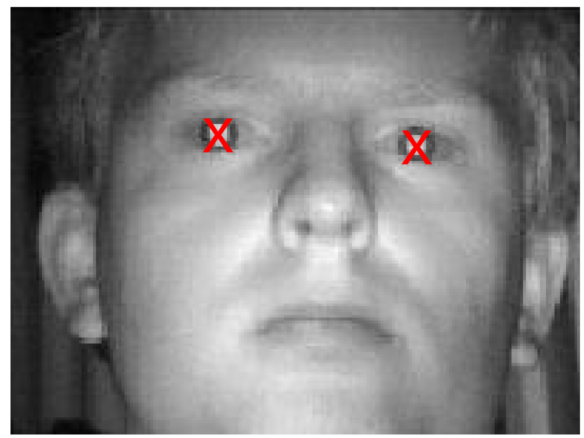
Feature Detection

- 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
  - Corners
  - Application-specific patterns



Drowsiness detector
with IR illumination

[N. Eagle, EE368 class project]

Edge detection

- Idea (continous-space): Detect local gradient

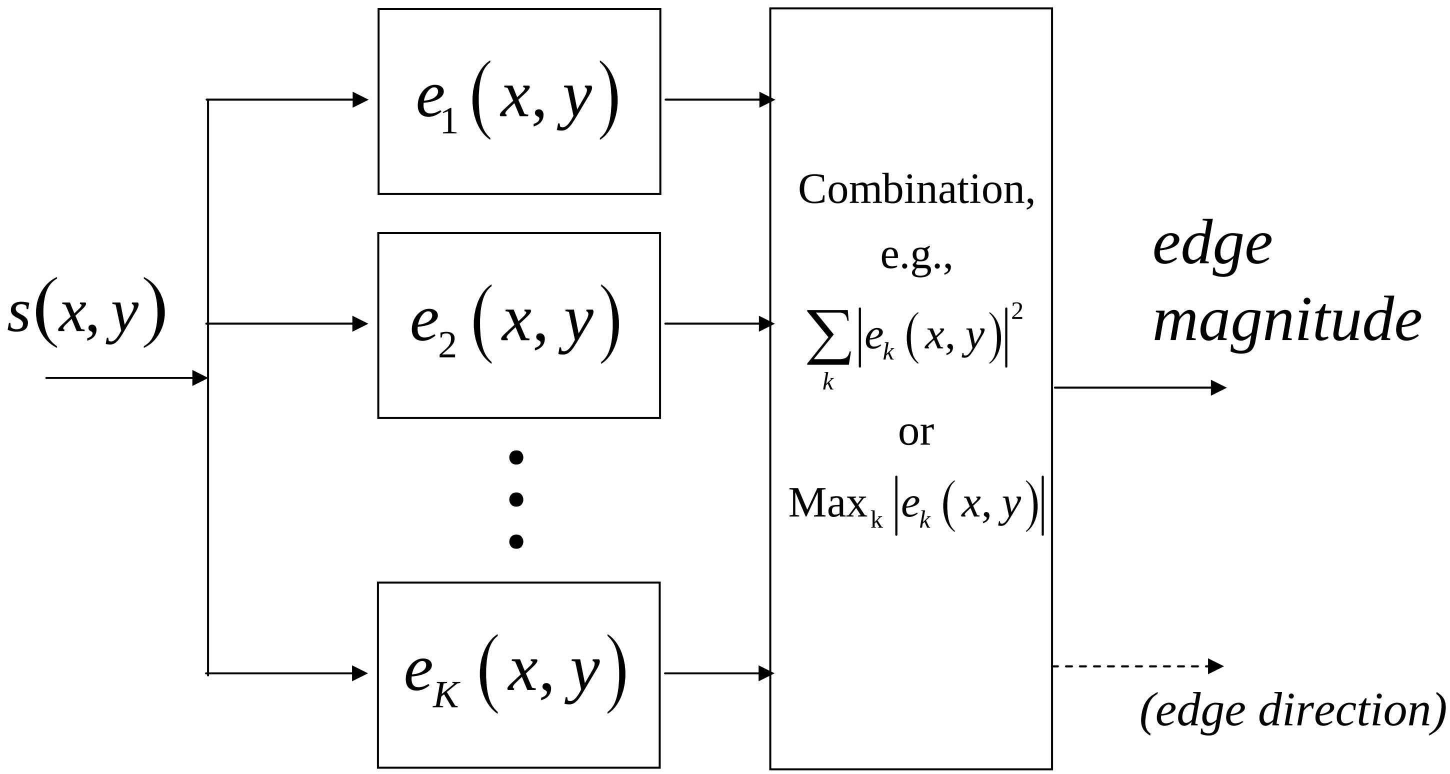
$$\left|grad\left(f\left(x,y\right)\right)\right| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

- Digital image:
  use finite differences
  instead

difference	$\begin{pmatrix} -1 & 1 \end{pmatrix}$
central difference	$\begin{pmatrix} -1 & [0] & 1 \end{pmatrix}$
Prewitt	$\begin{pmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{pmatrix}$
Sobel	$\begin{pmatrix} -1 & 0 & 1 \\ -2 & [0] & 2 \\ -1 & 0 & 1 \end{pmatrix}$

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

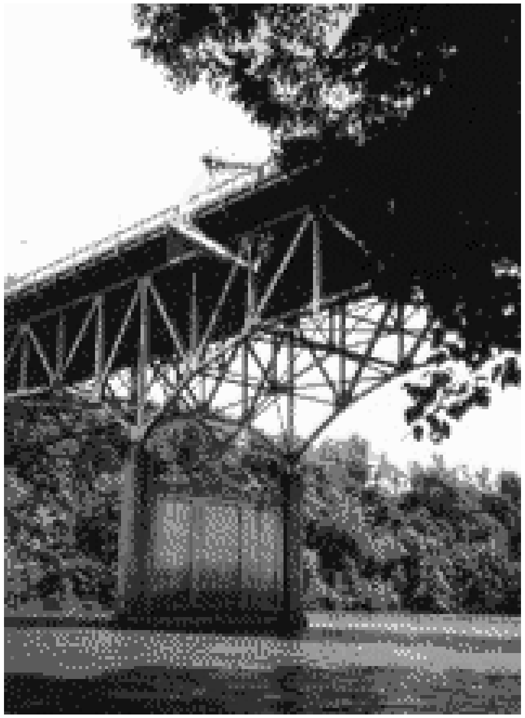
Prewitt $\begin{pmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} -1 & -1 & -1 \\ 0 & [0] & 0 \\ 1 & 1 & 1 \end{pmatrix}$

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

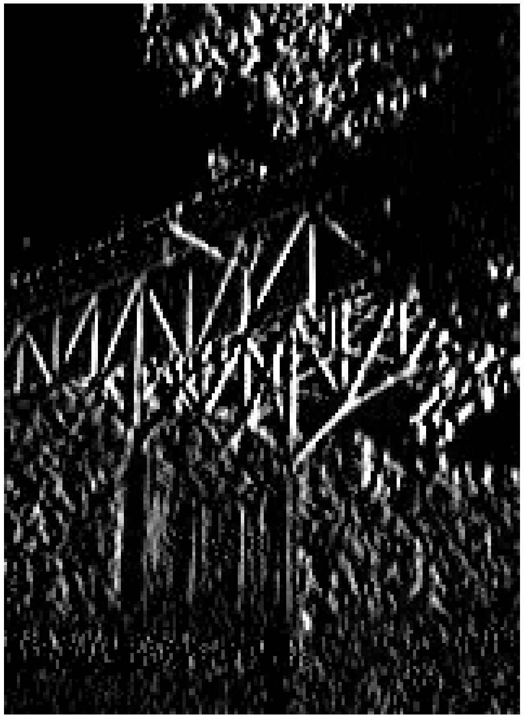
Kirsch $\begin{pmatrix} +5 & +5 & +5 \\ -3 & [0] & -3 \\ -3 & -3 & -3 \end{pmatrix} \begin{pmatrix} -3 & +5 & +5 \\ -3 & [0] & +5 \\ -3 & -3 & -3 \end{pmatrix} \begin{pmatrix} -3 & -3 & +5 \\ -3 & [0] & +5 \\ -3 & -3 & +5 \end{pmatrix} \begin{pmatrix} -3 & -3 & -3 \\ -3 & [0] & +5 \\ -3 & +5 & +5 \end{pmatrix}$

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

Prewitt operator example

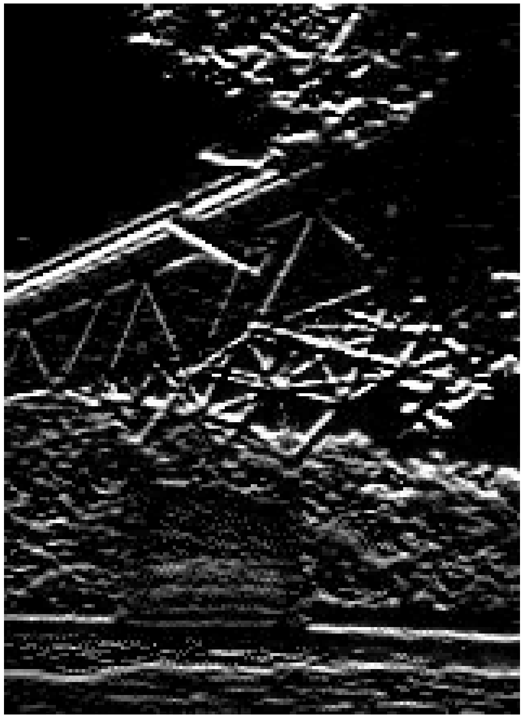


Original *Bridge*
220x160



magnitude of
image filtered with

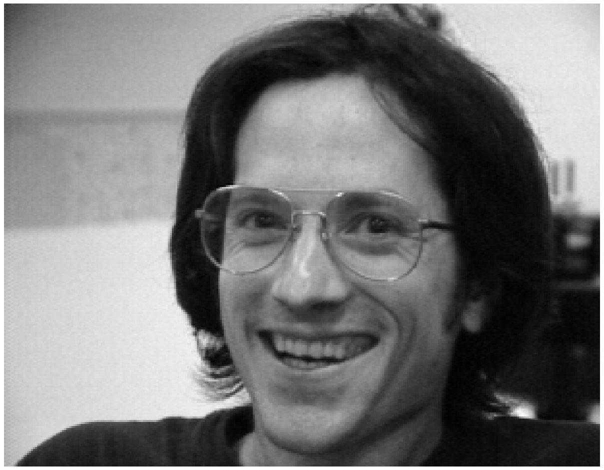
$$\begin{pmatrix} -1 & 0 & 1 \\ -1 & [0] & 1 \\ -1 & 0 & 1 \end{pmatrix}$$



magnitude of
image filtered with

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

Prewitt operator example (cont.)



Original *Billsface*
310x241

log magnitude of image filtered with

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

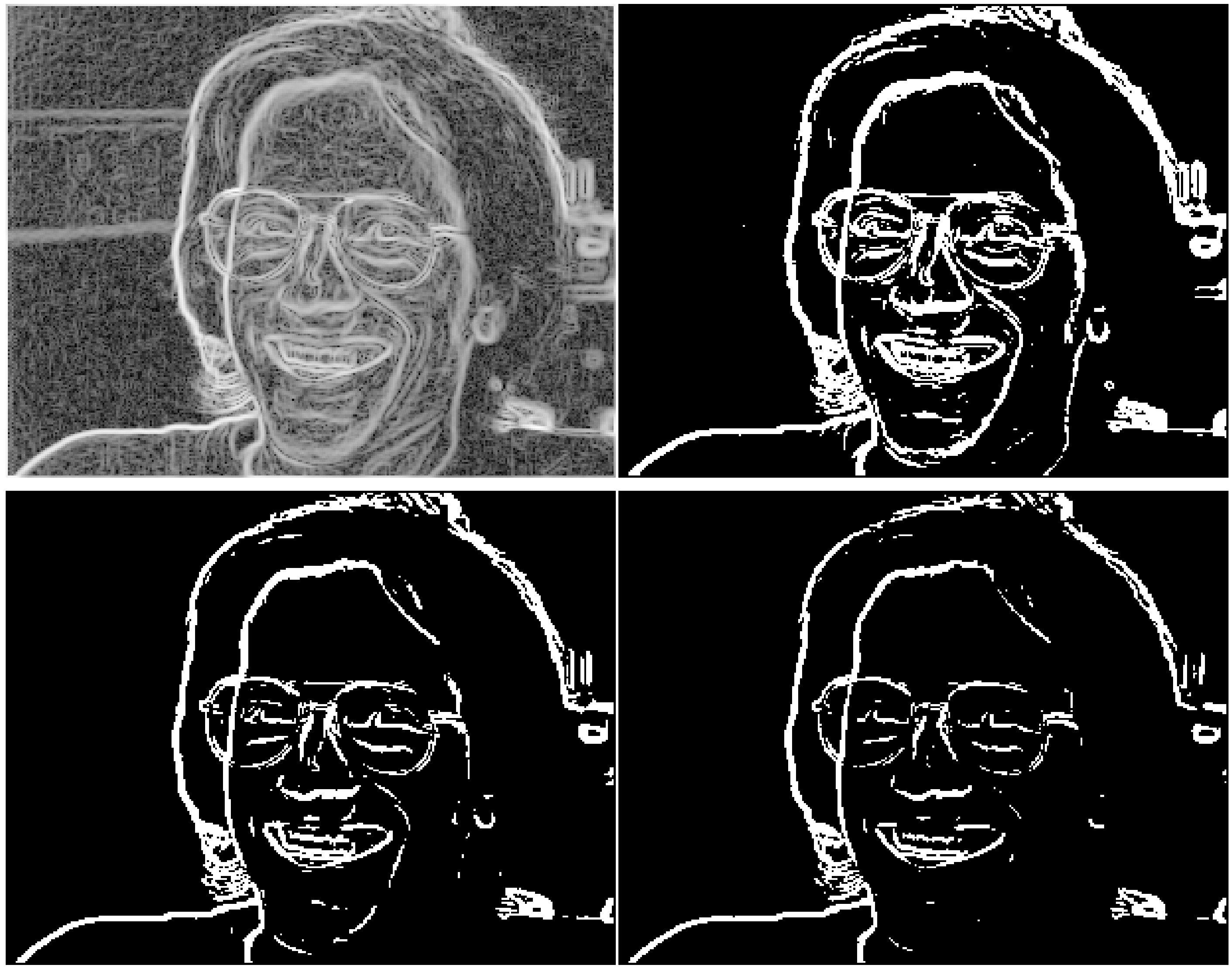
log magnitude of image filtered with

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

Prewitt operator example (cont.)

