Feature Detection

- Image features: Interesting/important local patterns
- Detecting features can be an important step in localizing or recognizing objects in the image (feature-based methods)
- Example features
  - Edges
  - Lines, curves
  - Application-specific patterns
- Outline
  - Edge detection
  - Detection of lines/curves

Edge Detection

- Idea (continuous-space): Detect local gradient

$$|\nabla f(x,y)|_2 = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

- Digital image:
  Use finite differences instead

difference $\begin{bmatrix} -1 & 1 \end{bmatrix}$

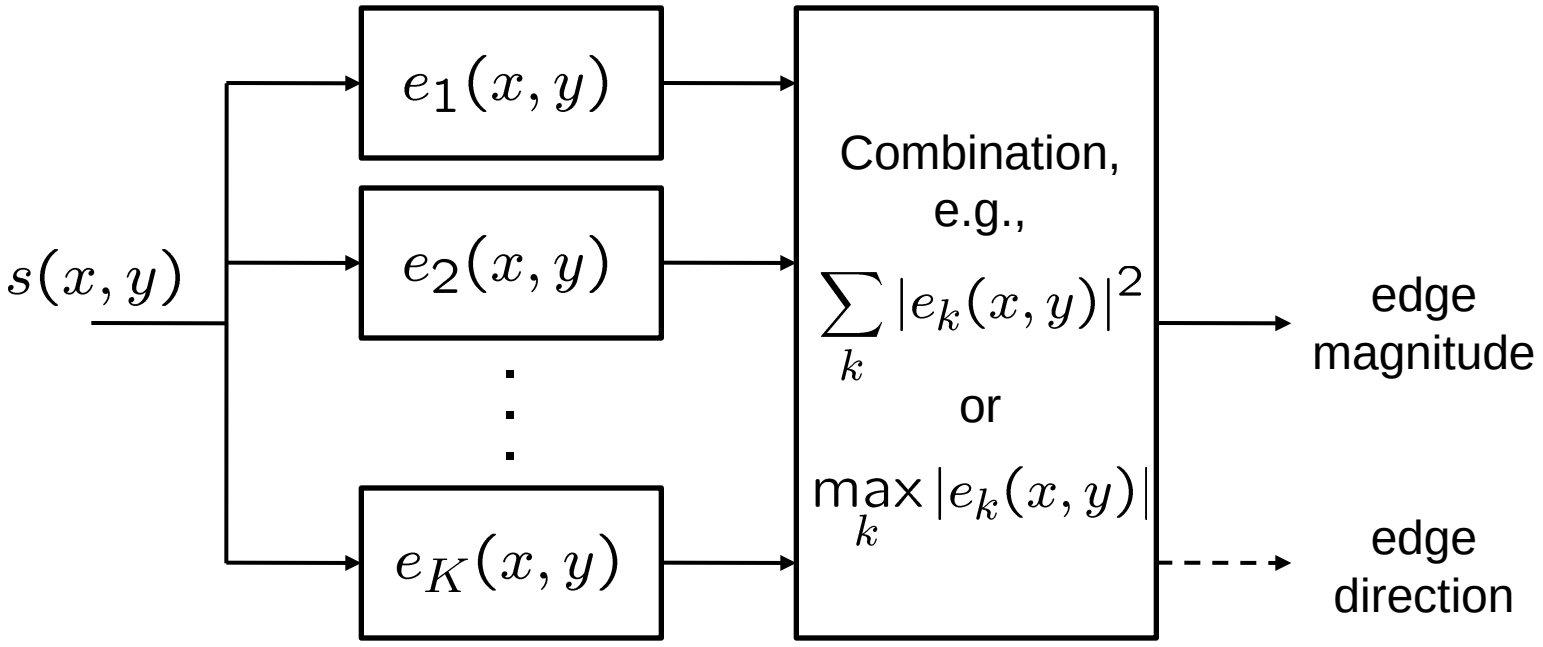
central difference $\begin{bmatrix} -1 & [0] & 1 \end{bmatrix}$

Prewitt $\begin{bmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{bmatrix}$

Sobel $\begin{bmatrix} -1 & 0 & 1 \\ -2 & [0] & 2 \\ -1 & 0 & 1 \end{bmatrix}$

Practical Edge Detectors

- Edges can have any orientation
- Typical edge detection scheme uses K=2 edge templates
- Some use K>2



Edge Detection Filters

Roberts

$$\begin{bmatrix} [0] & 1 \\ -1 & 0 \end{bmatrix} \quad \begin{bmatrix} [1] & 0 \\ 0 & -1 \end{bmatrix}$$

Prewitt

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ 0 & [0] & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Sobel

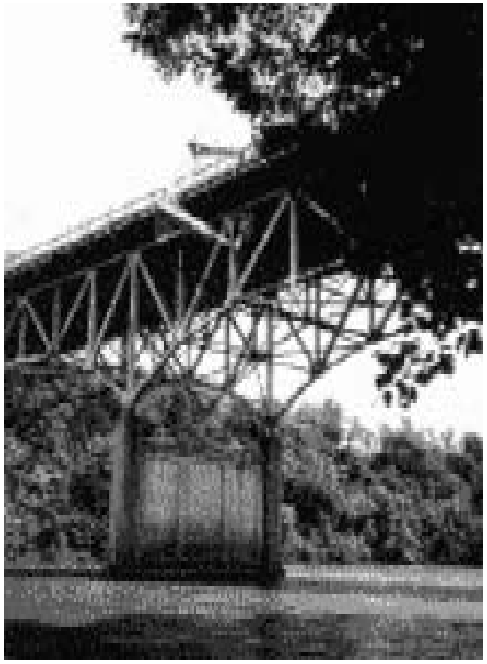
$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & [0] & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & -2 & -1 \\ 0 & [0] & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Kirsch

