Lecture 7 Further Edge Detection

COMP6223 Computer Vision (MSc)

What better ways are there to detect edges?

Book
pp
131 -
177

Department of
Electronics and
Computer Science

UNIVERSITY OF
Southampton
School of Electronics
and Computer Science

Content

1. How can we improve first-order edge detection?
2. How can we detect edges using second order differentiation/ differencing

Applying Sobel operator

Sobel is a good basic operator

Blurred edges

(a) original image (b) Sobel edge magnitude (c) thresholded magnitude

Noisy edges

Canny edge detection operator

Formulated with three main objectives:

- optimal detection with no spurious responses;
- good localisation with minimal distance between detected and true edge position; and
- single response to eliminate multiple responses to a single edge.

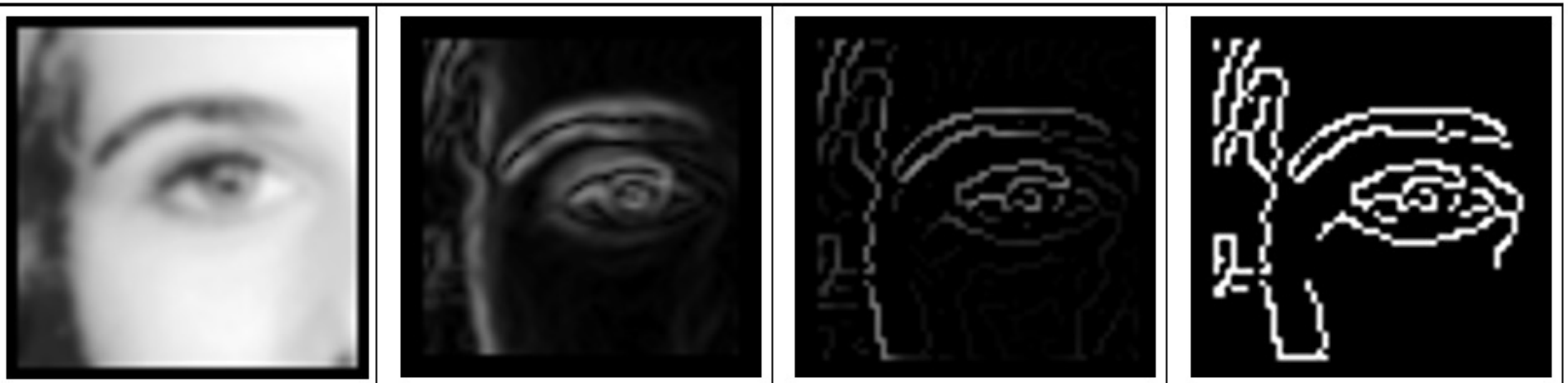
Approximation

1. use Gaussian smoothing;
2. use the Sobel operator;

combine?

3. use **non-maximum suppression**; and
4. threshold with **hysteresis** to connect edge points.

Stages in Canny edge detection operator

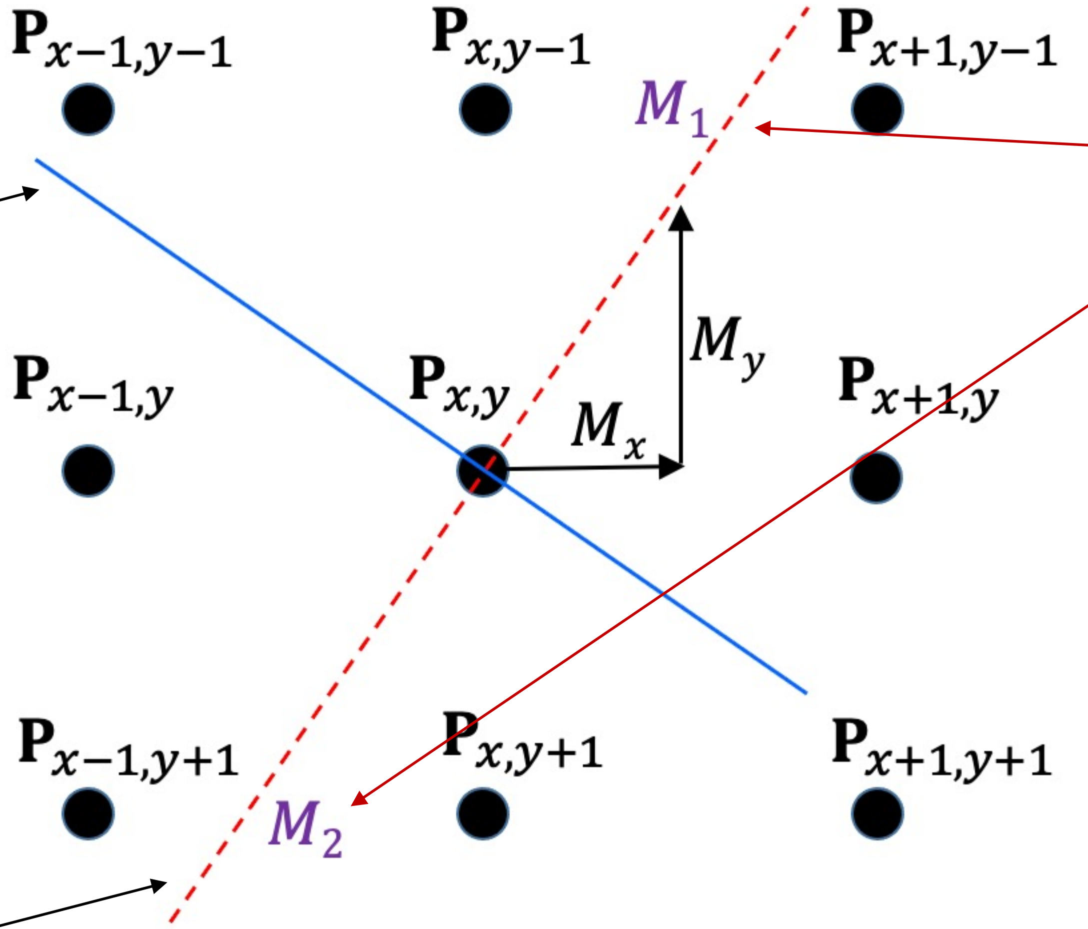


(a) Gaussian smoothing | (b) Sobel edge detection | (c) non-maximum suppression | (d) hysteresis thresholding