log sum of squared horizontal and vertical gradients

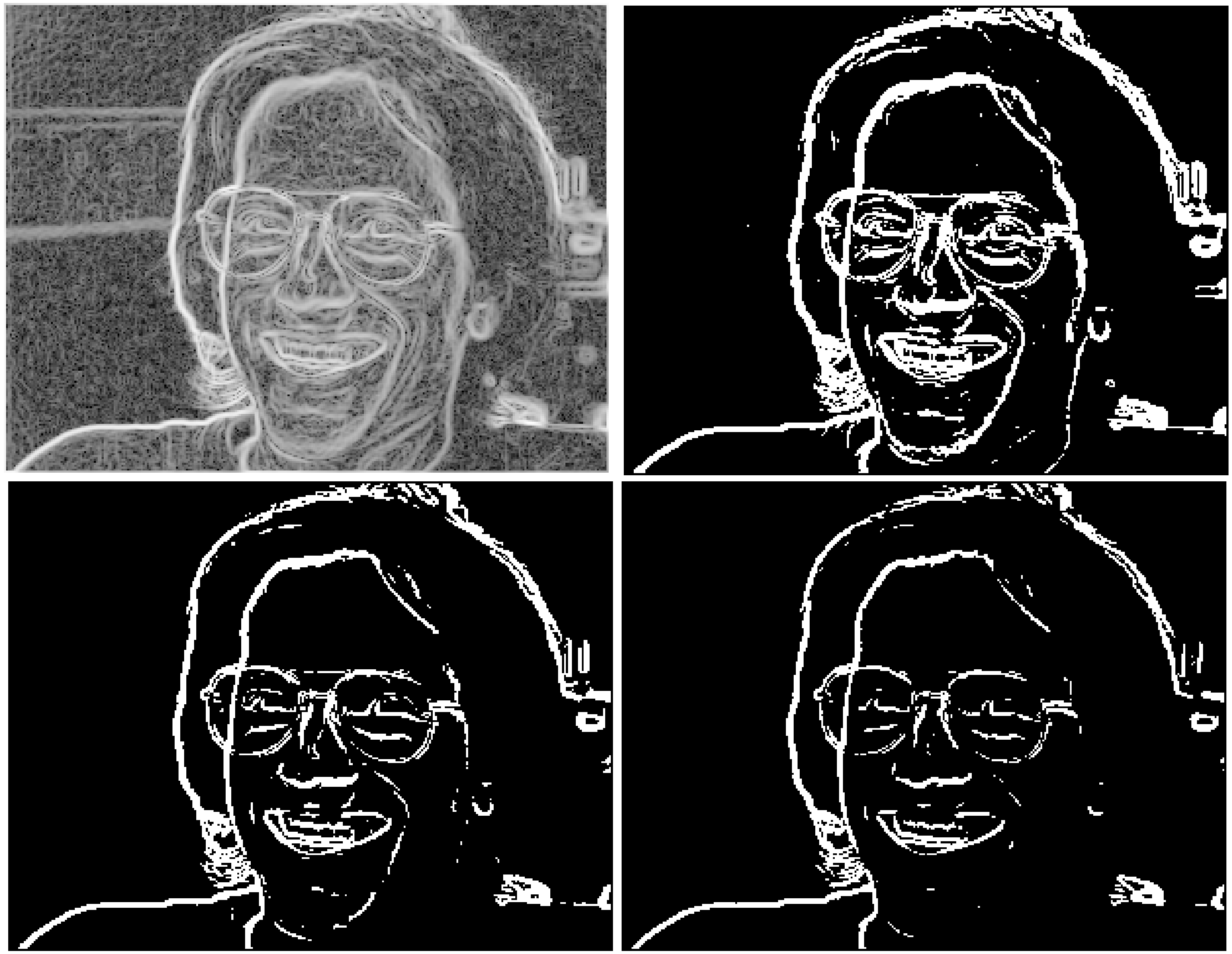
different thresholds



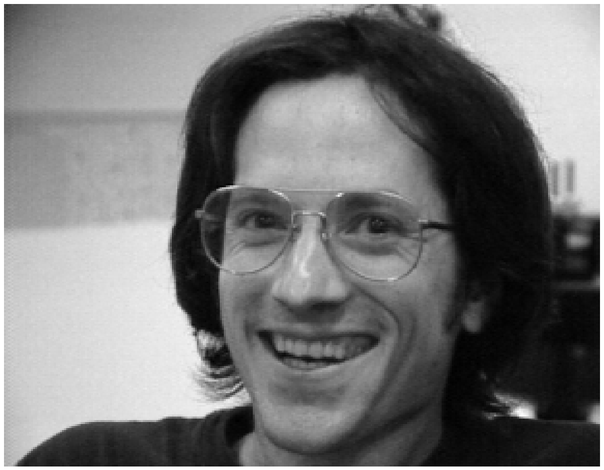
Sobel operator example

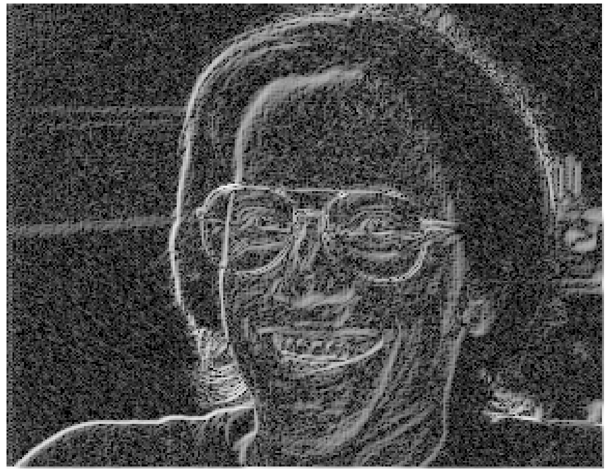
log sum of squared horizontal and vertical gradients

different thresholds



Roberts operator example





Original *Billsface*
309x240

log magnitude of image filtered with

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

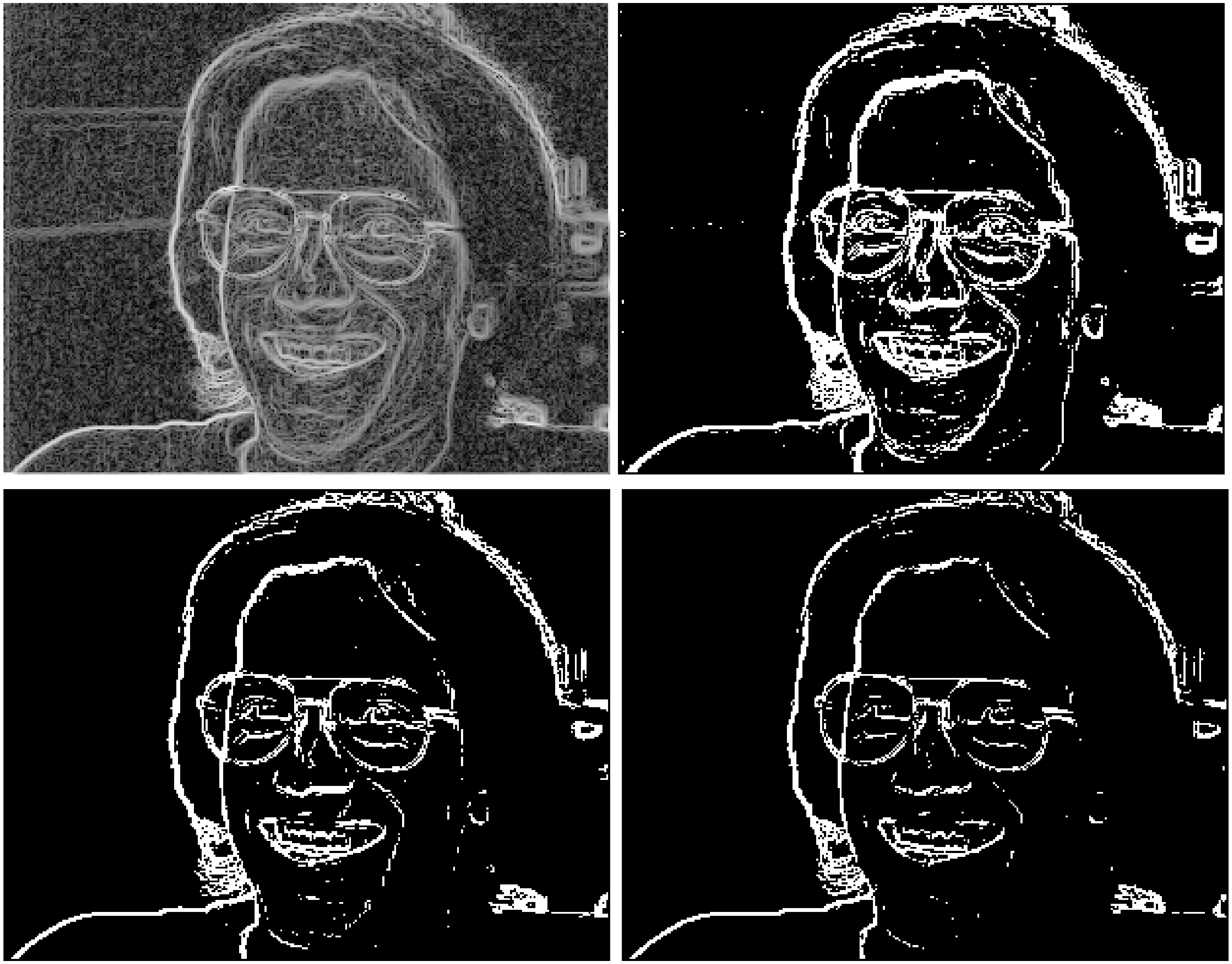
log magnitude of image filtered with

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

Roberts operator example (cont.)

log sum of squared diagonal gradients

different thresholds



Laplacian operator

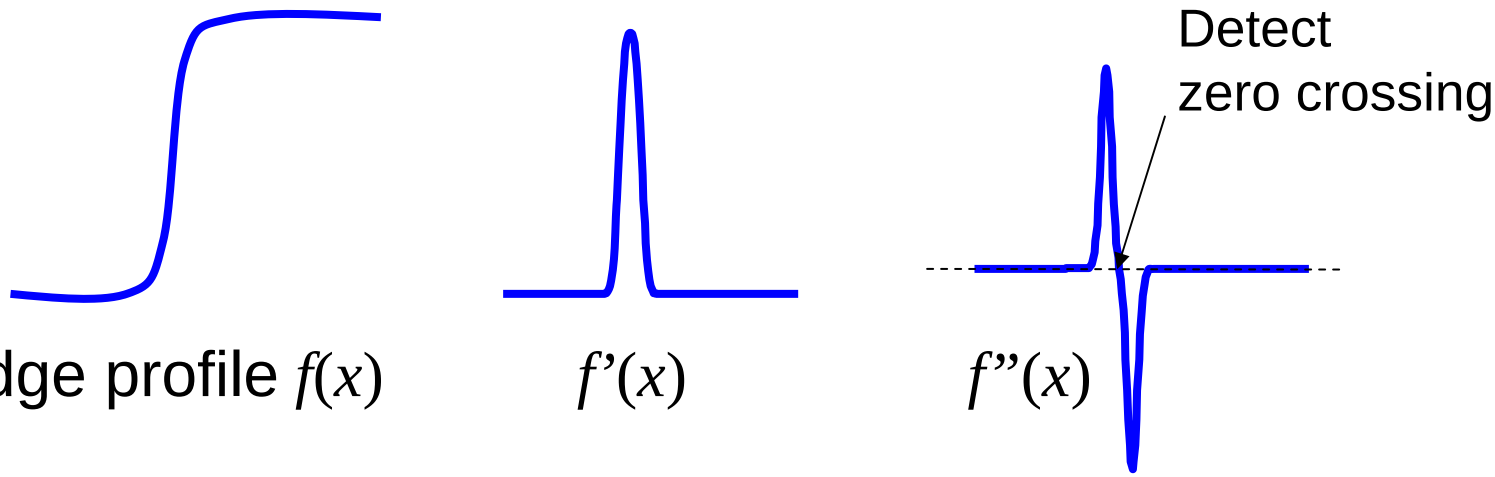
- Detect discontiuities by considering second derivative

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

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

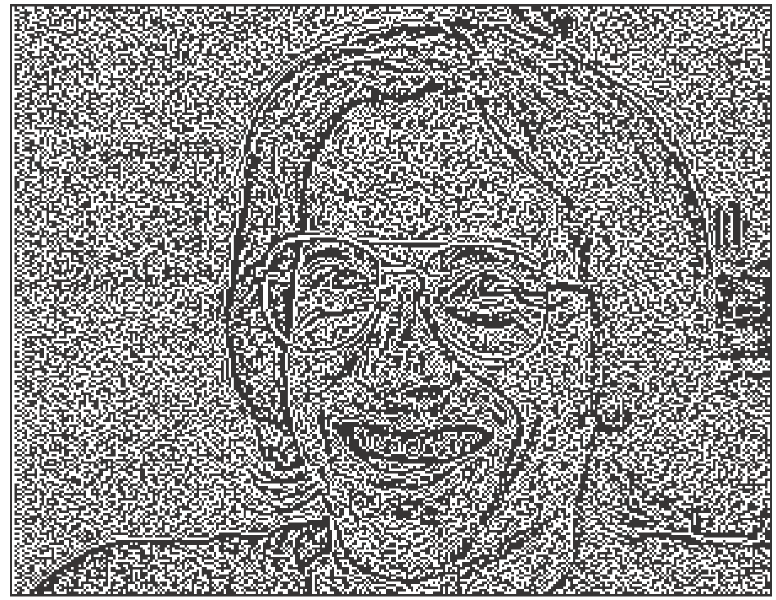
$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & [-4] & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 1 & 1 & 1 \\ 1 & [-8] & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