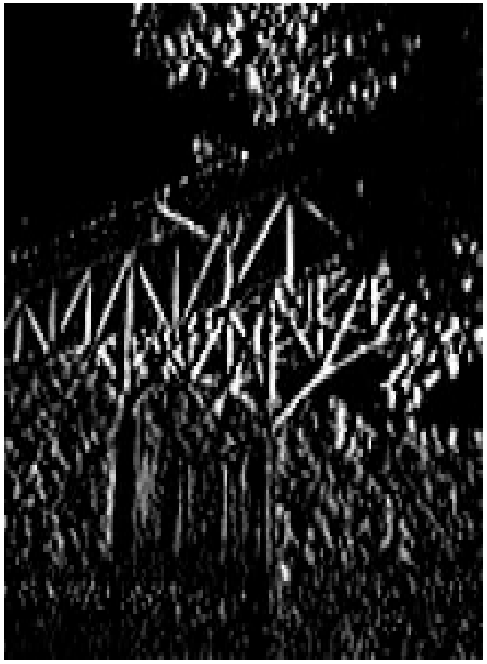
$$\begin{bmatrix} +5 & +5 & +5 \\ -3 & [0] & -3 \\ -3 & -3 & -3 \end{bmatrix} \quad \begin{bmatrix} -3 & +5 & +5 \\ -3 & [0] & +5 \\ -3 & -3 & -3 \end{bmatrix} \quad \begin{bmatrix} -3 & -3 & +5 \\ -3 & [0] & +5 \\ -3 & -3 & +5 \end{bmatrix} \quad \begin{bmatrix} -3 & -3 & -3 \\ -3 & [0] & +5 \\ -3 & +5 & +5 \end{bmatrix}$$

$$\begin{bmatrix} -3 & -3 & -3 \\ -3 & [0] & -3 \\ +5 & +5 & +5 \end{bmatrix} \quad \begin{bmatrix} -3 & -3 & -3 \\ +5 & [0] & -3 \\ +5 & +5 & -3 \end{bmatrix} \quad \begin{bmatrix} +5 & -3 & -3 \\ +5 & [0] & -3 \\ +5 & -3 & -3 \end{bmatrix} \quad \begin{bmatrix} +5 & +5 & -3 \\ +5 & [0] & -3 \\ -3 & -3 & -3 \end{bmatrix}$$

Prewitt Operator Example

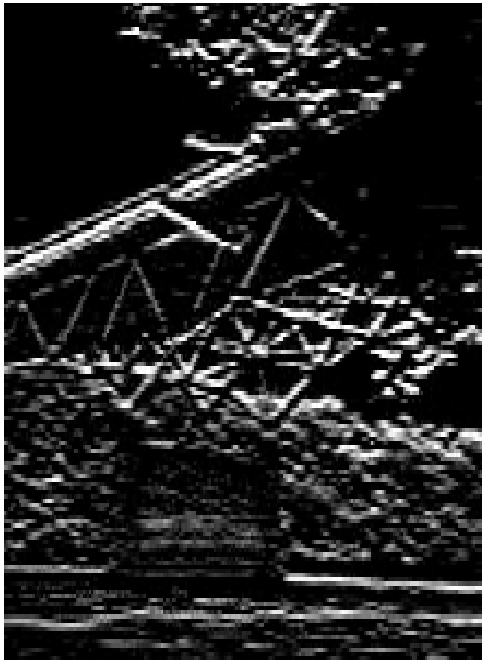


original
Bridge



magnitude of
image filtered with

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{bmatrix}$$



magnitude of
image filtered with

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & [0] & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Prewitt Operator Example

original
Billsface

log magnitude of
image filtered with

$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

log magnitude of
image filtered with

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & [0] & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Prewitt Operator Example

log sum of squared horizontal and vertical gradients

different thresholds

Sobel Operator Example

log sum of squared horizontal and vertical gradients

different thresholds

Roberts Operator Example

original
Billsface

log magnitude of
image filtered with

$$\begin{bmatrix} [1] & 0 \\ 0 & -1 \end{bmatrix}$$

log magnitude of
image filtered with

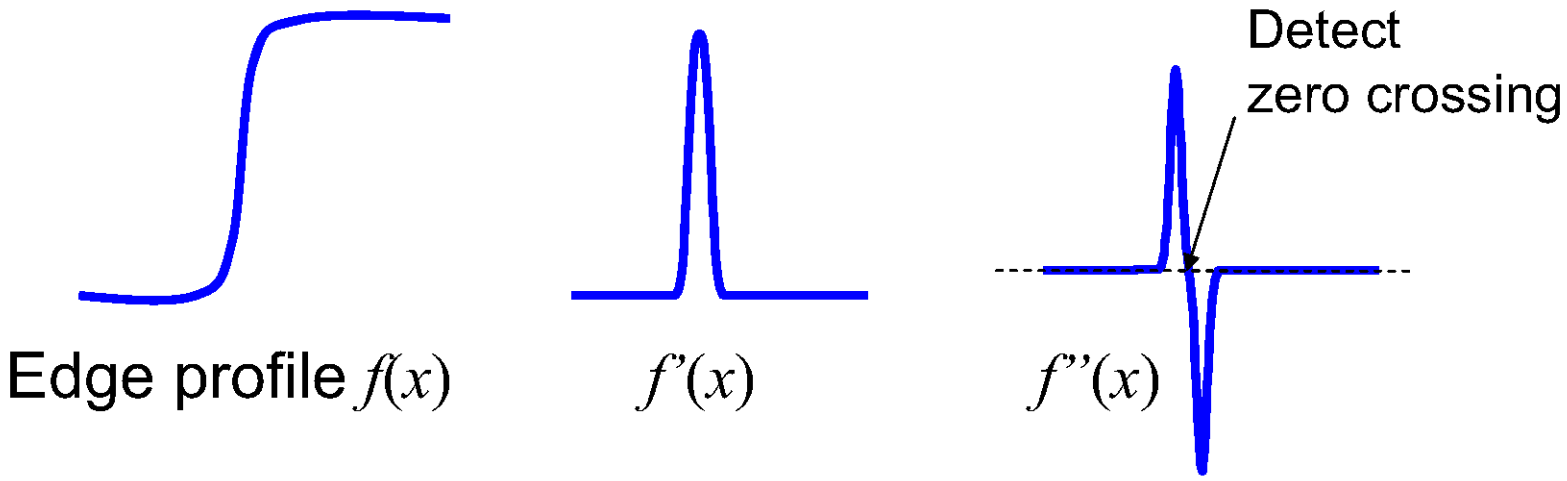
$$\begin{bmatrix} [0] & 1 \\ -1 & 0 \end{bmatrix}$$