Stages in Canny Edge Detection

Canny gives thin edges in the right place, but is more complex

Interpolation in Non-maximum Suppression



Edge Direction at $\mathbf{P}_{x,y}$

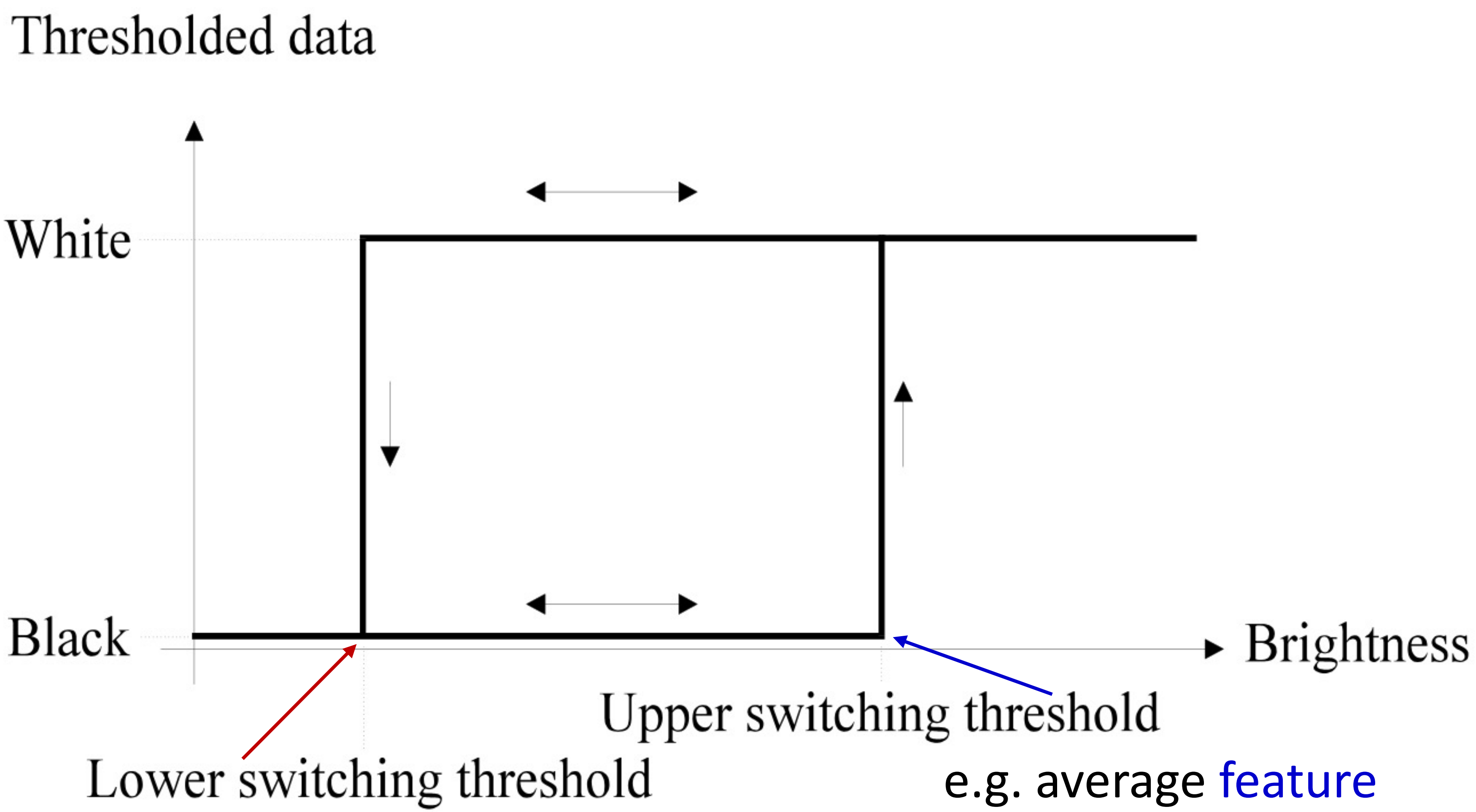
Normal to Edge direction

Need to use points which are not on the image grid

Use linear interpolation

Mark the point $\mathbf{P}_{x,y}$ as a maximum if its gradient magnitude is larger than both at M_1 and M_2; otherwise set it to 0.