1-d illustration of 2^{nd} derivative edge detector



Edge profile $f(x)$ $f'(x)$ $f''(x)$

Zero crossings of Laplacian



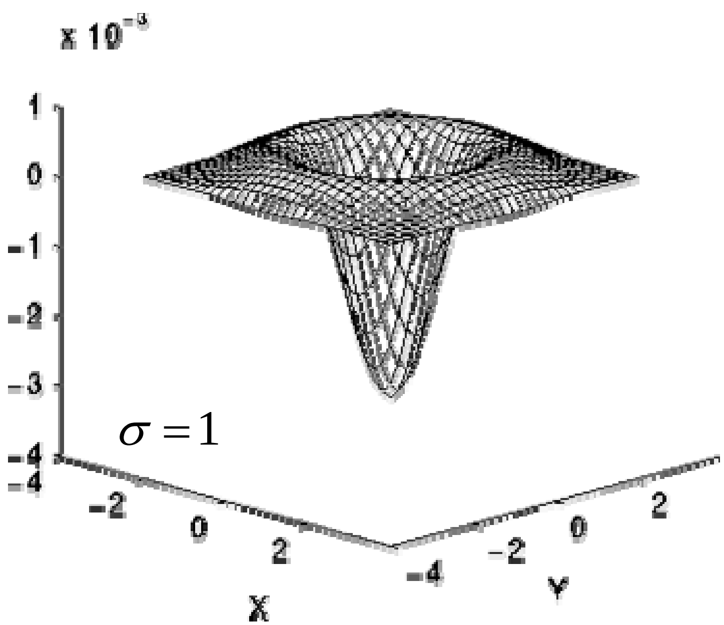
- Sensitive to very fine detail and noise ➔ blur image first
- Responds equally to strong and weak edges
  ➔ suppress edges with low gradient magnitude

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

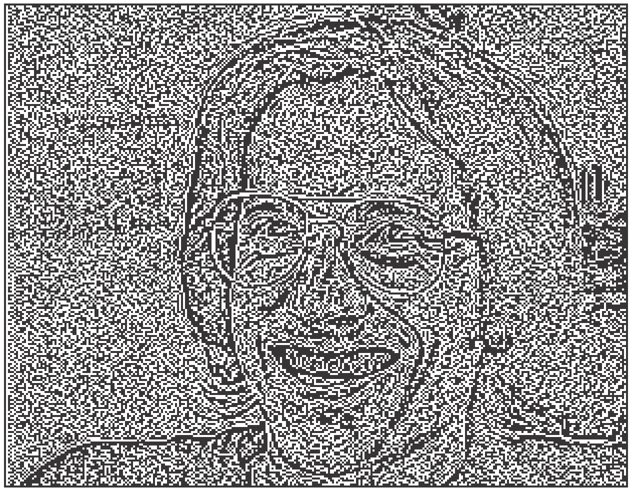


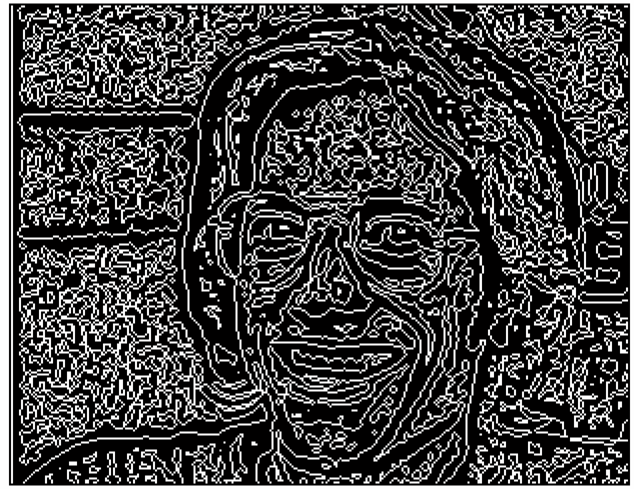
$\sigma = 1.4$

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

Zero crossings of LoG

w/o Gaussian





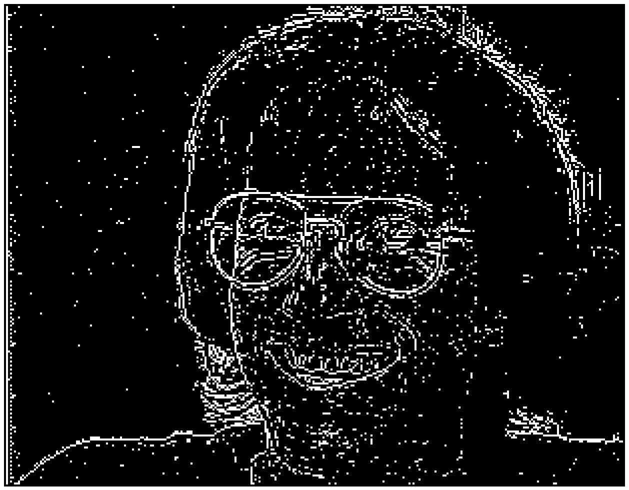
$\sigma = 1.4$

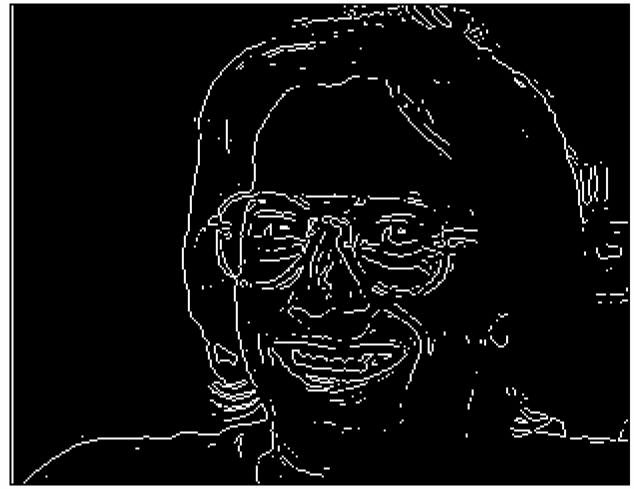
$\sigma = 3$

$\sigma = 6$

Zero crossings of LoG – gradient-based threshold

w/o Gaussian





$\sigma = 1.4$

$\sigma = 3$

$\sigma = 6$

Canny edge detector

1. Smooth image with a Gaussian filter
2. Compute gradient magnitude and angle (Sobel, Prewitt . . .)

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

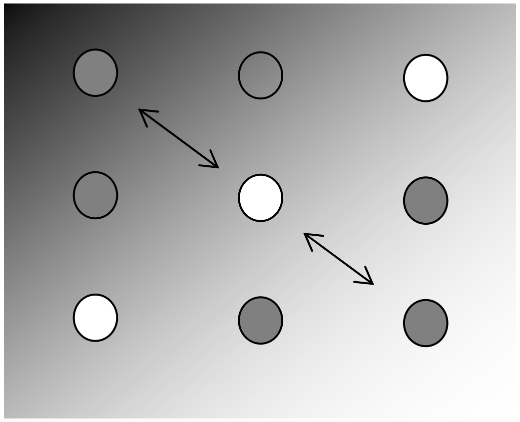
$$\alpha(x,y)=\tan^{-1}\left(\frac{\partial f}{\partial y}\Big/\frac{\partial f}{\partial x}\right)$$

3. Apply nonmaxima suppression to gradient magnitude image
4. Double thresholding to detect strong and weak edge pixels
5. Reject weak edge pixels not connected with strong edge pixels

[Canny, IEEE Trans. PAMI, 1986]

Canny nonmaxima suppression

- Quantize edge normal to one of four directions: horizontal, -45^o, vertical, +45^o
- If $M(x,y)$ is smaller than either of its neighbors in edge normal direction ➔ suppress; else keep.



[Canny, IEEE Trans. PAMI, 1986]

Canny thresholding and suppression of weak edges

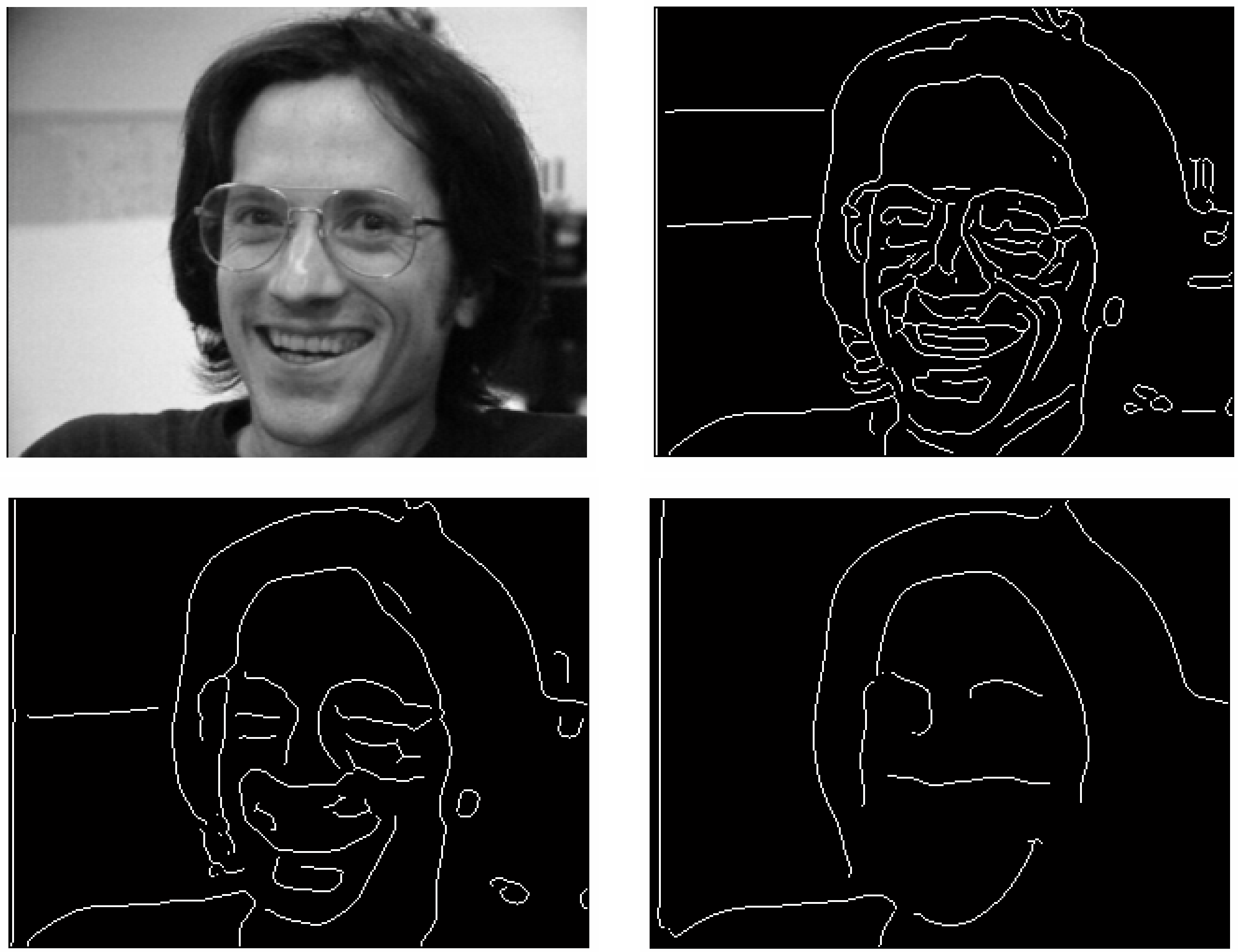
- Double-thresholding of gradient magnitude

$$\text{Strong edge:} \qquad M(x,y) \geq \theta_{high}$$
$$\text{Weak edge:} \quad \theta_{high} > M(x,y) \geq \theta_{low}$$

- Typical setting: $\theta_{high} / \theta_{low} = 2...3$
- Region labeling of edge pixels
- Reject regions without strong edge pixels

[Canny, IEEE Trans. PAMI, 1986]

Canny edge detector



$\sigma = 1.4$

$\sigma = 3$

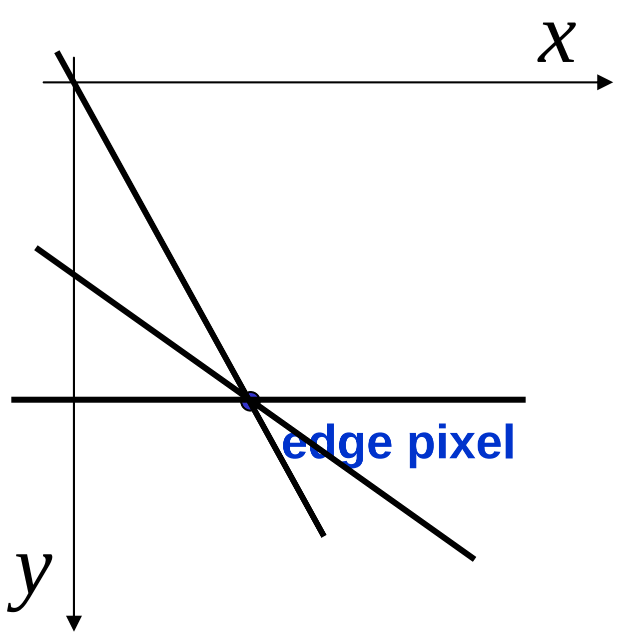
$\sigma = 6$

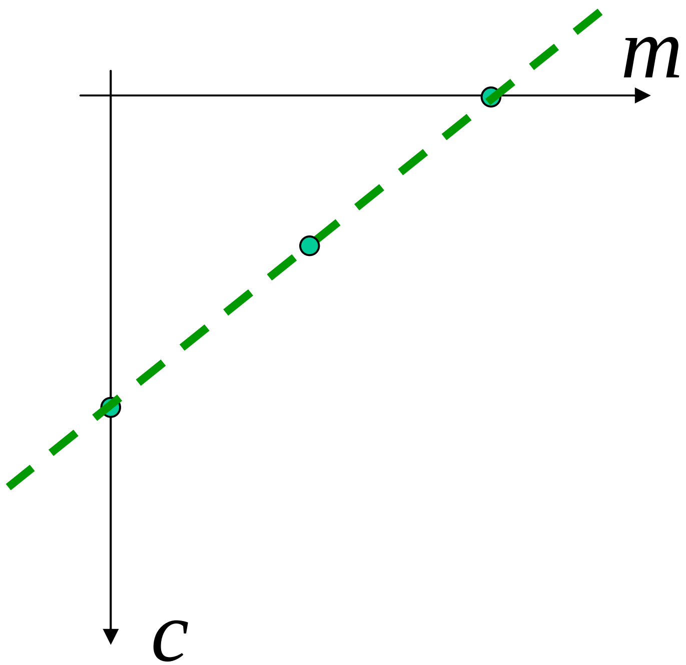
Canny edge detector

$$\sigma = 1.4$$

Hough transform

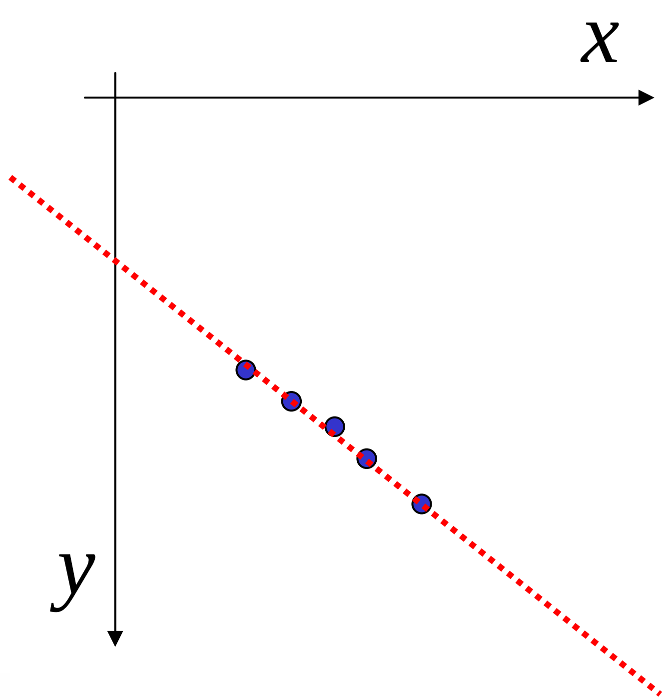
- Problem: fit a straight line (or curve) to a set of edge pixels
- Hough transform (1962): generalized template matching technique
- Consider detection of straight lines $y = mx + c$