Roberts Operator Example

log sum of squared horizontal and vertical gradients

different thresholds

1-D Illustration of 2nd derivative Edge Detector



Laplacian Operator

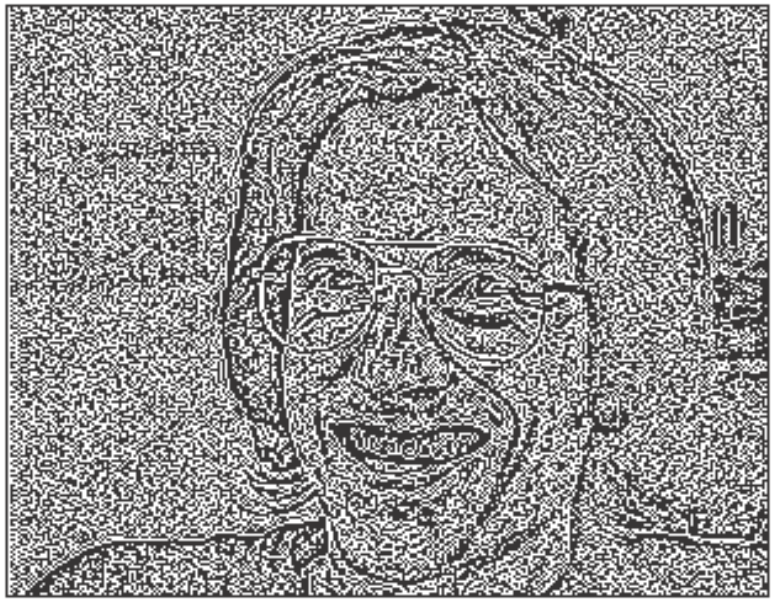
- Detect discontinuities by considering second derivative

$$\nabla^2 f(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

- Isotropic (rotationally invariant) operator
- Zero-crossings mark edge location
- Discrete-space approximation by convolution with 3x3 impulse response

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & [-4] & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{or} \quad \begin{bmatrix} 1 & 1 & 1 \\ 1 & [-8] & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Zero Crossings of Laplacian



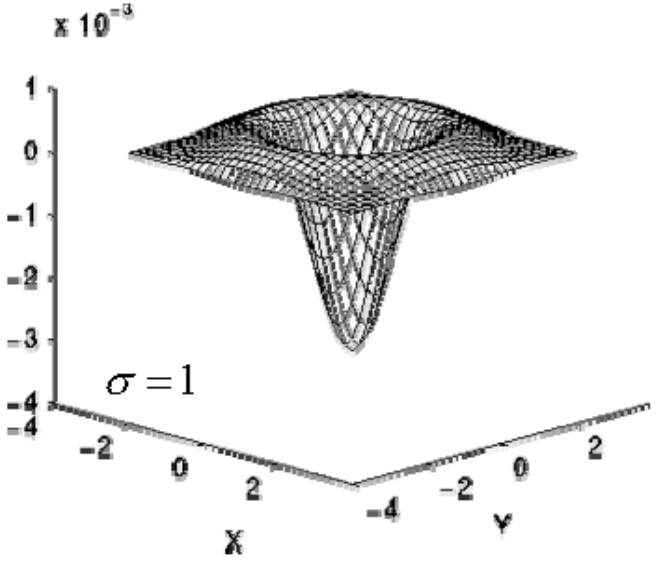
- Sensitive to very fine detail and noise $\rightarrow$ Blur image first
- Responds equally to strong and weak edges
  $\rightarrow$ Suppress edges with low gradient magnitude

Laplacian of Gaussian

- Blurring of image with Gaussian and Laplacian operator can be combined into convolution with Laplacian of Gaussian (LoG) operator

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

- Continuous function and discrete approximation

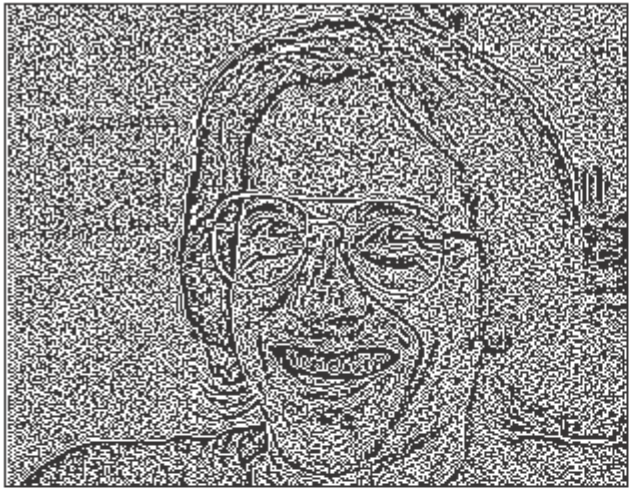


$\sigma = 1.4$

$$\begin{bmatrix} 0 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 0 \\ 1 & 2 & 4 & 5 & 5 & 5 & 4 & 2 & 1 \\ 1 & 4 & 5 & 3 & 0 & 3 & 5 & 4 & 1 \\ 2 & 5 & 3 & -12 & -24 & -12 & 3 & 5 & 2 \\ 2 & 5 & 0 & -24 & -40 & -24 & 0 & 5 & 2 \\ 2 & 5 & 3 & -12 & -24 & -12 & 3 & 5 & 2 \\ 1 & 4 & 5 & 3 & 0 & 3 & 5 & 4 & 1 \\ 1 & 2 & 4 & 5 & 5 & 5 & 4 & 2 & 1 \\ 0 & 1 & 1 & 2 & 2 & 2 & 1 & 1 & 0 \end{bmatrix}$$