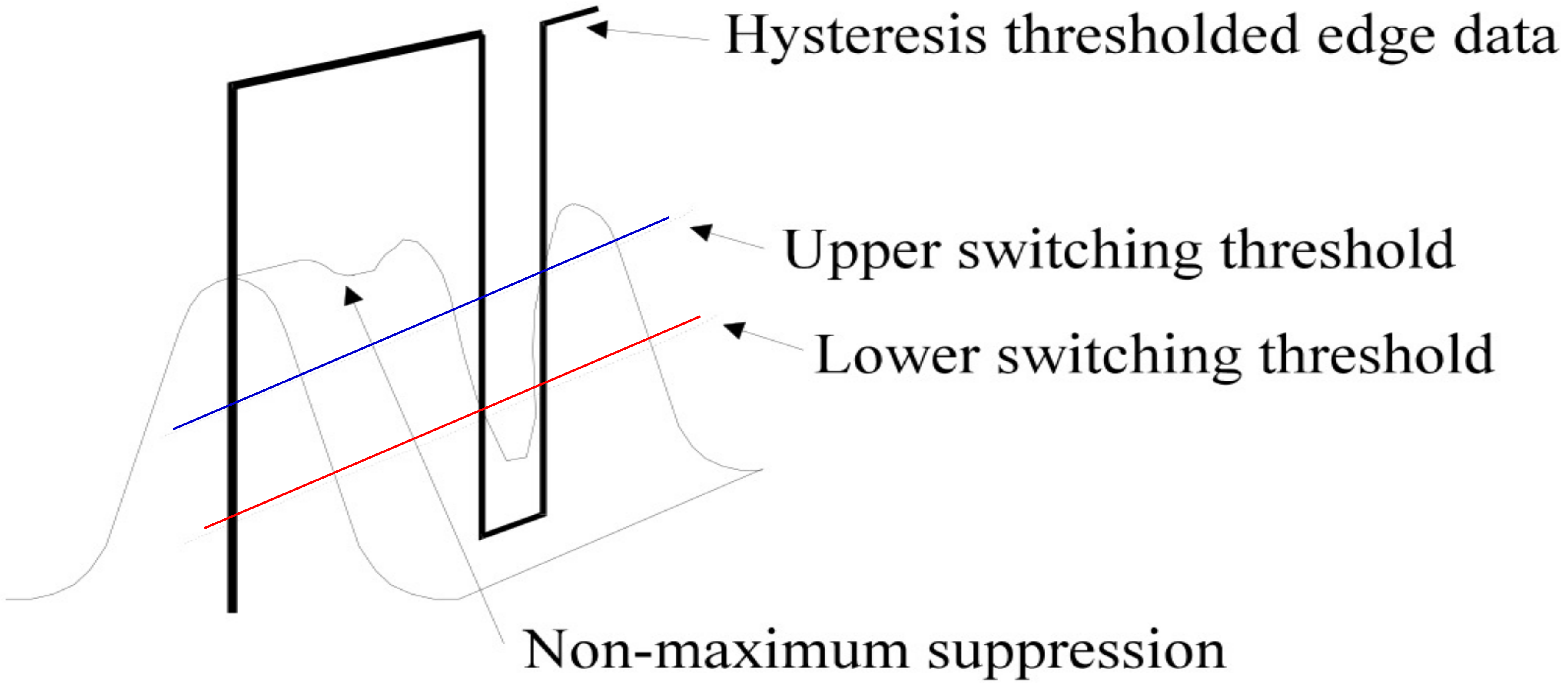
Hysteresis thresholding transfer function



