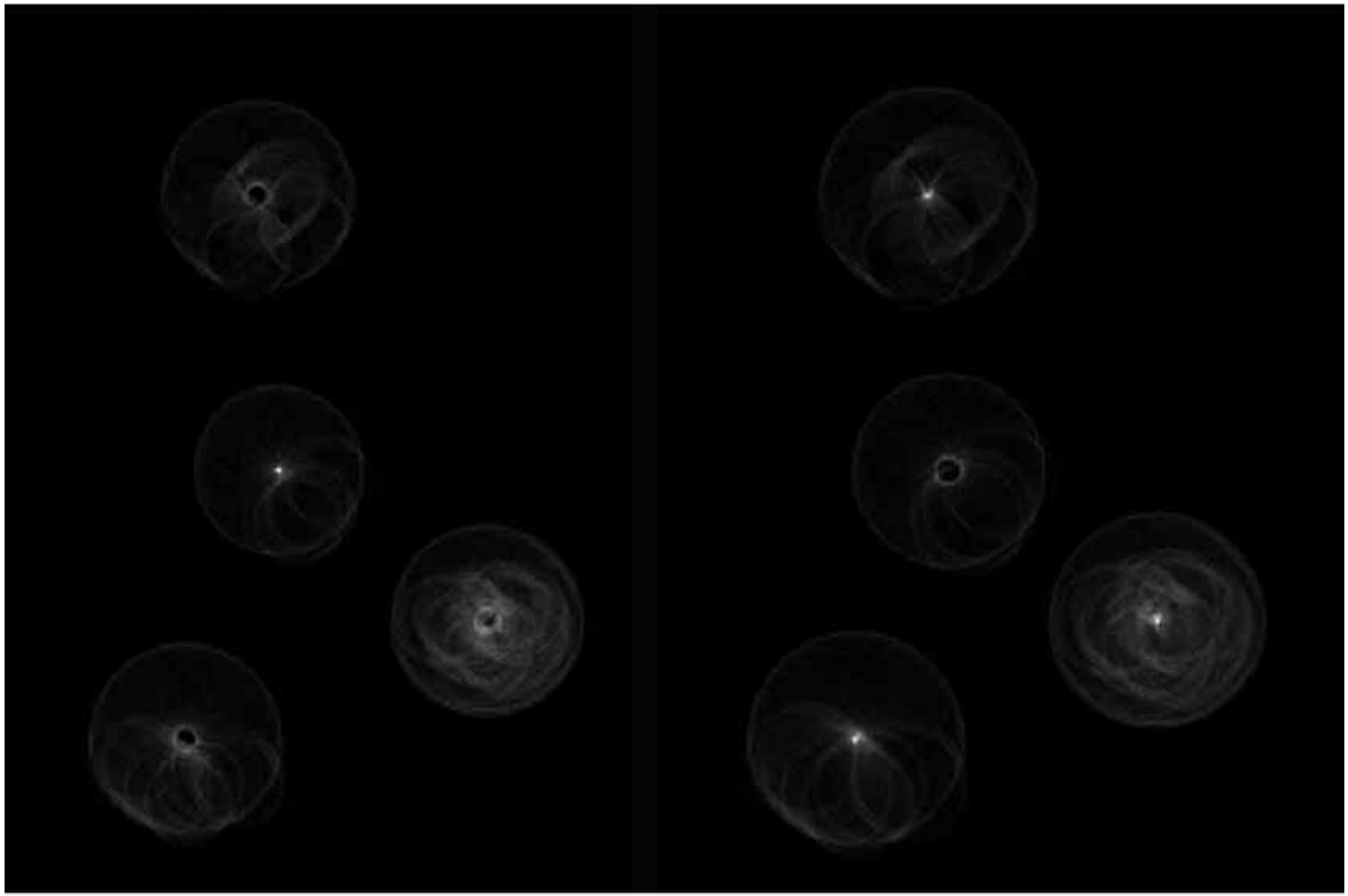
Edges (note noise)