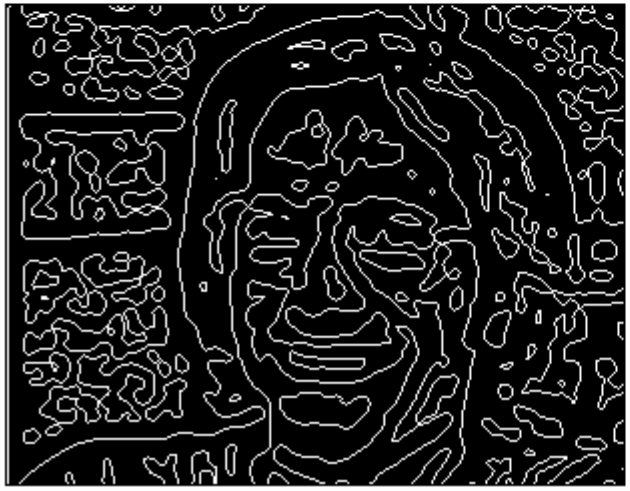
Zero Crossings of LoG

w/o Gaussian



$\sigma = 1.4$

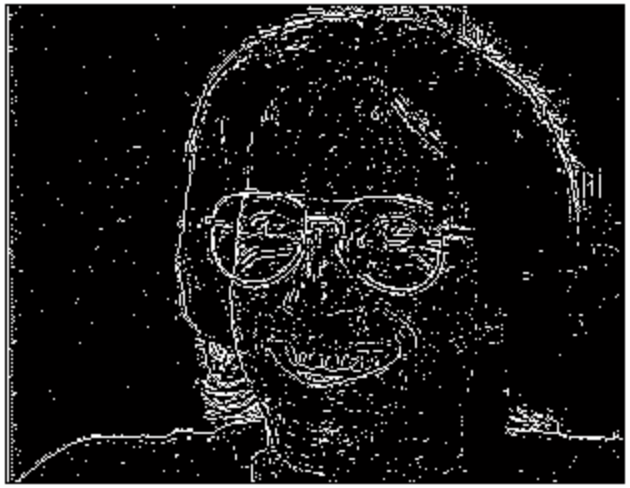
$\sigma = 3$



$\sigma = 6$

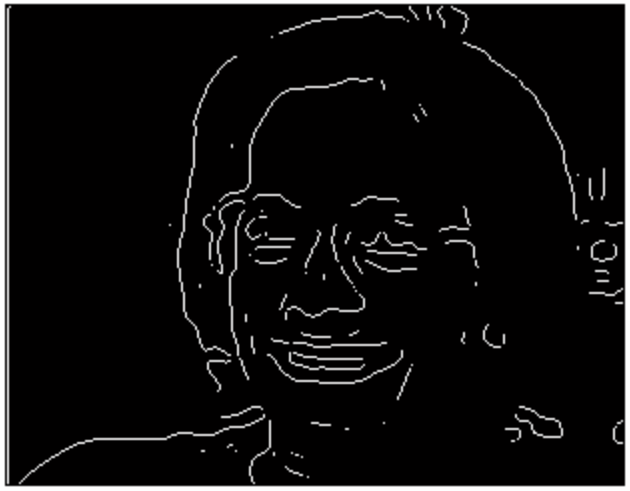
Zero Crossings of LoG – Gradient-Based Threshold