e.g. average noise

e.g. average feature boundary

Action of non-maximum suppression and hysteresis thresholding



Walk along top of ridge

Gives thin edges in the right place

Comparing hysteresis thresholding with uniform thresholding

(a) hysteresis thresholding,
upper level = 40,
lower level = 10

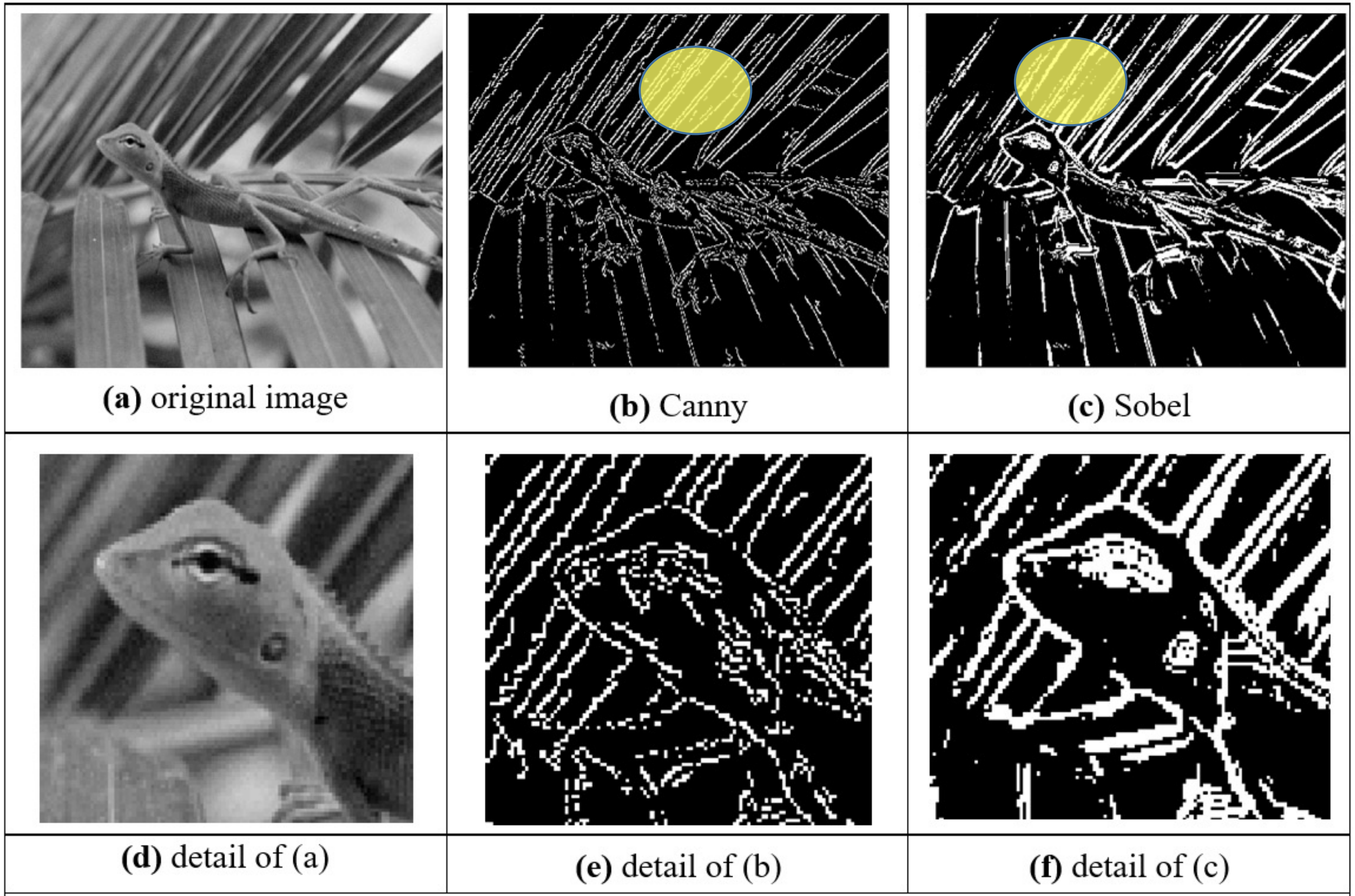
(b) uniform thresholding,
level = 40

(c) uniform thresholding,
level = 10

Hysteresis thresholding gives all points > upper threshold
plus any connected points > lower threshold

Comparing Canny with Sobel

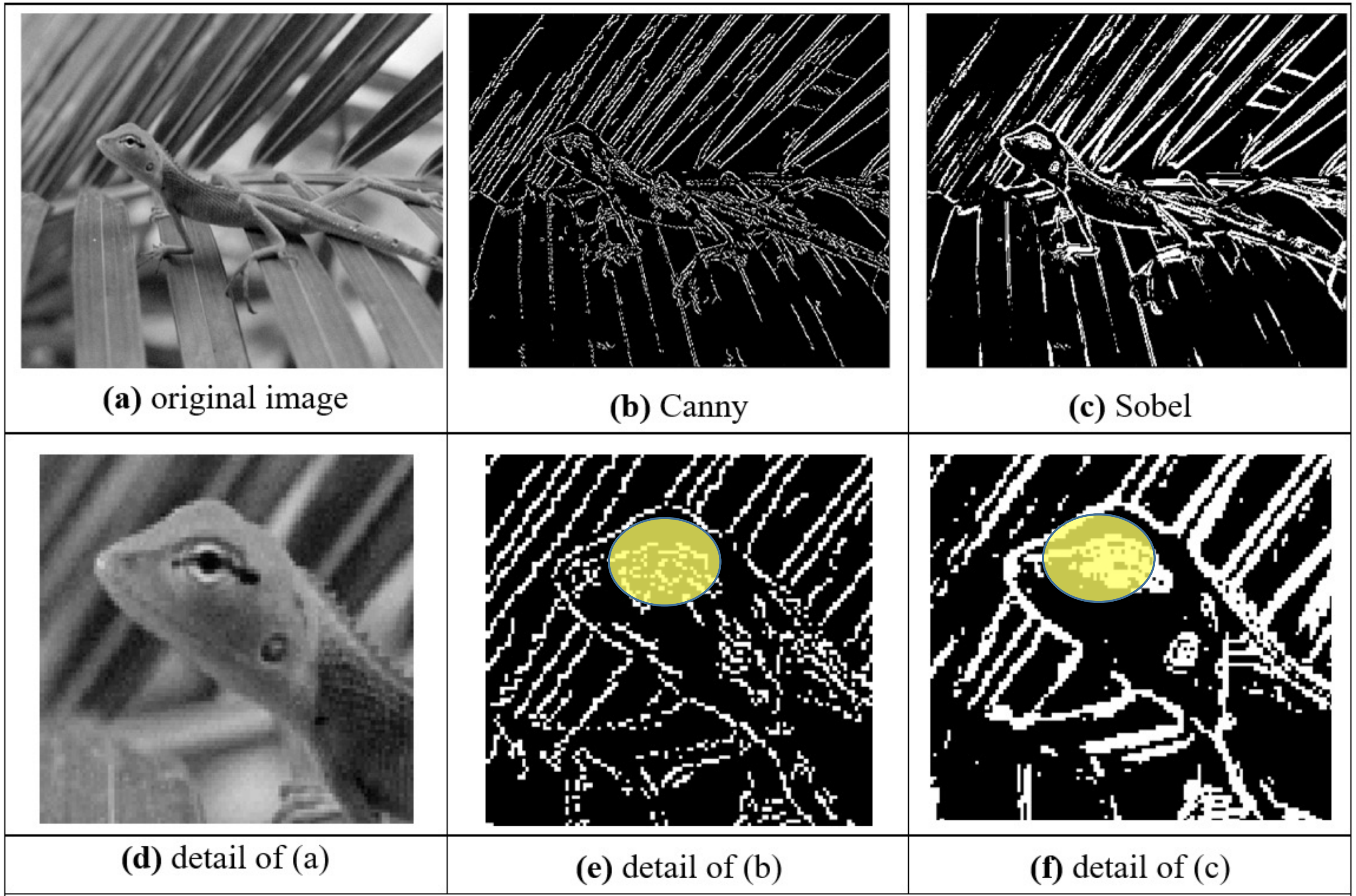
The lines are thinner here, making Sobel look better!