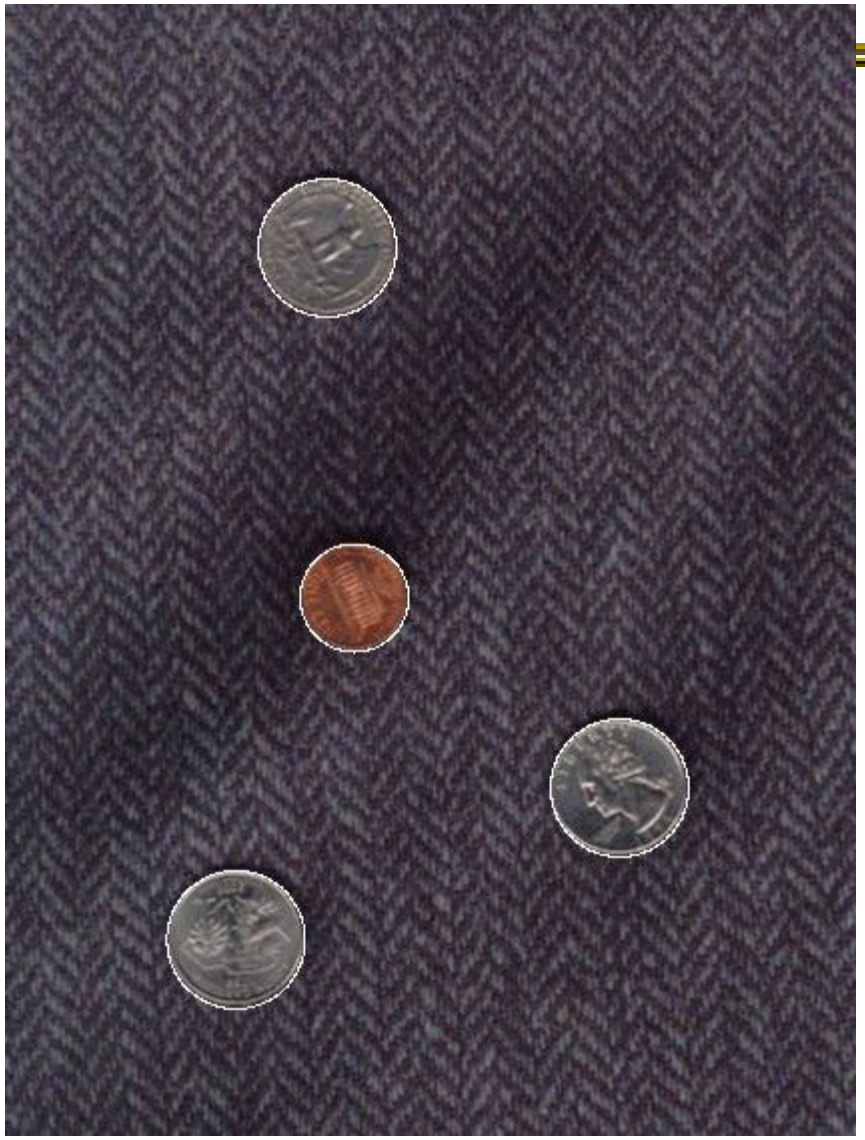
Finding Coins (Continued)

Penn

Quarters



Finding Coins (Continued)