w/o Gaussian



$\sigma = 1.4$

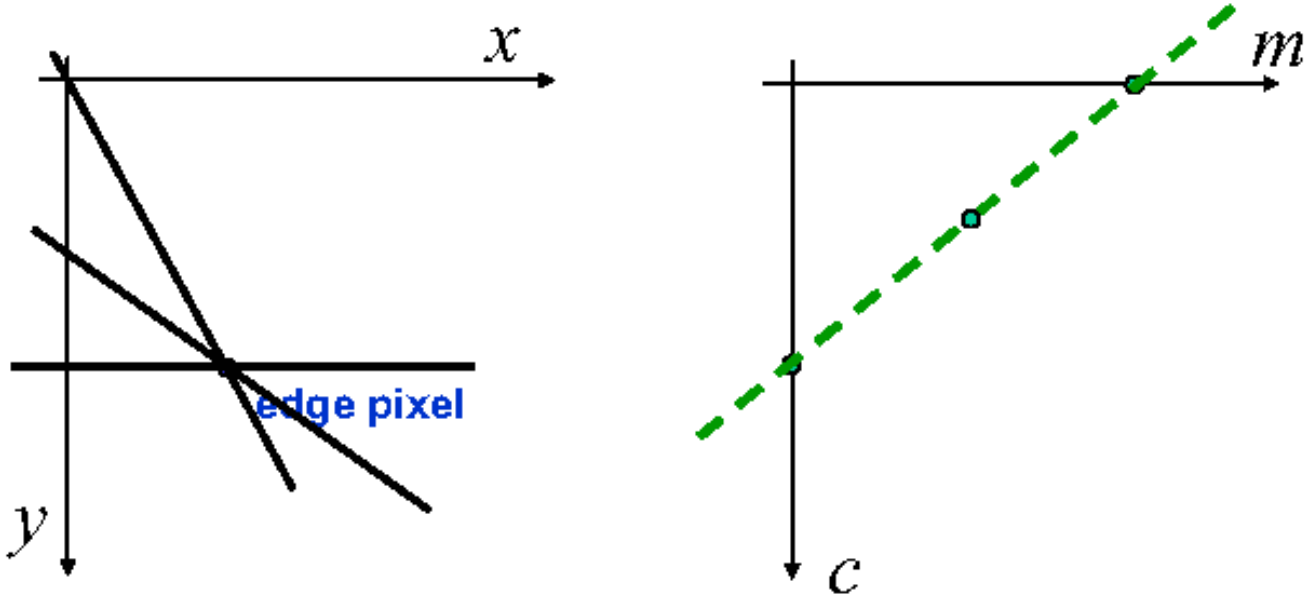
$\sigma = 3$



$\sigma = 6$

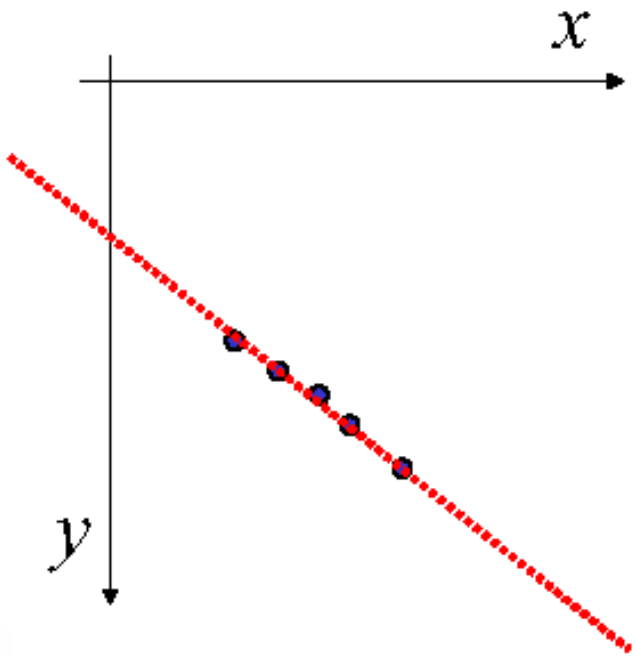
Hough Transform

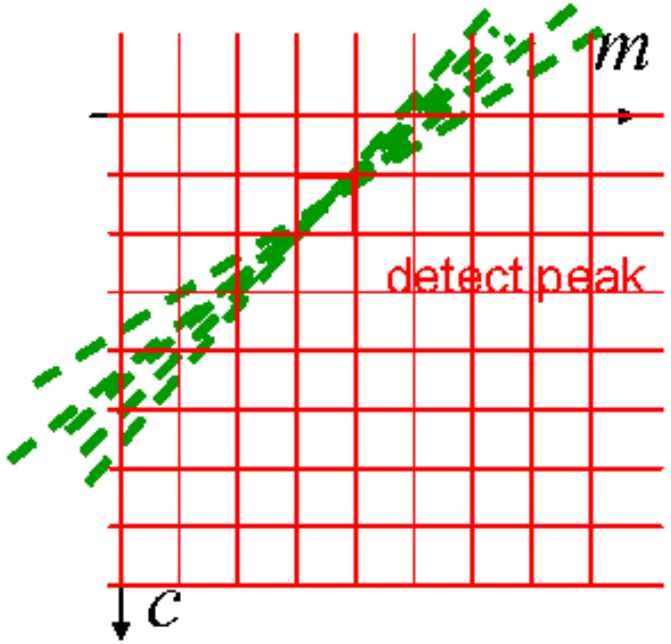
- Problem: Fit a straight line (or curve) to a set of edge pixels
- Hough transform (1962): Generalized template matching technique
- Consider detection of straight lines y = mx + c



Hough Transform

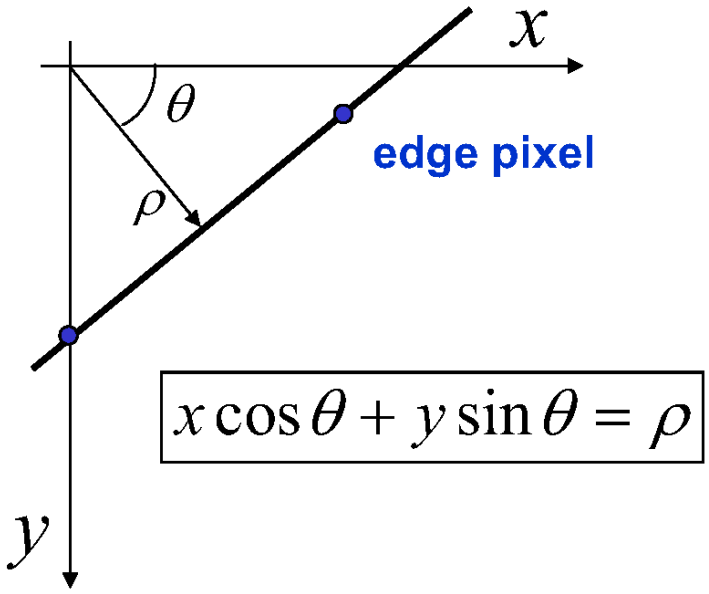
- Subdivide (m,c) plane into discrete “bins,” initialize all bin counts by 0
- Draw a line in the parameter space m,c for each edge pixel x,y and increment bin counts along line
- Detect peak(s) in (m,c) plane



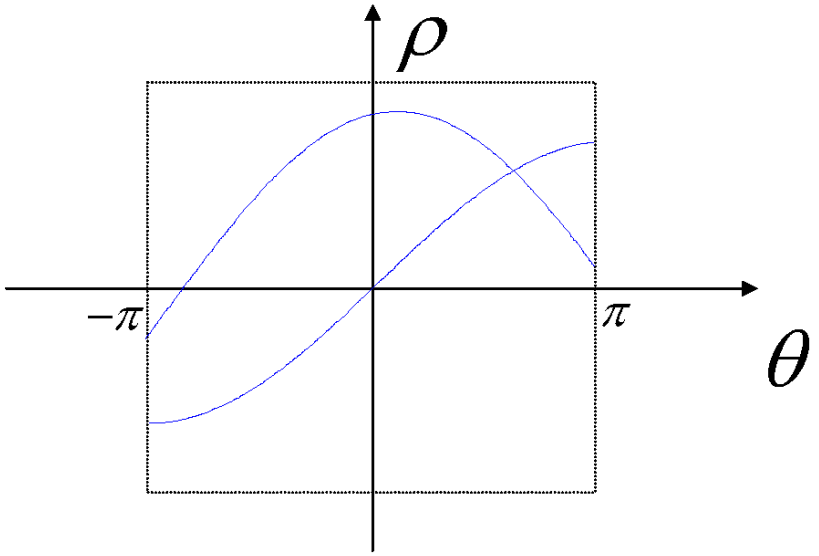


Hough Transform

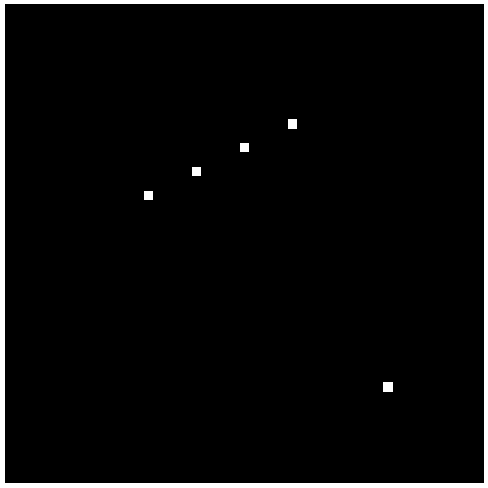
- Alternative parameterization avoids infinite-slope problem