(a) original image (b) Canny (c) Sobel

(d) detail of (a) (e) detail of (b) (f) detail of (c)

Comparing Canny with Sobel



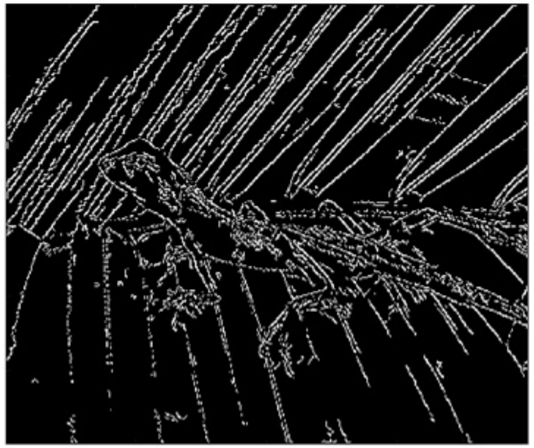
(a) original image (b) Canny (c) Sobel

(d) detail of (a) (e) detail of (b) (f) detail of (c)

The lines are indeed thinner

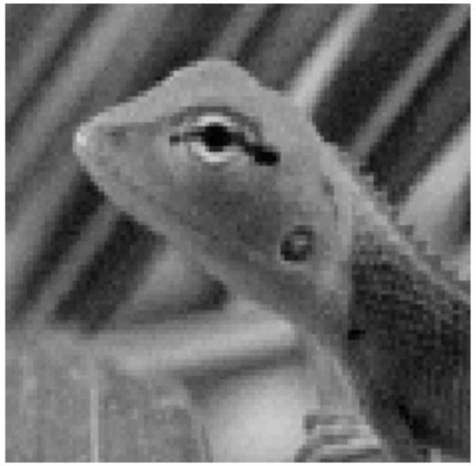
Comparing Canny with Sobel

(a) original image

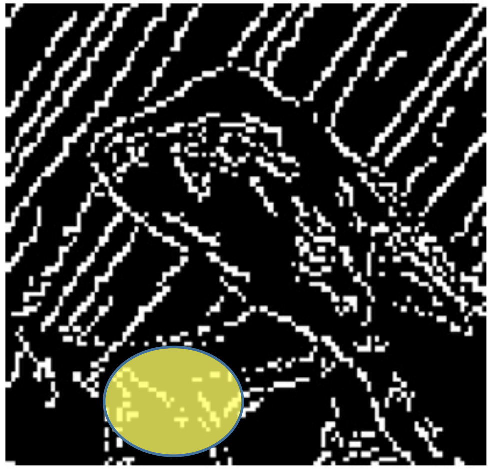


(b) Canny

(c) Sobel



(d) detail of (a)

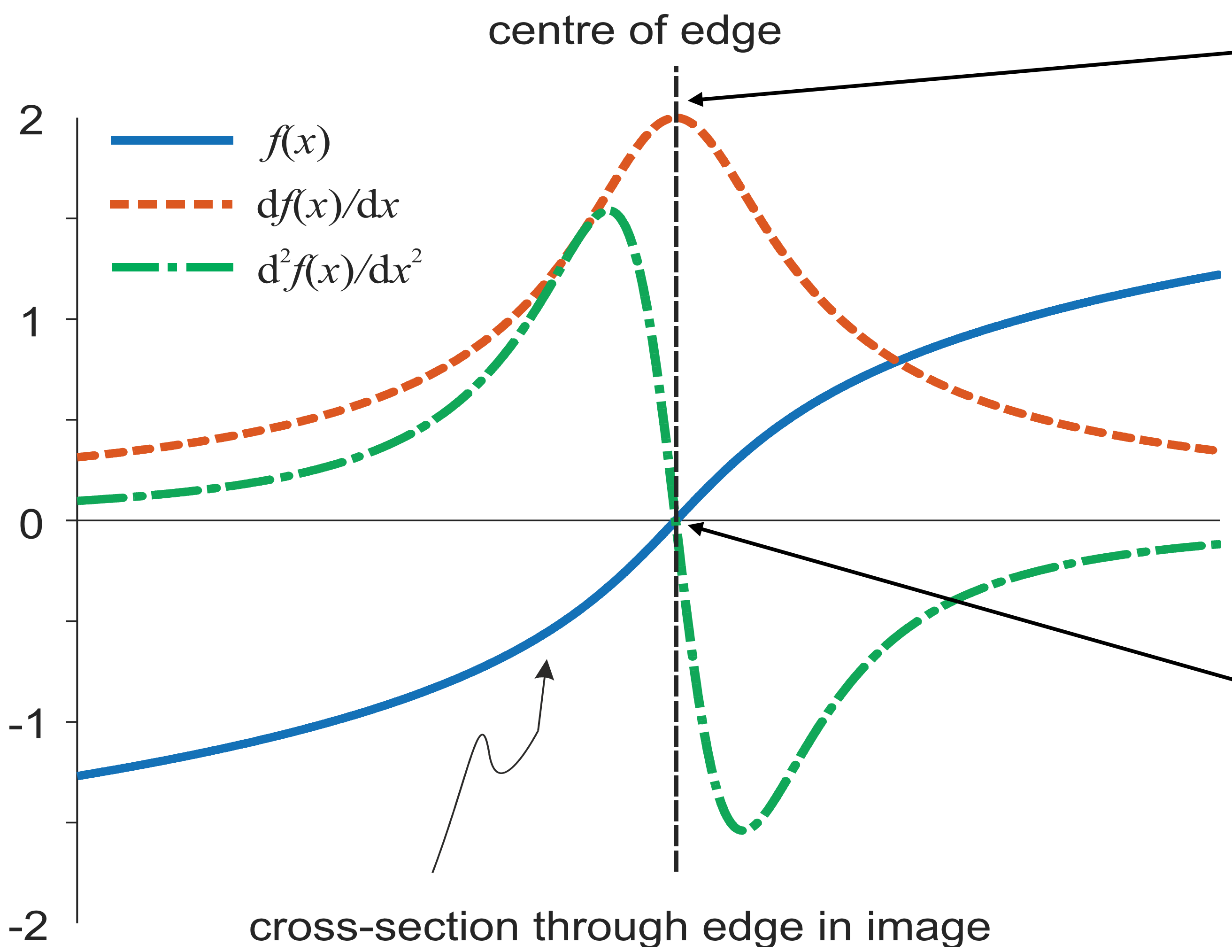


(e) detail of (b)