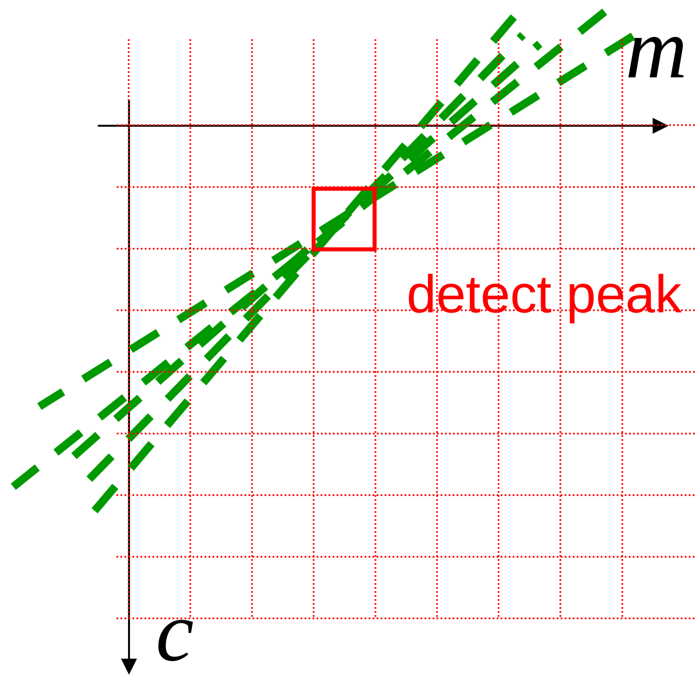




Hough transform (cont.)

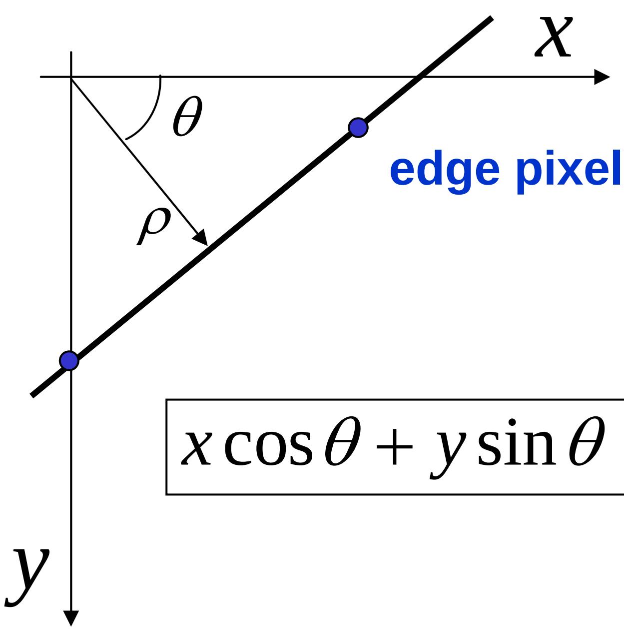
- Subdivide (m,c) plane into discrete “bins,” initialize all bin counts by 0
- Draw a line in the parameter space m,c for each edge pixel x,y and increment bin counts along line.
- Detect peak(s) in (m,c) plane



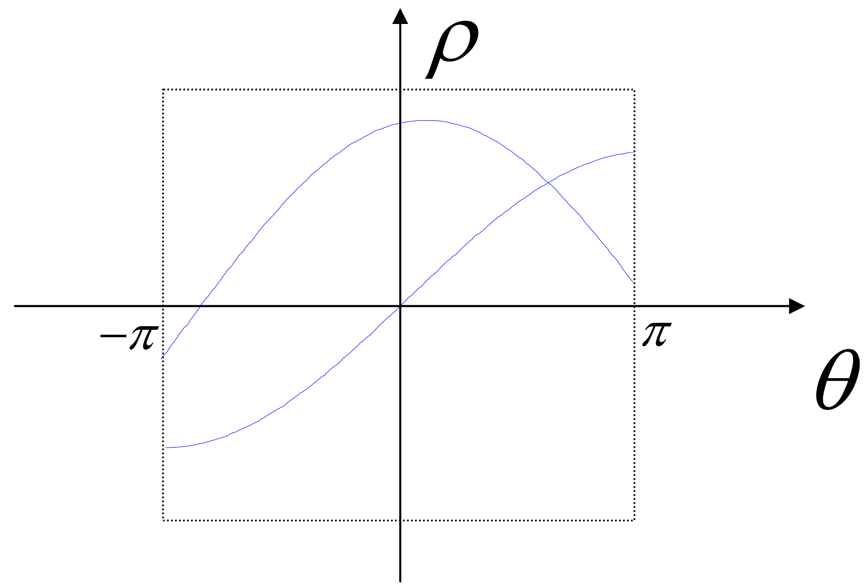


Hough transform (cont.)

- Alternative parameterization avoids infinite-slope problem



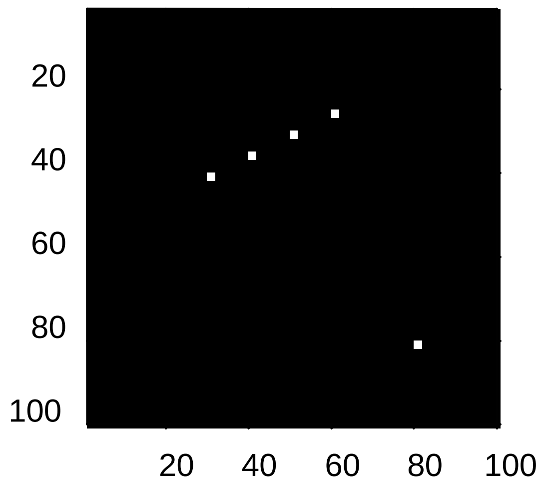
$$\boxed{x\cos\theta + y\sin\theta = \rho}$$