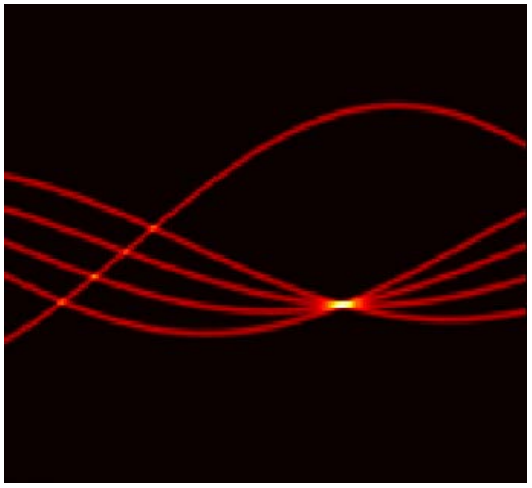


$$x\cos\theta + y\sin\theta = \rho$$

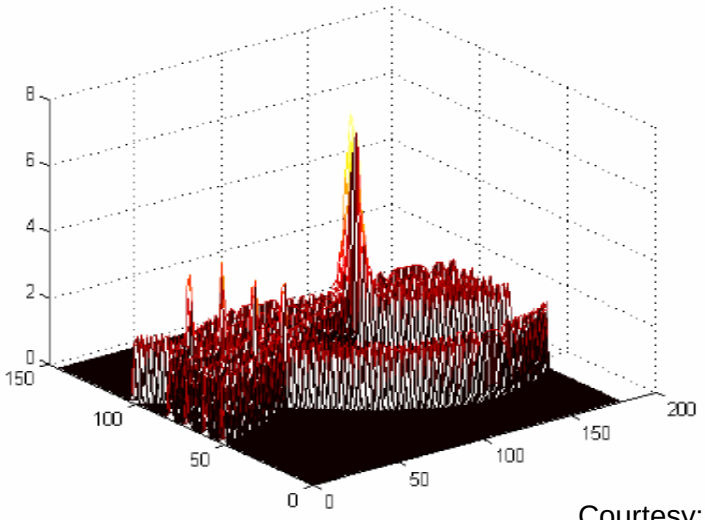


Hough Transform Example A



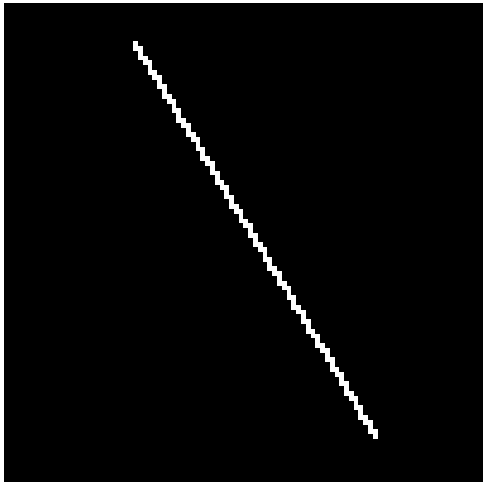


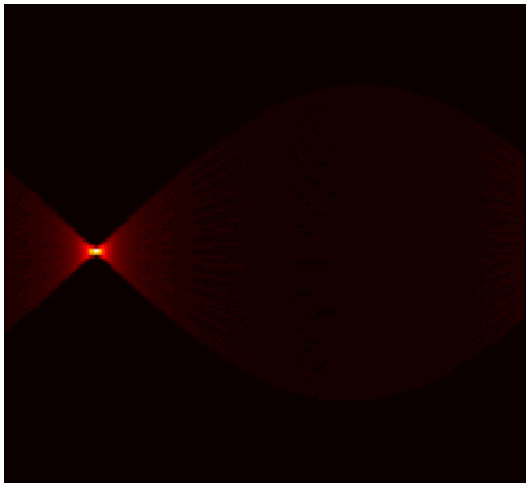
original image



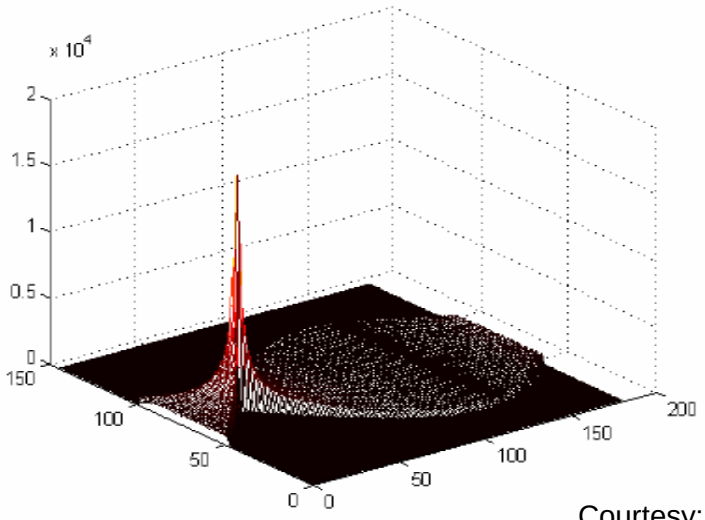
Courtesy: P. Salembier

Hough Transform Example B



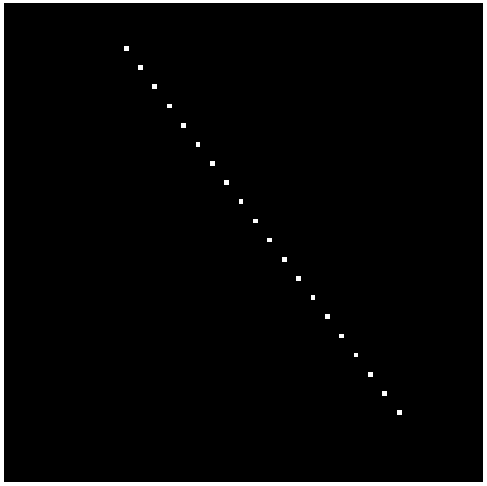