(f) detail of (c)

The noise is less

First and second order edge detection



First order:
single differentiation
with **thresholding**

Second order:
twice differentiation
with **zero-crossing detection**

Edge detection via the Laplacian operator

0	-1	0
-1	4	-1
0	-1	0

$$f'(x) = (f(x+\Delta x) - f(x)) \,/\, \Delta x$$

$$f'(x+\Delta x) = (f(x+2\Delta x) - f(x+\Delta x)) \,/\, \Delta x$$

$$f''(x+\Delta x) = (f'(x+\Delta x) - f'(x)) \,/\, \Delta x$$

$$= (f(x+2\Delta x) - 2f(x+\Delta x) + f(x)) \,/\, \Delta x$$

1	2	3	4	1	1	2	1
2	2	3	0	1	2	2	1
3	0	38	39	37	36	3	0
4	1	40	44	41	42	2	1
1	2	43	44	40	39	3	1
2	0	39	41	42	40	2	0
0	2	0	2	2	3	1	1
0	2	1	3	1	0	4	2

(a) image data

0	0	0	0	0	0	0	0
0	1	-31	-47	-36	-32	0	0
0	-44	70	37	31	60	-28	0
0	-42	34	12	1	50	-41	0
0	-37	47	8	-6	31	-32	0
0	-45	72	37	45	74	-36	0
0	6	-44	-38	-40	-31	-6	0
0	0	0	0	0	0	0	0

(b) result of the Laplacian operator

Simple, but sensitive to noise!

Edge detection is about differentiation

Gaussian function has the smoothing effect

Can also add a constant: $\frac{1}{2\pi\sigma^2}$

Take a Gaussian function: $g(x, y, \sigma) = e^{\frac{-(x^2+y^2)}{2\sigma^2}}$

Differentiate **once**: $\frac{\partial g(x,y,\sigma)}{\partial x} = -\frac{x}{\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}$

And **again**: $\frac{\partial^2 g(x,y,\sigma)}{\partial x^2} = \left(\frac{x^2}{\sigma^2} - 1\right) \frac{e^{\frac{-(x^2+y^2)}{2\sigma^2}}}{\sigma^2}$

Mathbelts on...

Second order in x and y is:

$$\nabla^2 g(x,y,\sigma) = \frac{\partial^2 g(x,y,\sigma)}{\partial x^2} U_x + \frac{\partial^2 g(x,y,\sigma)}{\partial y^2} U_y$$

$$= \left(\frac{x^2}{\sigma^2} - 1\right) \frac{e^{\frac{-(x^2+y^2)}{2\sigma^2}}}{\sigma^2} + \left(\frac{y^2}{\sigma^2} - 1\right) \frac{e^{\frac{-(x^2+y^2)}{2\sigma^2}}}{\sigma^2}$$

$$= \frac{1}{\sigma^2}\left(\frac{x^2+y^2}{\sigma^2} - 2\right) e^{\frac{-(x^2+y^2)}{\sigma^2}}$$

Second order = Laplacian of Gaussian = Marr Hildreth

Google: “Laplacian of Gaussian”

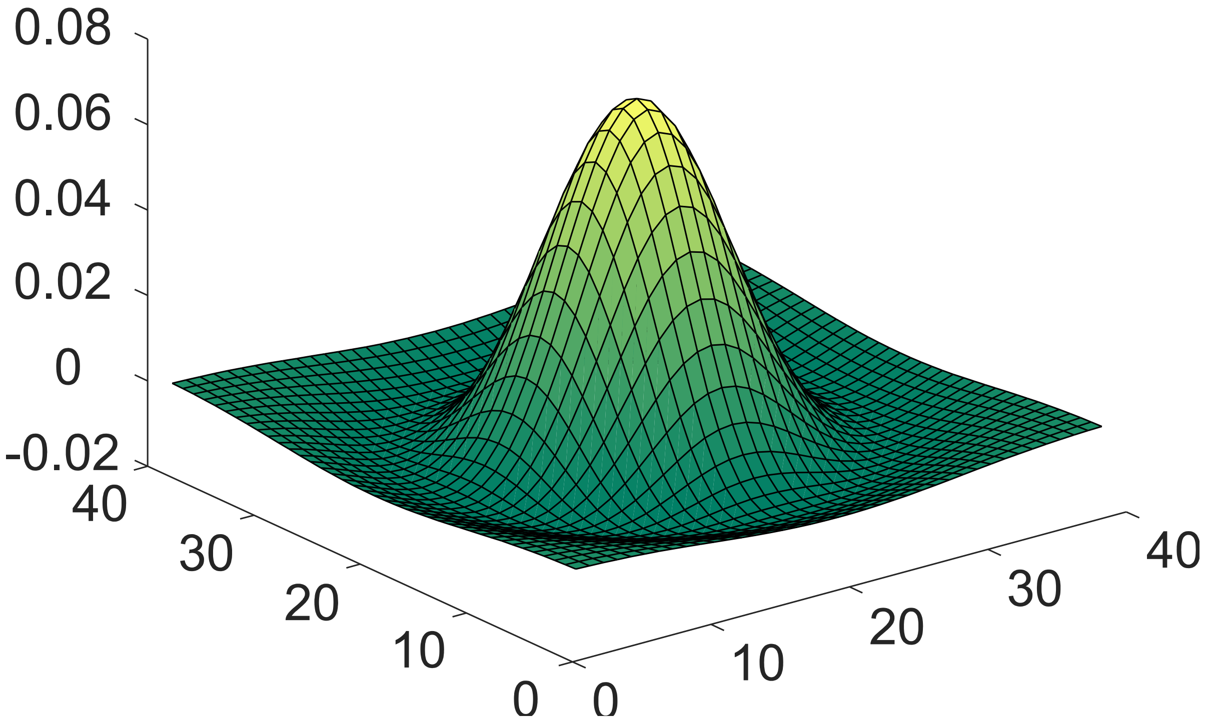
$$LoG \triangleq \Delta G_\sigma(x,y) = \frac{\partial^2}{\partial x^2}G_\sigma(x,y)+\frac{\partial^2}{\partial y^2}G_\sigma(x,y) = \frac{x^2+y^2-2\sigma^2}{\sigma^4}e^{-(x^2+y^2)/2\sigma^2}$$