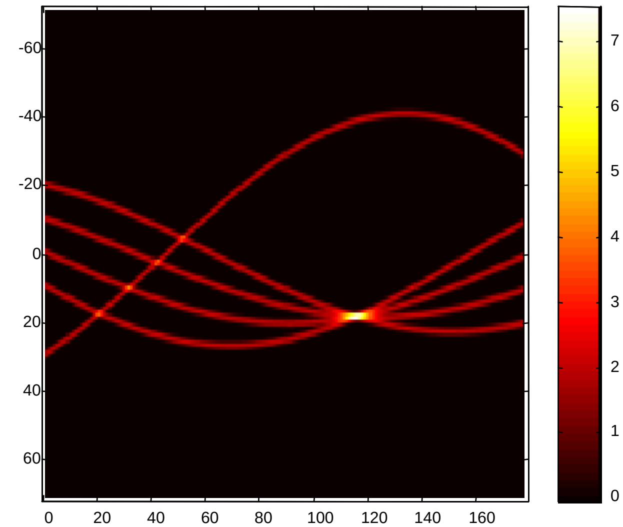


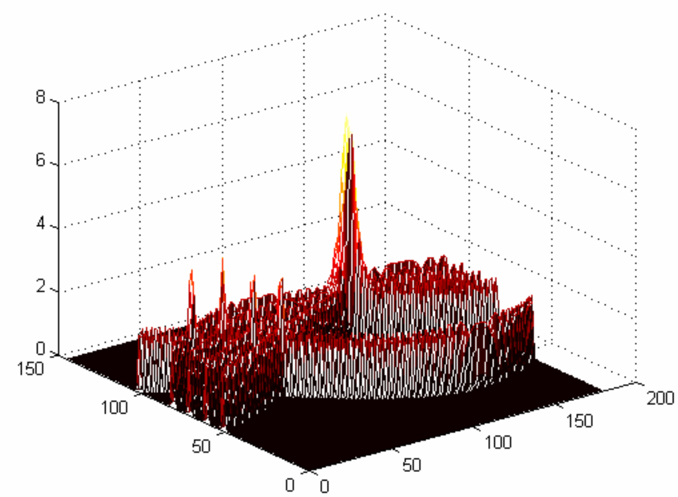
- Similar to Radon transform

Hough transform Example A

Original image



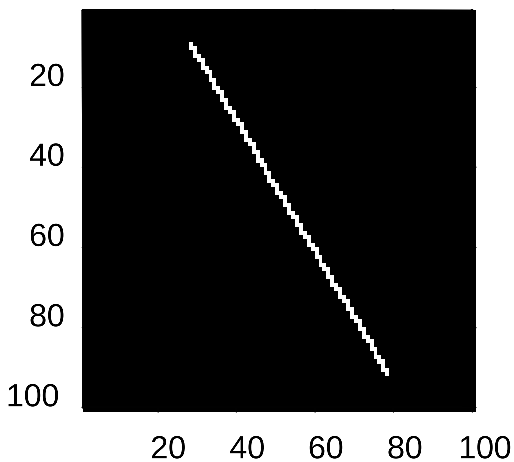


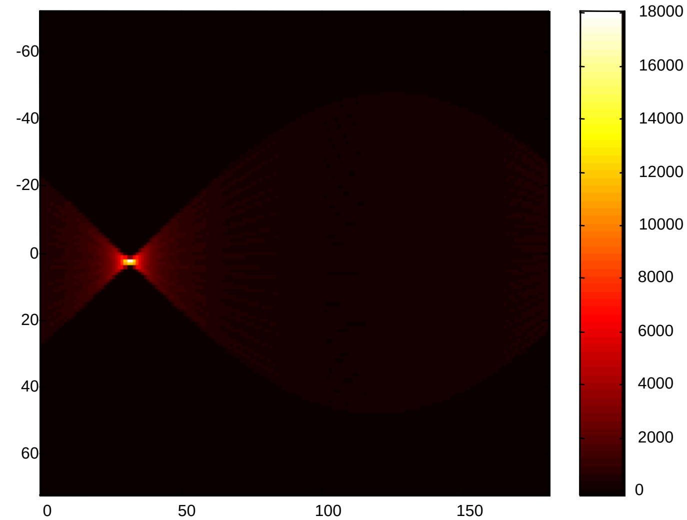


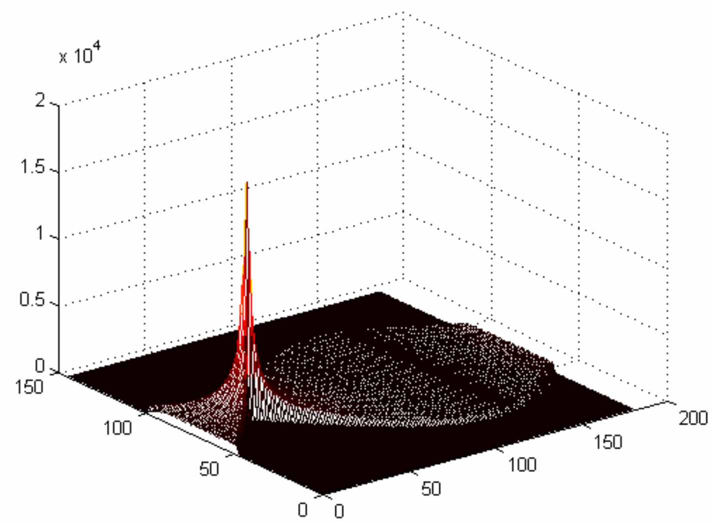
Courtesy: P. Salembier

Hough transform Example B

Original image



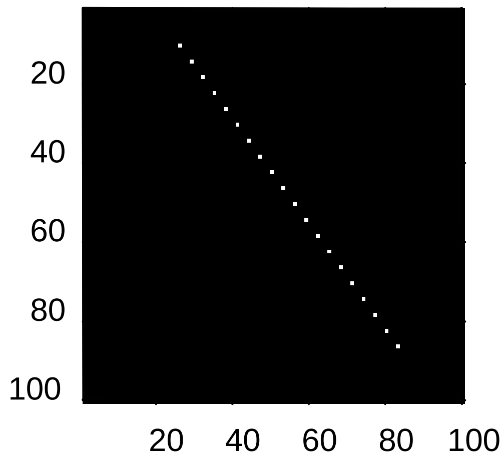


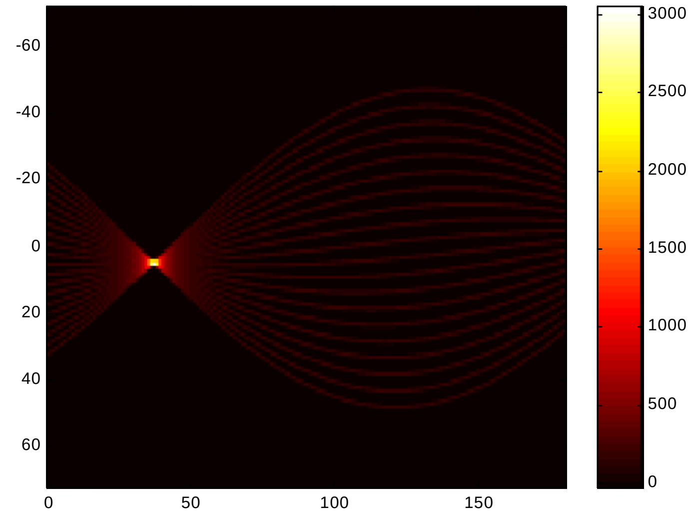


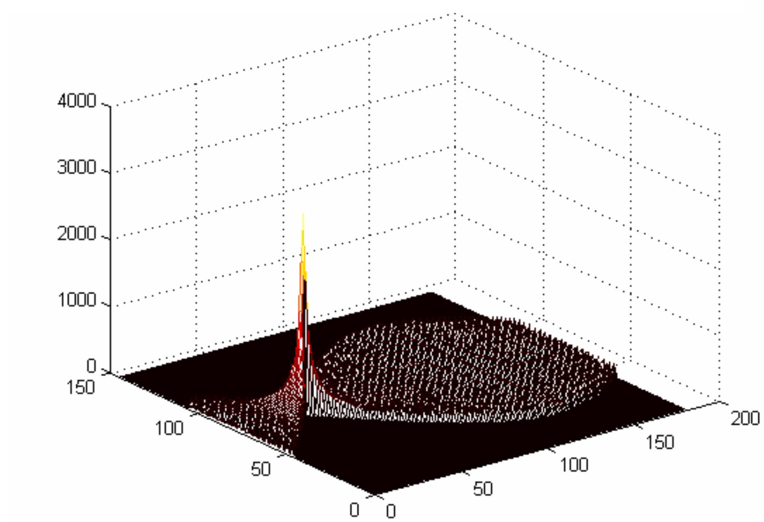
Courtesy: P. Salembier

Hough transform Example C

Original image



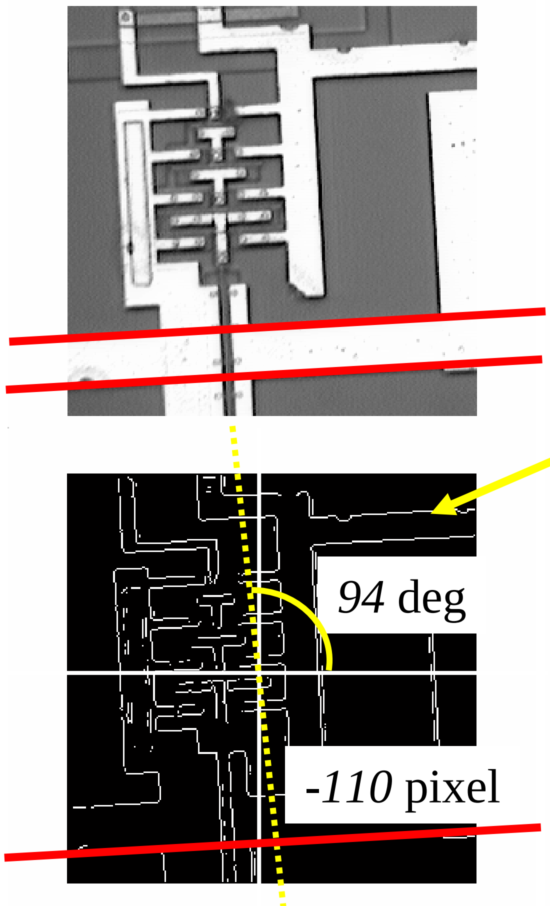




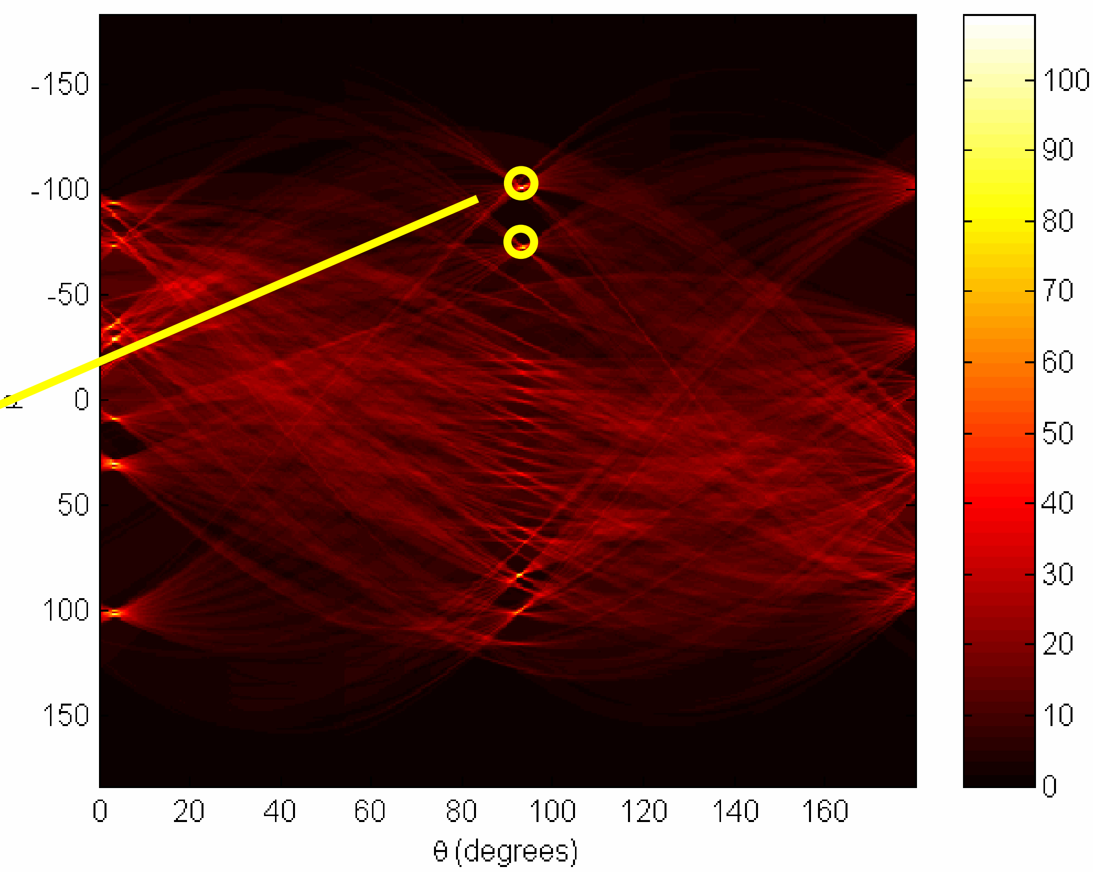
Courtesy: P. Salembier

Hough transform example

Original IC image (256x256)

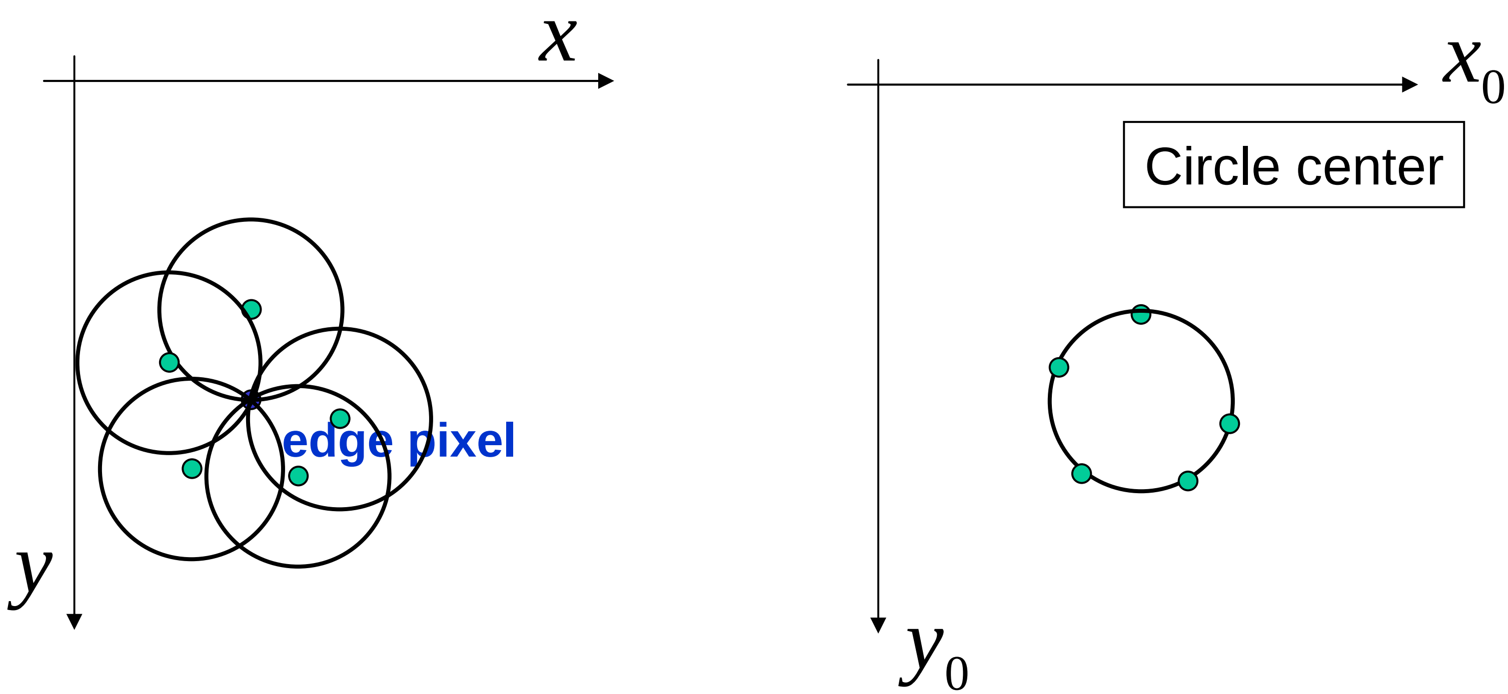


Edge detection (Prewitt)



Circle detection by Hough transform

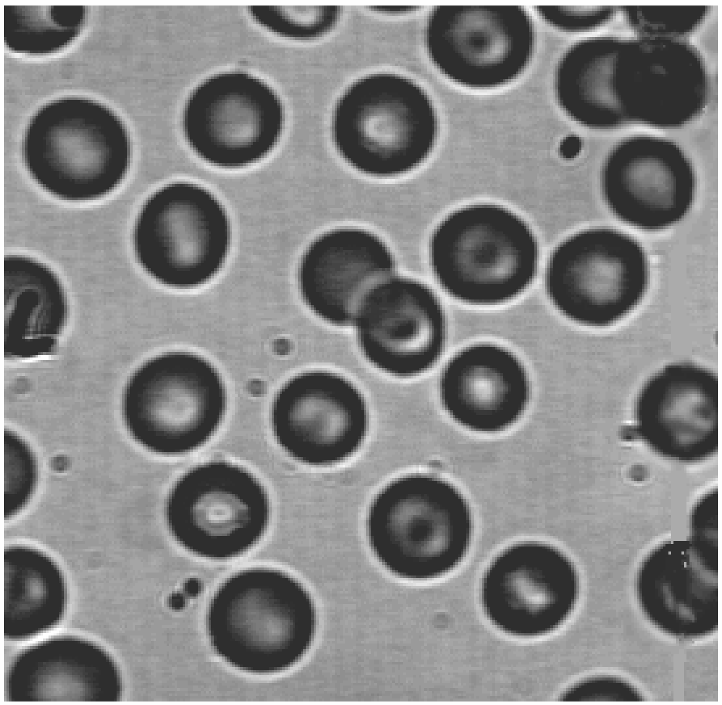
- Find circles of fixed radius r



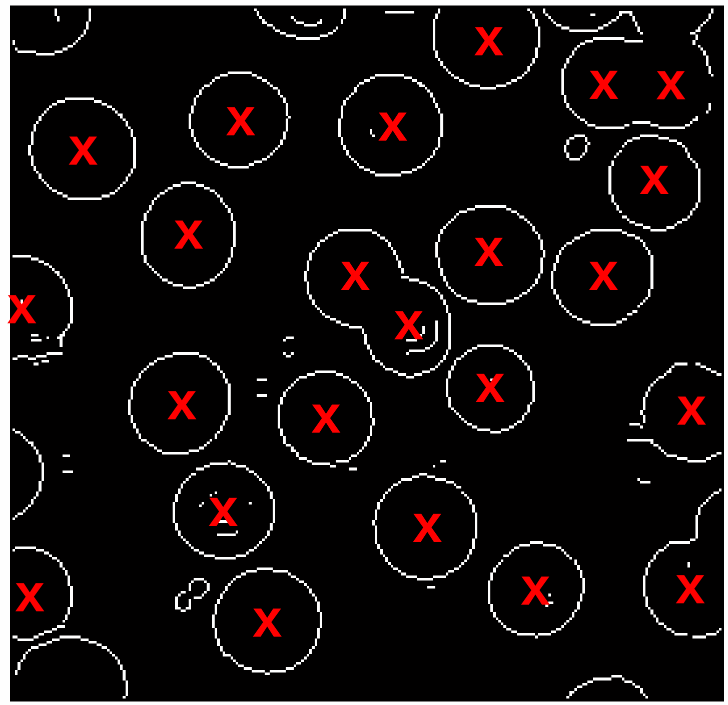
- For circles of undetermined radius, use 3-d Hough transform for parameters (x_0, y_0, r)

Example: circle detection by Hough transform

Original *blood* image

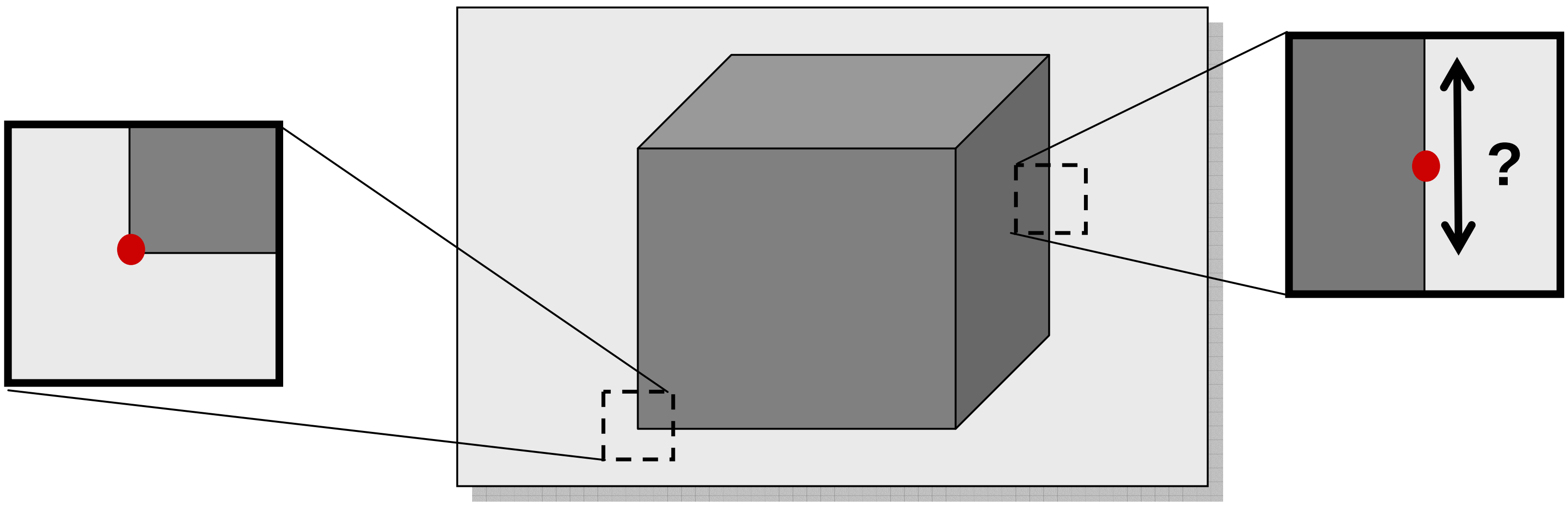


Prewitt edge detection



Detecting corner points

- Many applications benefit from features localized in (x,y)
- Edges well localized only in one direction ➔ detect corners



- Desirable properties of corner detector
  - Accurate localization
  - Invariance against shift, rotation, scale, brightness change
  - Robust against noise, high repeatability

What patterns can be localized most accurately?