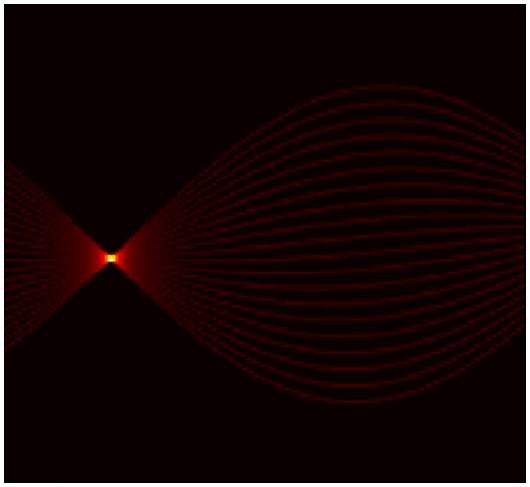
original image



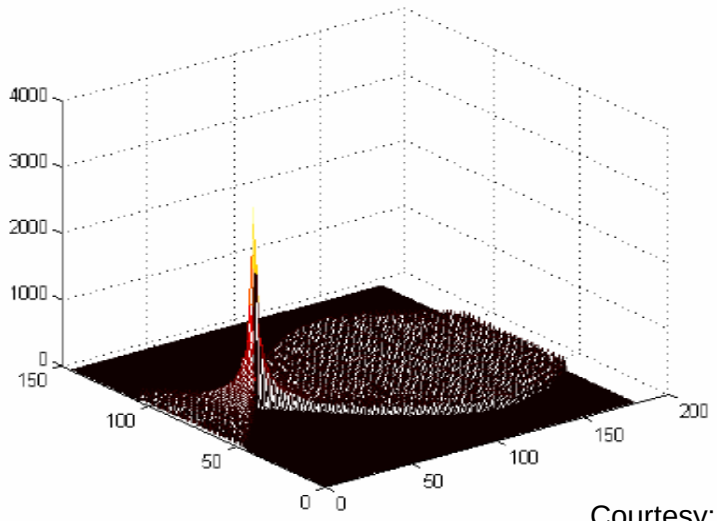
Courtesy: P. Salembier

Hough Transform Example C





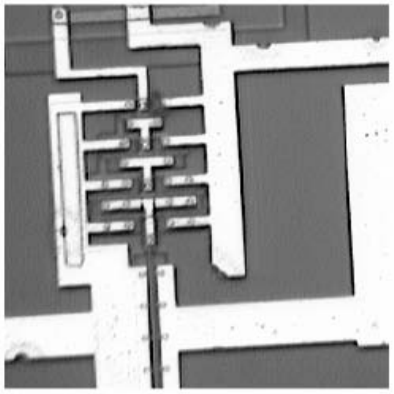
original image

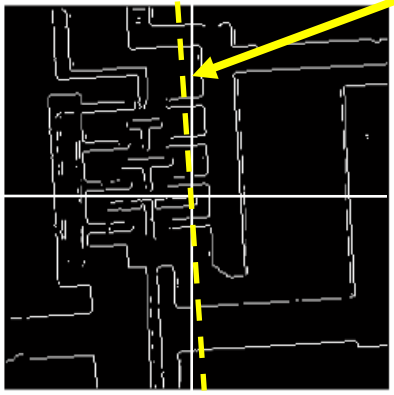


Courtesy: P. Salembier

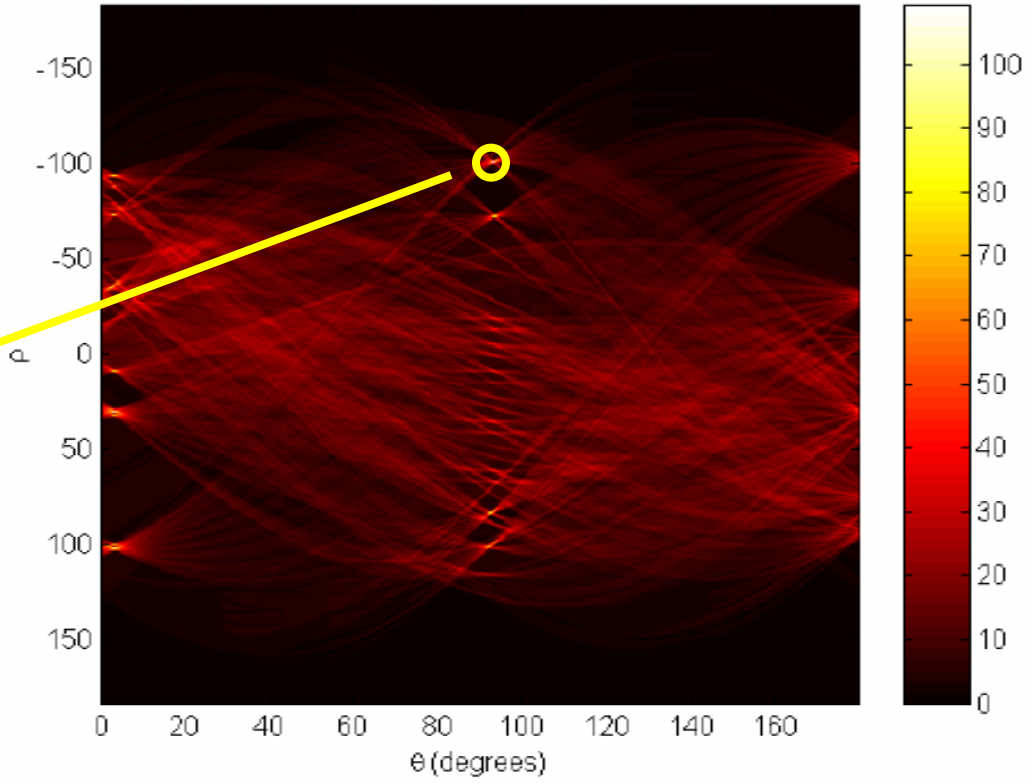
Hough Transform Example D

original IC image





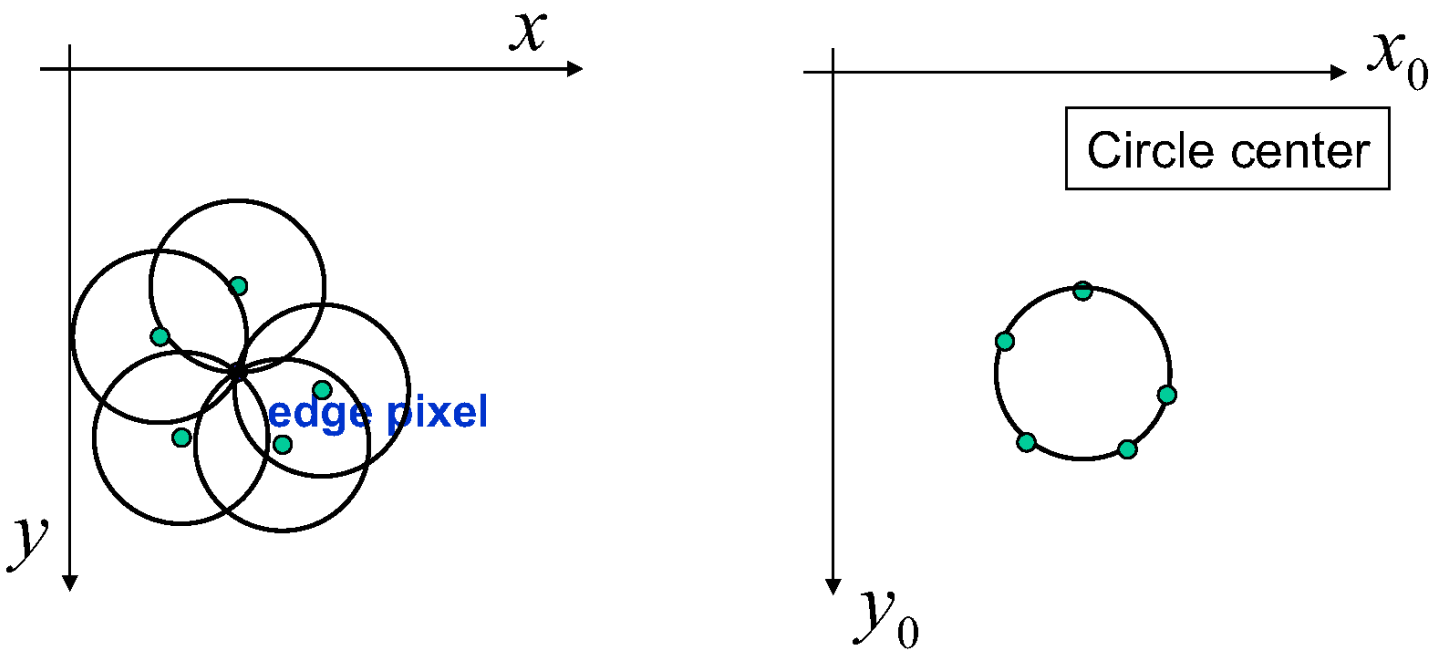