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

Difference comes from the constant: $\frac{1}{2\pi\sigma^2}$ in front of the Gaussian function.

http://homepages.inf.ed.ac.uk/rbf/HIPR2/log.htm; http://fourier.eng.hmc.edu/e161/lectures/gradient/node8.html ; http://academic.mu.edu/phys/matthysd/web226/Lab02.htm

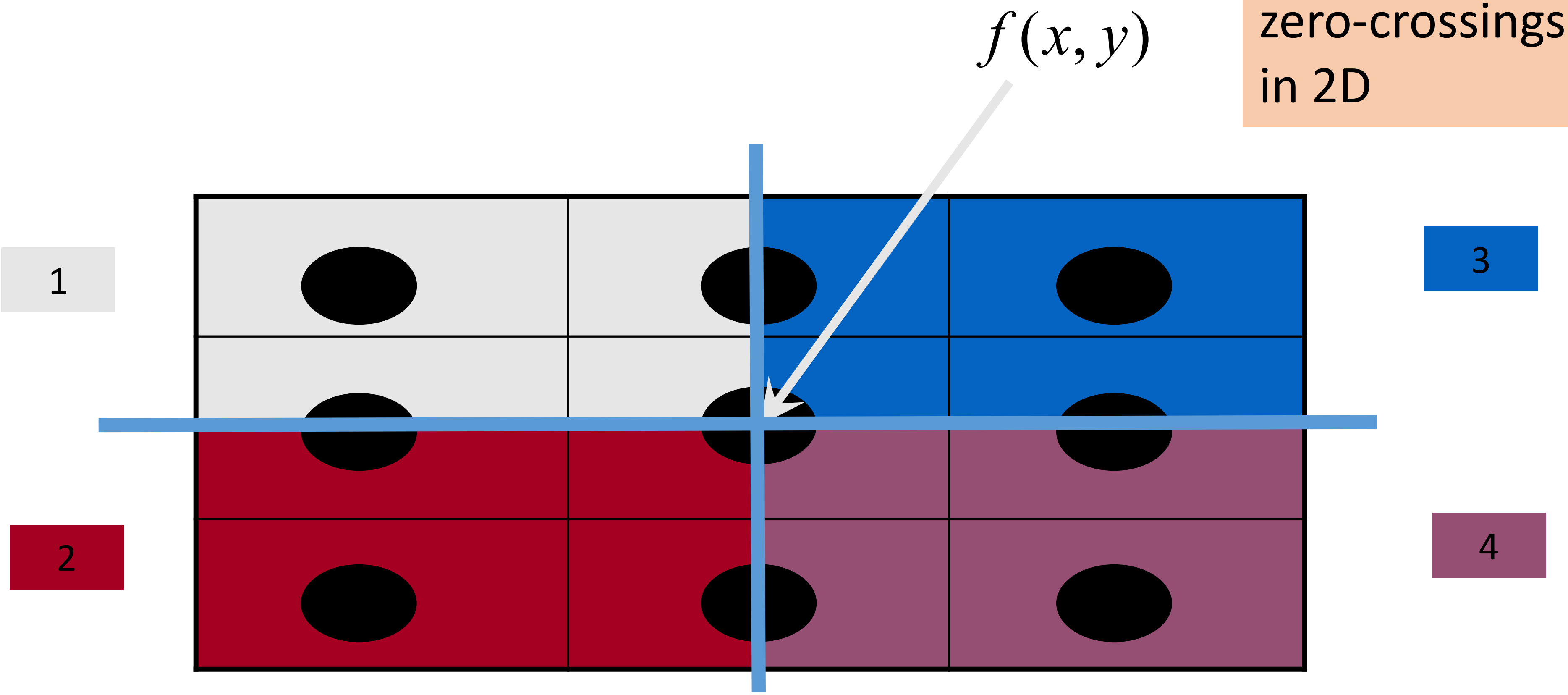
Shape of Laplacian of Gaussian operator



It’s called the ‘Mexican hat operator’