- Local displacement sensitivity

$$S(\Delta x, \Delta y) = \sum_{(x,y)\in window} \left[f(x,y) - f(x+\Delta x, y+\Delta y) \right]^2$$

- Linear approximation for small $\Delta x, \Delta y$

$$f(x+\Delta x, y+\Delta y) \approx f(x,y) + f_x(x,y)\Delta x + f_y(x,y)\Delta y$$

$$\begin{aligned}
S(\Delta x, \Delta y) &\approx \sum_{(x,y)\in window} \left[\left(\begin{matrix} f_x(x,y) & f_x(x,y) \end{matrix} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \right) \right]^2 \\
&= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \left(\sum_{(x,y)\in window} \begin{bmatrix} f_x^2(x,y) & f_x(x,y) f_y(x,y) \\ f_x(x,y) f_y(x,y) & f_y^2(x,y) \end{bmatrix} \right) \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix} \\
&= \begin{pmatrix} \Delta x & \Delta y \end{pmatrix} \mathbf{M} \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}
\end{aligned}$$

- Iso-sensitivity curves are ellipses

Keypoint detection

Often based on eigenvalues λ_1, λ_2 of "structure matrix" (aka "normal matrix" aka "second-moment matrix")

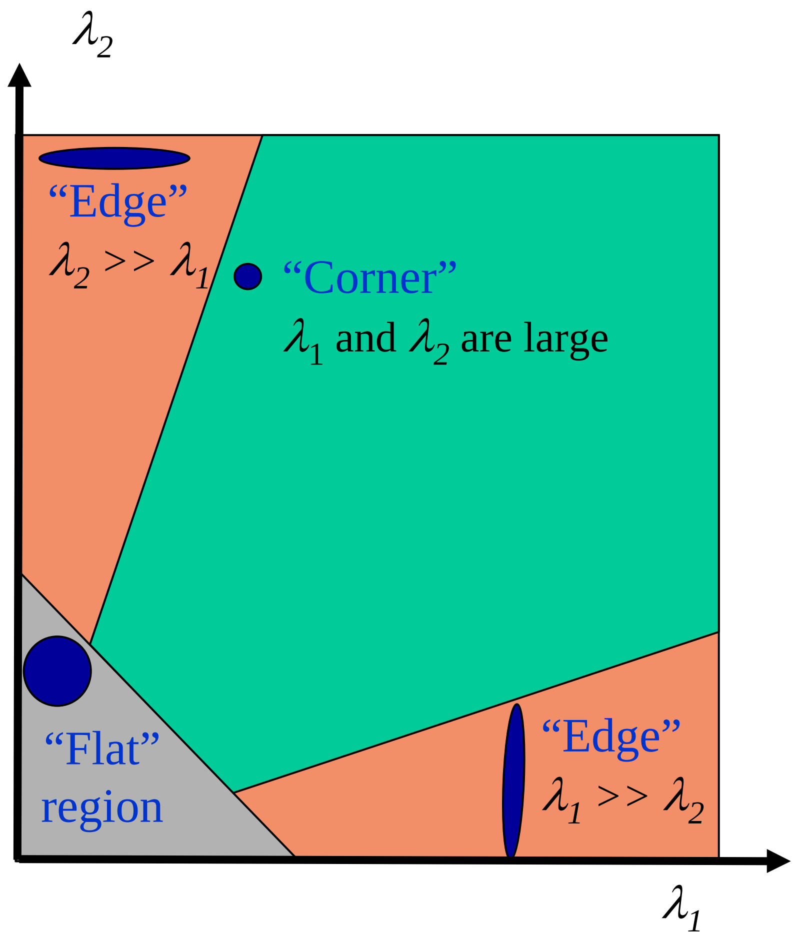
$$\mathbf{M} = \begin{bmatrix} \sum_{(x,y)\in window} f_x^2(x,y) & \sum_{(x,y)\in window} f_x(x,y)\, f_y(x,y) \\ \sum_{(x,y)\in window} f_x(x,y)\, f_y(x,y) & \sum_{(x,y)\in window} f_y^2(x,y) \end{bmatrix}$$

$f_x(x,y)$ – horizontal image gradient
$f_y(x,y)$ – vertical image gradient

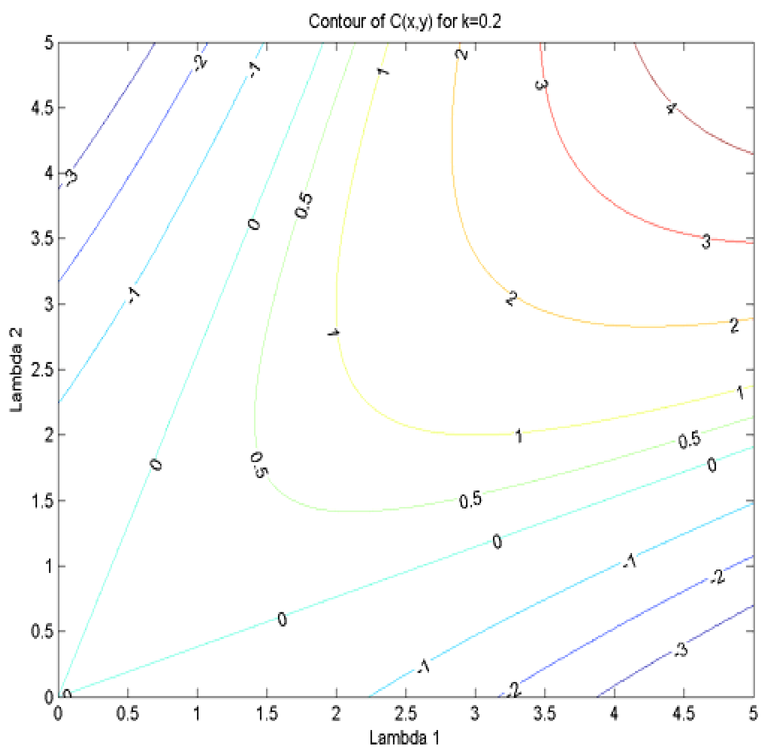
Measure of "cornerness"

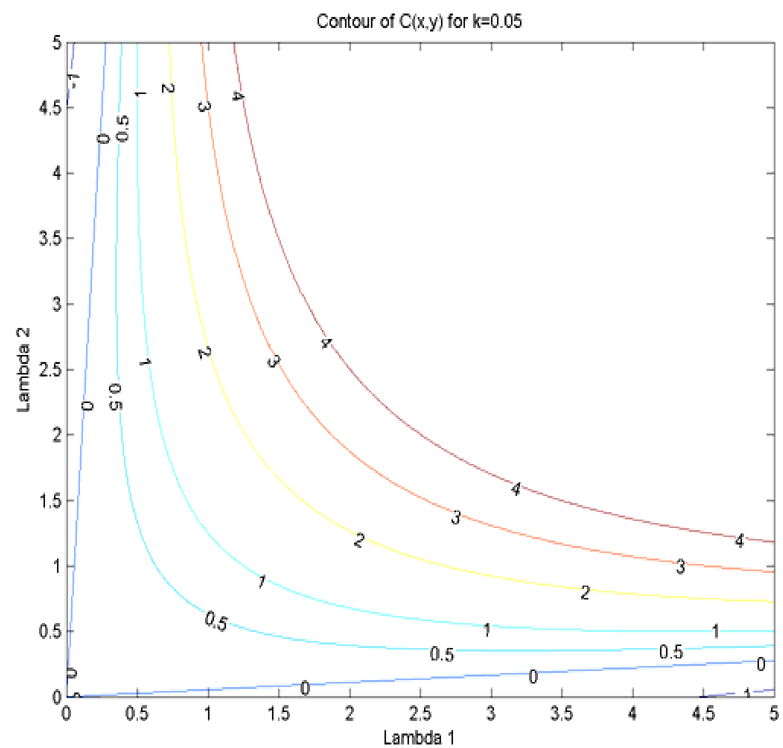
$$C(x,y) = \det(\mathbf{M}) - k \cdot \left(trace(\mathbf{M})\right)^2 = \lambda_1\lambda_2 - k \cdot \left(\lambda_1 + \lambda_2\right)$$

[Harris, Stephens, 1988]