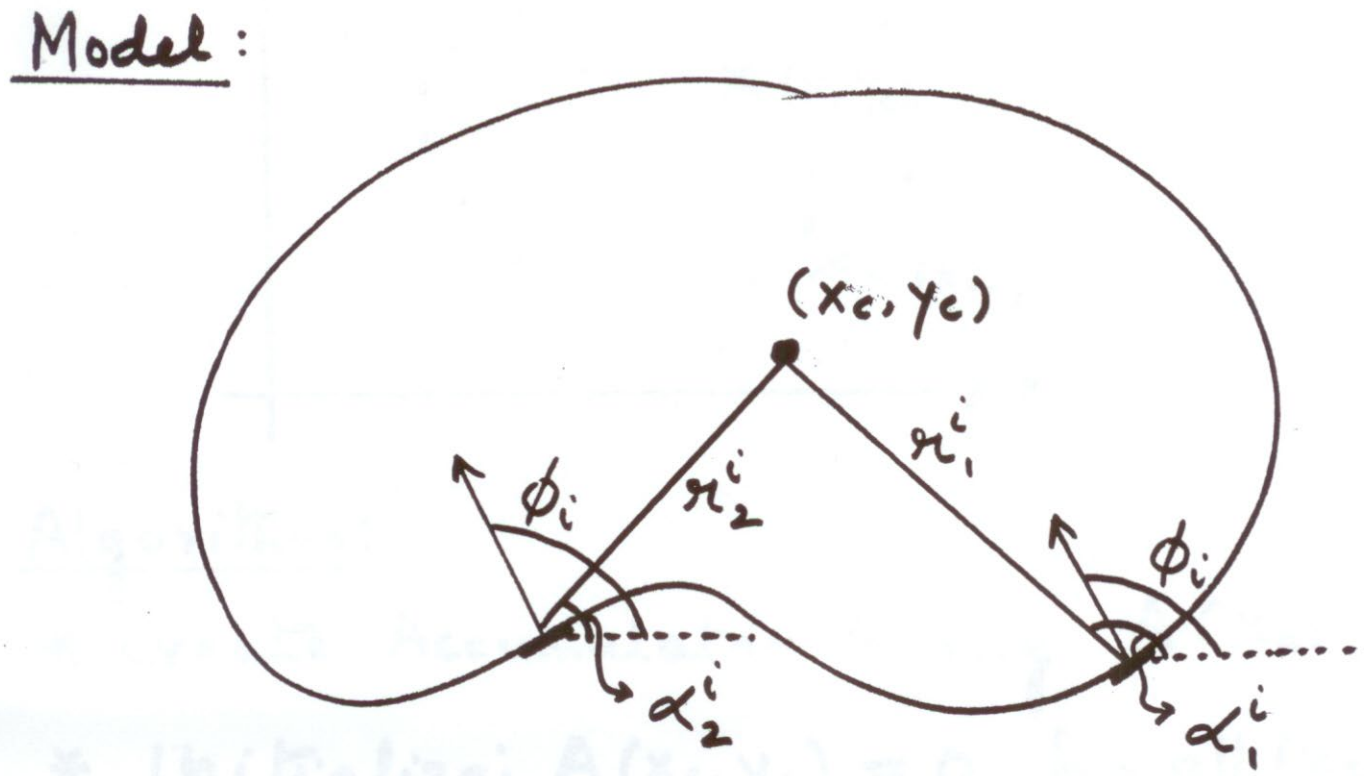


Note that because the quarters and penny are different sizes, a different Hough transform (with separate accumulators) was used for each circle size.

Coin finding sample images from: Vivek Kwatra

Generalized Hough Transform

- Model Shape NOT described by equation



Generalized Hough Transform

- Model Shape NOT described by equation

ϕ - Table

Edge Direction	$\bar{r} = (r, \alpha)$
ϕ_1	$\bar{r}_1^1, \bar{r}_2^1, \bar{r}_3^1$
ϕ_2	$\bar{r}_1^2, \bar{r}_2^2$
ϕ_i	$\bar{r}_1^i \vdots \bar{r}_2^i$
ϕ_n	$\bar{r}_1^n, \bar{r}_2^n$

Generalized Hough Transform

Find Object Center (x_c, y_c) given edges (x_i, y_i, ϕ_i)

Create Accumulator Array $A(x_c, y_c)$

Initialize: $A(x_c, y_c) = 0 \quad \forall (x_c, y_c)$

For each edge point (x_i, y_i, ϕ_i)

For each entry $\overline{r}_k^i$ in table, compute:

$$x_c = x_i + r_k^i \cos \alpha_k^i$$

$$y_c = y_i + r_k^i \sin \alpha_k^i$$

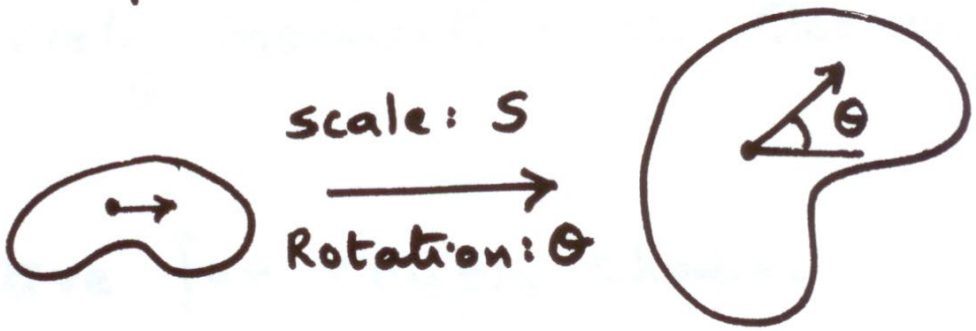
Increment Accumulator: $A(x_c, y_c) = A(x_c, y_c) + 1$

Find Local Maxima in $A(x_c, y_c)$

Scale & Rotation:

Use Accumulator Array:

$$A[x_c, y_c, S, \theta]$$



Use:

$$x_c = x_i + r^i_k S \cos(\alpha^i_k + \theta)$$

$$y_c = y_i + r^i_k S \sin(\alpha^i_k + \theta)$$

$$A(x_c, y_c, S, \theta) = A(x_c, y_c, S, \theta) + 1 .$$

Hough Transform: Comments

- Works on Disconnected Edges
- Relatively insensitive to occlusion
- Effective for simple shapes (lines, circles, etc)
- Trade-off between work in Image Space and Parameter Space
- Handling inaccurate edge locations:
    - Increment Patch in Accumulator rather than a single point