94º



edge detection (Prewitt)

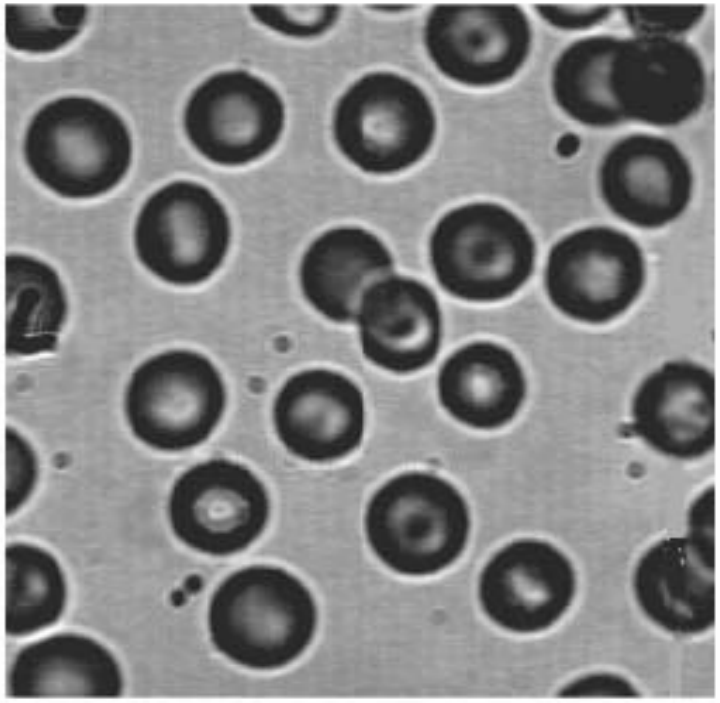
Circle Detection by Hough Transform

- Find circles of fixed radius r

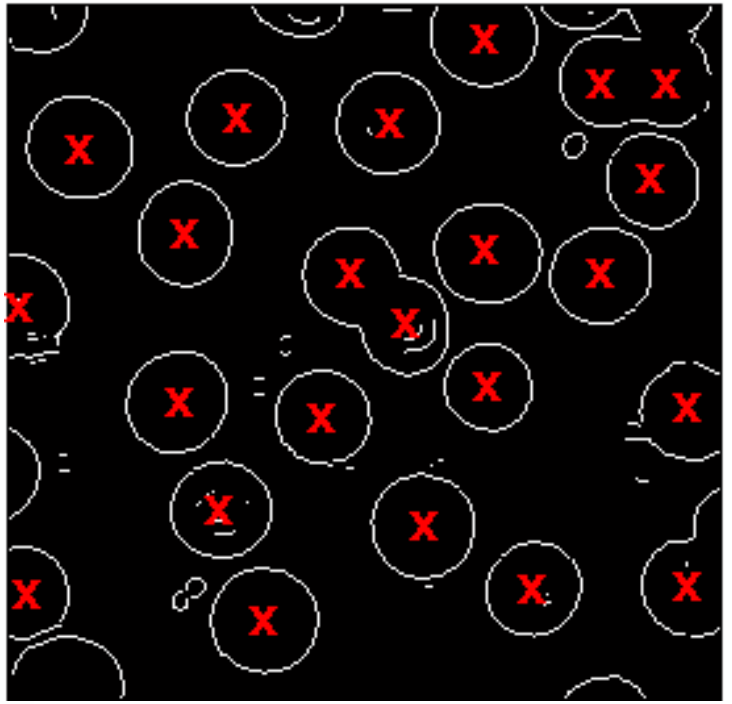


- For circles of undetermined radius, use 3-d Hough transform for parameters (x_0, y_0, r)

Example: Circle Detection by Hough Transform



original *Blood* image



Prewitt edge detection