Contour plot of Harris cornerness

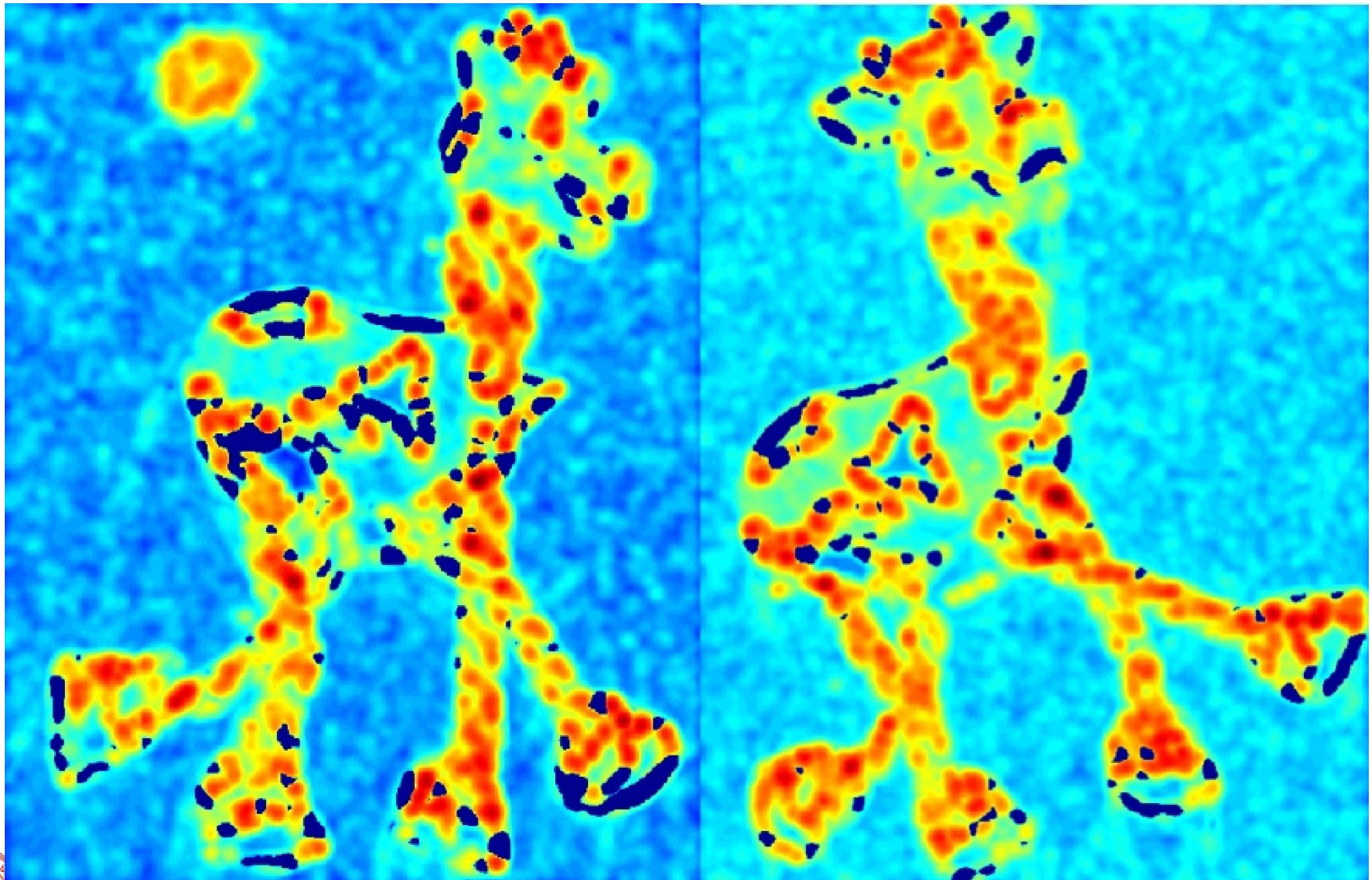




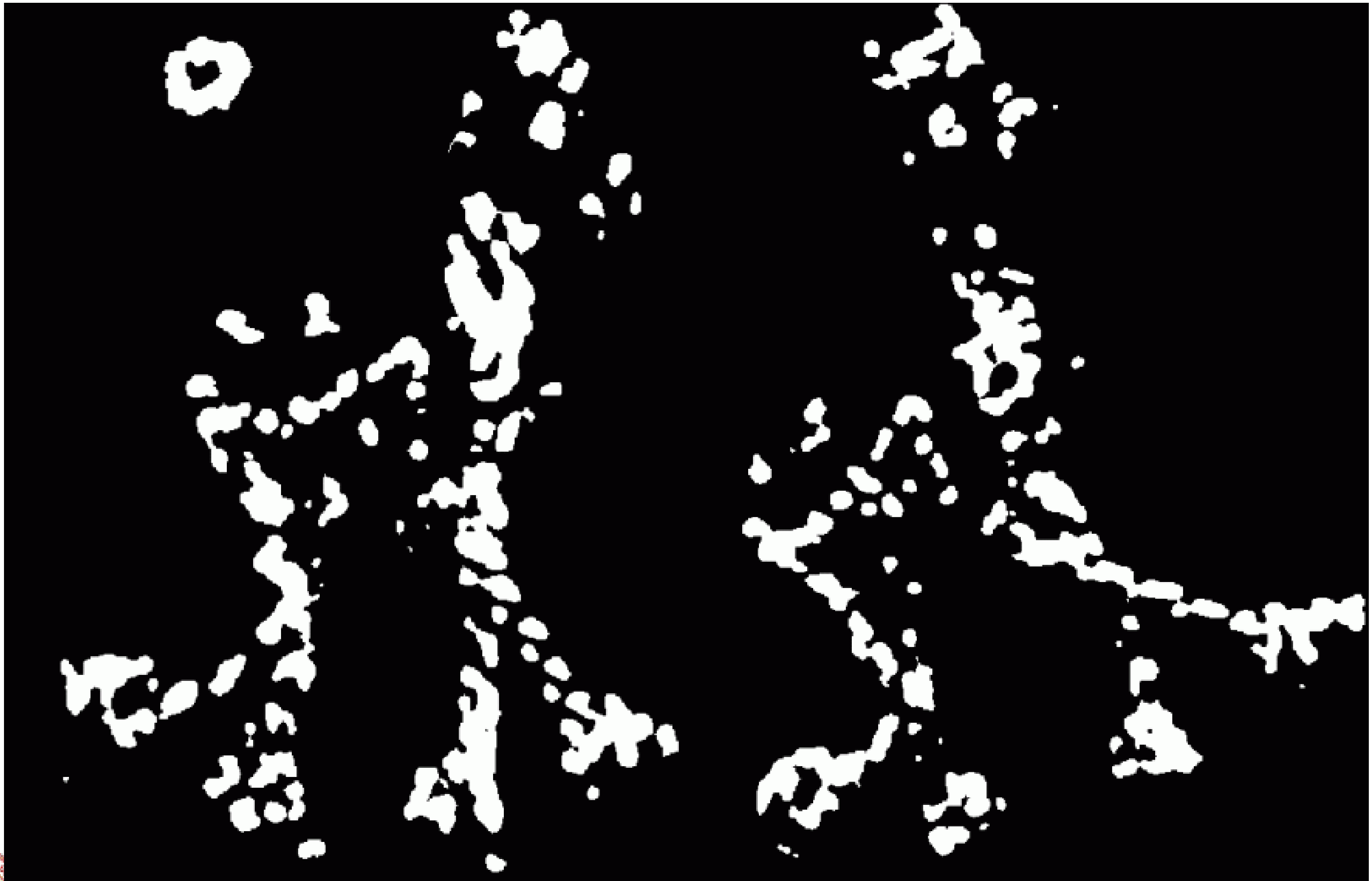
$$C(x,y) = \det(\mathbf{M}) - k \cdot \left(trace(\mathbf{M})\right)^2$$
$$= \lambda_1 \lambda_2 - k \cdot (\lambda_1 + \lambda_2)$$

Keypoint Detection: Input

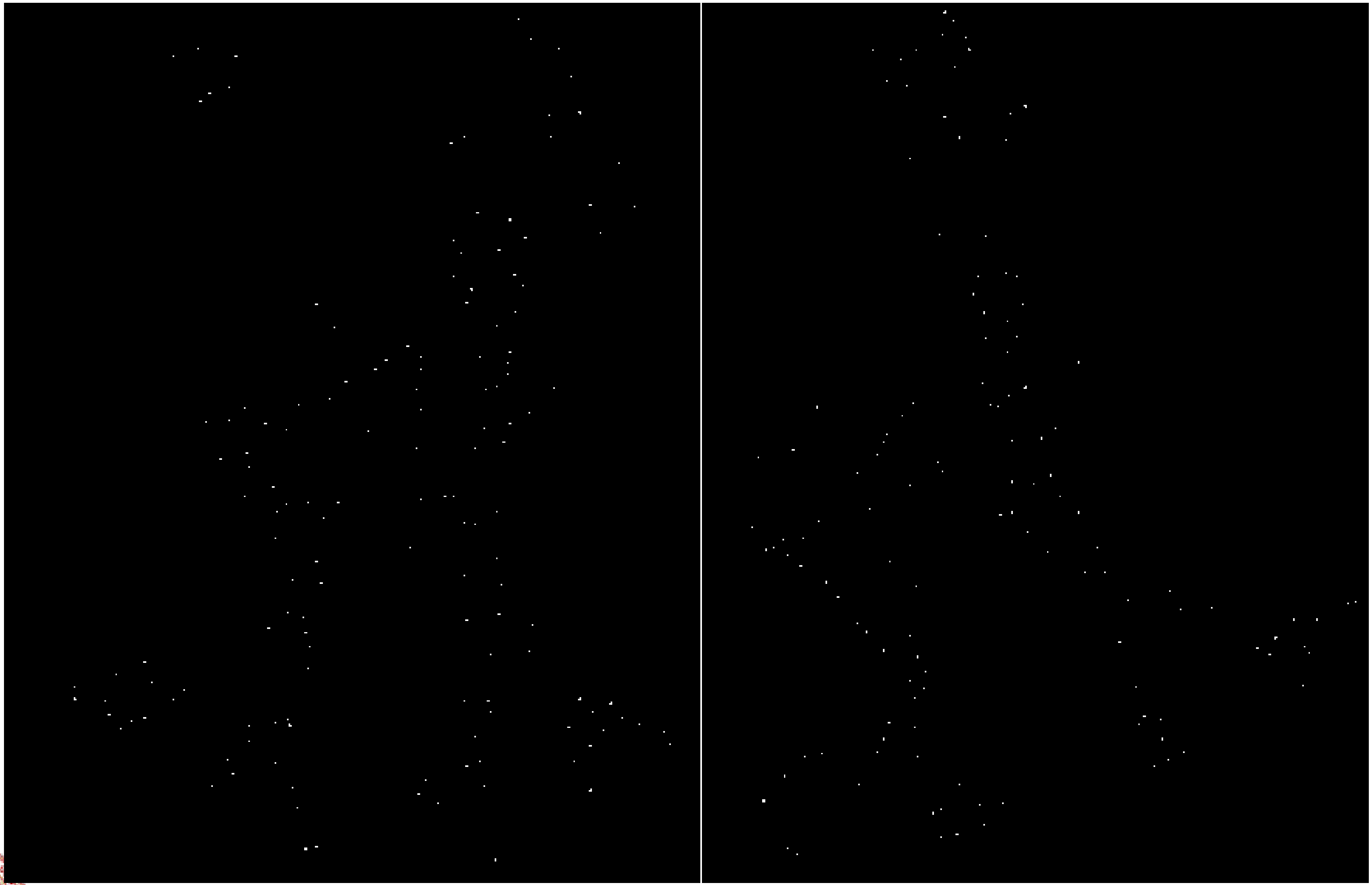
Harris cornerness



Thresholded cornerness



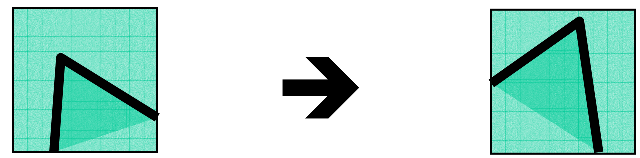
Local maxima of cornerness



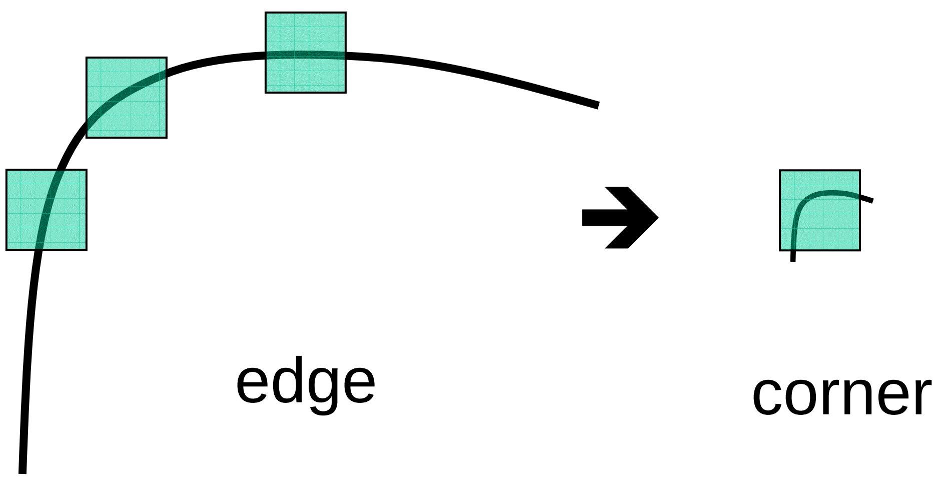
Superimposed keypoints

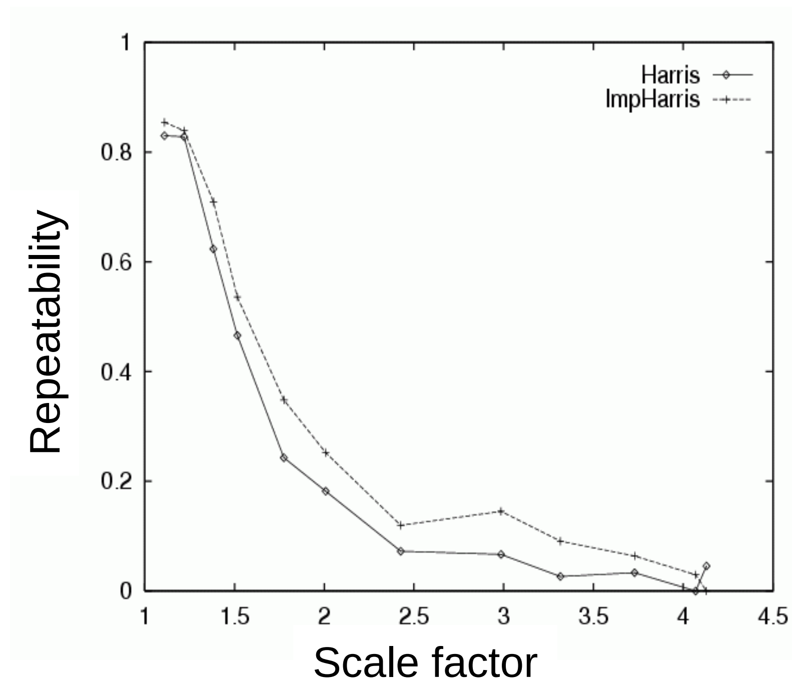
Robustness of Harris Corner Detector

- Invariant to brightness offset: $f(x,y) \rightarrow f(x,y) + c$
- Invariant to shift and rotation



- Not invariant to scaling





[Schmid, 2000]

Haralick Corner Detector

- Step1: Window-of-interest detection
  - Calculate horizontal gradient $f_x(x,y)$ and vertical gradient $f_y(x,y)$ using, e.g., Sobel operator
  - For each measurement window, determine

$$\mathbf{M} = \begin{bmatrix} \sum_{(x,y)\in window} f_x^2(x,y) & \sum_{(x,y)\in window} f_x(x,y)\, f_y(x,y) \\ \sum_{(x,y)\in window} f_x(x,y)\, f_y(x,y) & \sum_{(x,y)\in window} f_y^2(x,y) \end{bmatrix}$$

Measure of "cornerness": $\det(\mathbf{M}) = \lambda_1 \lambda_2$

Circularity of ellipse: $q = 1 - \left(\frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2} \right)^2$

λ_1, λ_2 - eigenvalues of $\mathbf{M}$

  - Thresholding:

$$w = \begin{cases} \det(\mathbf{M}) & \text{if } \det(\mathbf{M}) > w_{\min} \text{ and } q > q_{\min} \\ 0 & \text{else} \end{cases}$$

  - Non-maximum suppression on *w*

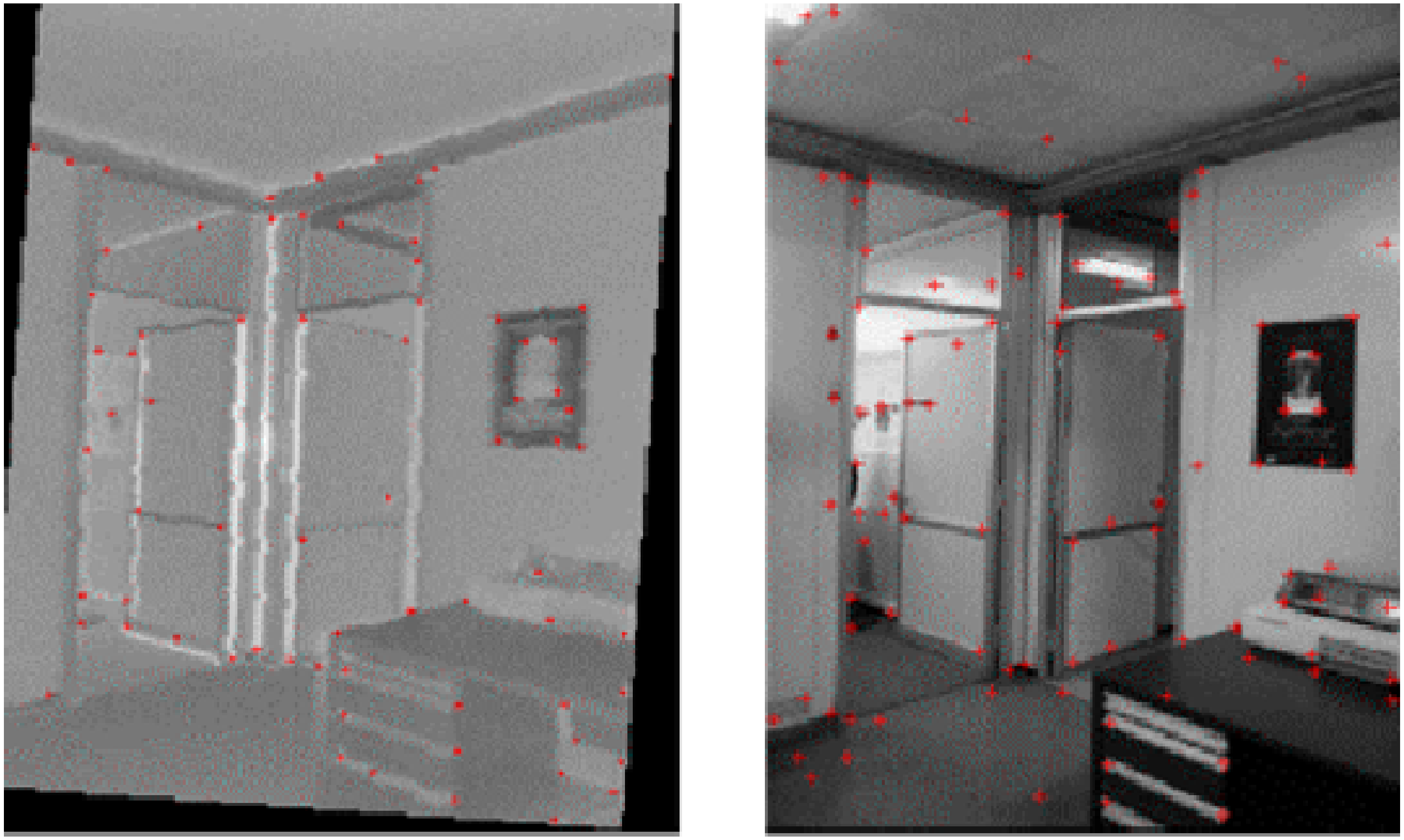
Haralick Corner Detector (cont.)

- Step 2: Extraction of interesting point in window of interest: centroid with product of horizontal and vertical gradient as weight

$$x = \frac{\sum\limits_{x,y \in window} \left(f_x(x,y) \times f_y(x,y) \times x \right)}{\sum\limits_{x,y \in window} \left(f_x(x,y) \times f_y(x,y) \right)}$$

$$y = \frac{\sum\limits_{x,y \in window} \left(f_x(x,y) \times f_y(x,y) \times y \right)}{\sum\limits_{x,y \in window} \left(f_x(x,y) \times f_y(x,y) \right)}$$

Haralick Corner Detector (cont.)

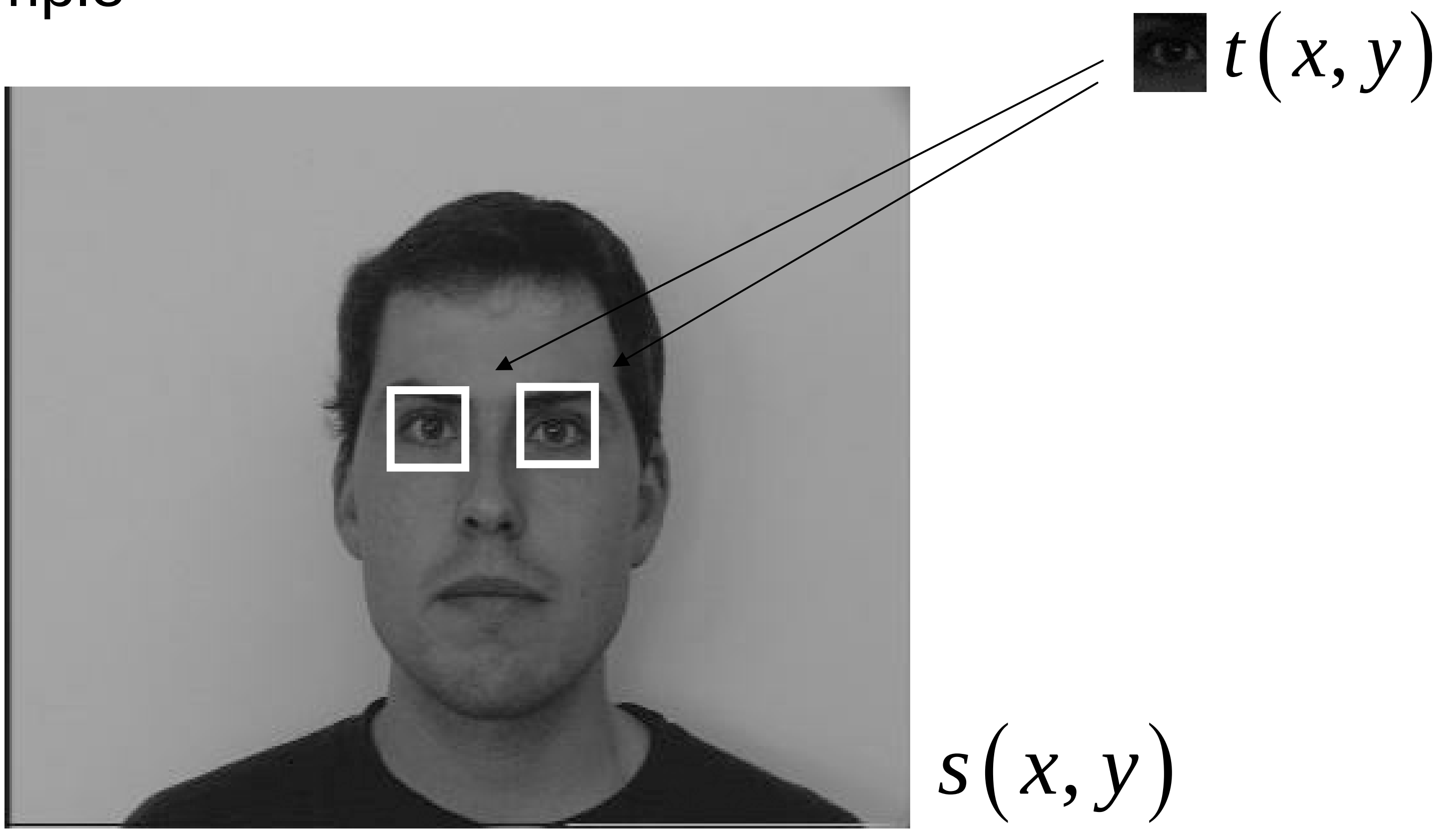


Corner detector applied to reflectance image, acquired by laser scanner (left) and by camera (right), to perform automatic image registration.

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/DIAS2/

Template matching

- Problem: locate an object, described by a template $t(x,y)$, in the image $s(x,y)$
- Example



Template matching (cont.)

- Search for the best match by minimizing mean-squared error

$$E(p,q) = \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} \left[s(x,y) - t(x-p,y-q) \right]^2$$

$$= \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} \left| s(x,y) \right|^2 + \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} \left| t(x,y) \right|^2 - 2 \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} s(x,y) \cdot t(x-p,y-q)$$

- Equivalently, maximize *area correlation*

$$r(p,q) = \sum_{x=-\infty}^{\infty} \sum_{y=-\infty}^{\infty} s(x,y) \cdot t(x-p,y-q) = s(p,q) * t(-p,-q)$$

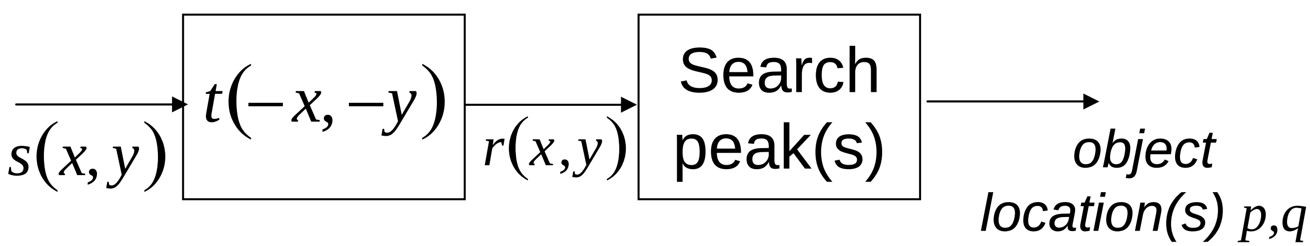
- Area correlation is equivalent to convolution of image $s(x,y)$ with impulse response $t(-x,-y)$

Template matching (cont.)

- From Cauchy-Schwarz inequality

$$r(p,q)=\sum_{x=-\infty}^{\infty}\sum_{y=-\infty}^{\infty} s(x,y)\cdot t(x-p,y-q)\le\sqrt{\left[\sum_{x=-\infty}^{\infty}\sum_{y=-\infty}^{\infty}\left|s(x,y)\right|^2\right]\cdot\left[\sum_{x=-\infty}^{\infty}\sum_{y=-\infty}^{\infty}\left|t(x,y)\right|^2\right]}$$

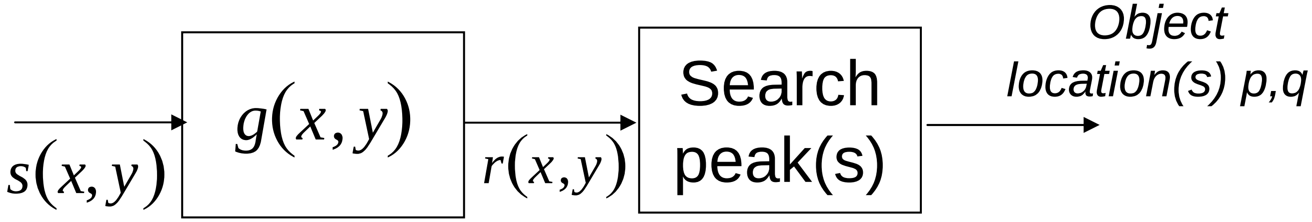
- Equality, iff $s(x,y)=\alpha\cdot t(x-p,y-q)\;$ with $\alpha\ge 0$
- Blockdiagram of template matcher



- Remove mean before template matching to avoid bias towards bright image areas

Matched filtering

- Consider signal detection problem



- Signal model

shifted template

Other objects: "noise" "clutter"
psd $\Phi_{nn}\left(e^{j\omega_x}, e^{j\omega_y}\right)$

$$s(x,y) = t(x-p, y-q) + n(x,y)$$

- Problem: design filter $g(x,y)$ to maximize

correct peak

$$SNR = \frac{\left| r(p,q) \right|^2}{E\left\{ \left| n(x,y) * g(x,y) \right|^2 \right\}}$$

false readings

Matched filtering (cont.)

- Optimum filter has frequency response

$$G\left(e^{j\omega_x}, e^{j\omega_y}\right) = \frac{T^*\left(e^{j\omega_x}, e^{j\omega_y}\right)}{\Phi_{nn}\left(e^{j\omega_x}, e^{j\omega_y}\right)}$$

- Proof:

$$SNR = \frac{\left|r(p,q)\right|^2}{E\left\{\left|n(x,y) * g(x,y)\right|^2\right\}} = \frac{\left|\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} G\left(e^{j\omega_x}, e^{j\omega_y}\right) T\left(e^{j\omega_x}, e^{j\omega_y}\right) d\omega_x d\omega_y\right|^2}{\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \left|G\left(e^{j\omega_x}, e^{j\omega_y}\right)\right|^2 \Phi_{nn}\left(e^{j\omega_x}, e^{j\omega_y}\right) d\omega_x d\omega_y}$$

$$= \frac{\left|\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} \left[G\Phi_{nn}^{1/2}\right]\left[\Phi_{nn}^{-1/2}T\right] d\omega_x d\omega_y\right|^2}{\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |G|^2 \Phi_{nn} d\omega_x d\omega_y} \leq \frac{\left[\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |G|^2 \Phi_{nn} d\omega_x d\omega_y\right]\left[\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |T|^2 \Phi_{nn}^{-1} d\omega_x d\omega_y\right]}{\int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |G|^2 \Phi_{nn} d\omega_x d\omega_y}$$

$$= \int_{-\pi}^{\pi}\int_{-\pi}^{\pi} |T|^2 \Phi_{nn}^{-1} d\omega_x d\omega_y$$

max. SNR

Schwarz inequality,

with equality, iff $G\Phi_{nn}^{1/2} = \alpha \cdot \left[\Phi_{nn}^{-1/2} T\right]^*$

Matched filtering (cont.)

- Optimum detection: prefiltering & template matching

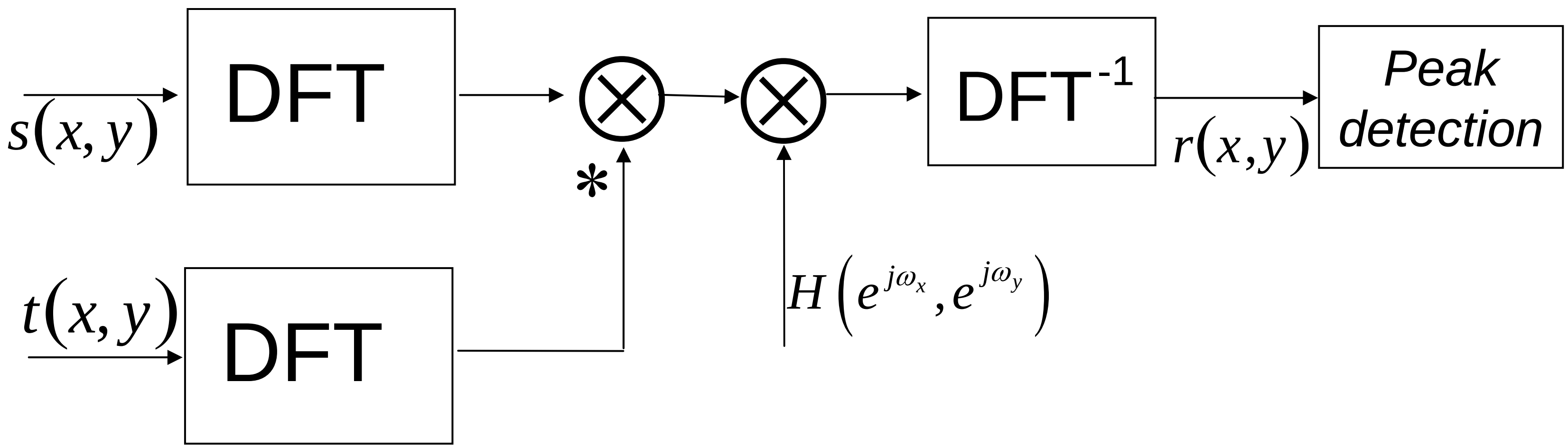
$$s(x,y) \rightarrow \boxed{h(x,y)} \rightarrow \boxed{t(-x,-y)} \xrightarrow{r(x,y)} \boxed{\text{Search peak(s)}} \rightarrow \textit{object location(s) } x,y$$

$$h(x,y)=\frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\frac{1}{\Phi_{nn}\left(e^{j\omega_x},e^{j\omega_y}\right)}e^{j\omega_x x+j\omega_y y}d\omega_x d\omega_y$$

- For white noise $n(x,y)$, no prefiltering $h(x,y)$ required
- Low frequency clutter: highpass prefilter

Frequency domain correlation

- Efficient implementation employing the Discrete Fourier transform



- Phase correlation

$$H\left(e^{j\omega_x}, e^{j\omega_y}\right) = \frac{1}{\left|S\left(e^{j\omega_x}, e^{j\omega_y}\right)\right|\left|T\left(e^{j\omega_x}, e^{j\omega_y}\right)\right|}$$