Zero crossing detection

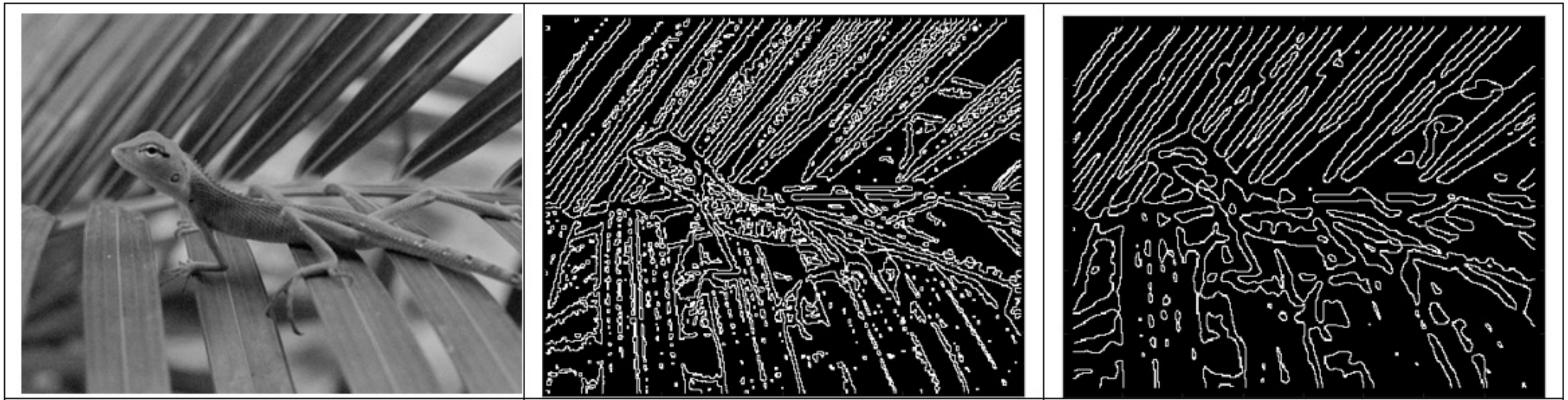
Need to find zero-crossings in 2D



Using e.g.: straight comparison

$$IF\left(\max(1,2,3,4)>0 \wedge \min(1,2,3,4)<0\right) \quad THEN \quad f(x,y)=\text{edge}$$

Marr-Hildreth edge detection

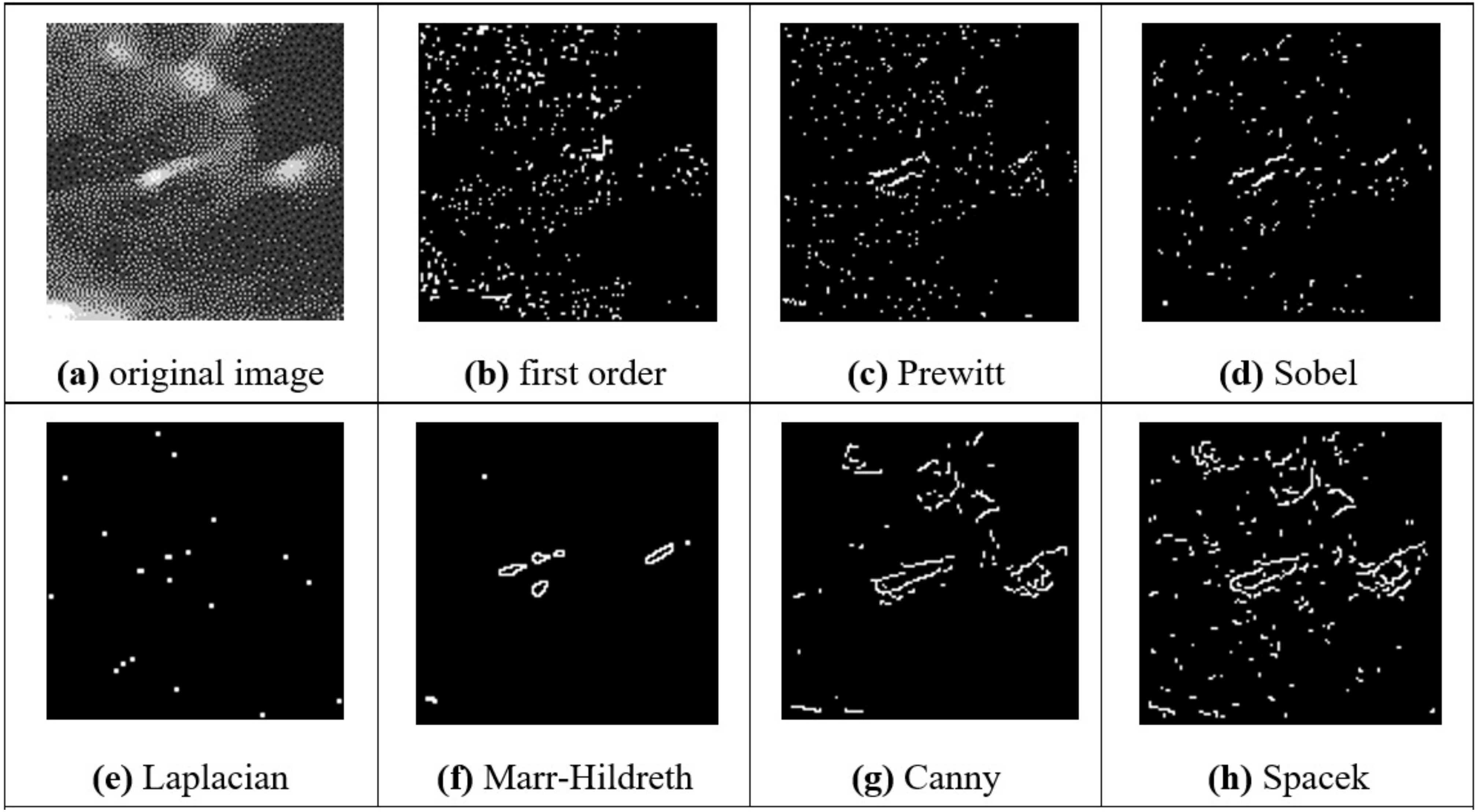


(a) image	**(b)** LoG $(9\times 9; \sigma = 1.4)$	**(c)** LoG $(19\times 19; \sigma = 3)$

Small template, small σ
for local features

Large template, large σ
for global features

Comparison of edge detection operators



Main points so far

1 – Canny provides thin edges in the right place

2 – second order (Marr-Hildreth) requires zero-crossing detection

3 – the results by Marr-Hildreth and Canny are well worth the extra computation

Now we need to collect the edges to find shape.
Coming next…

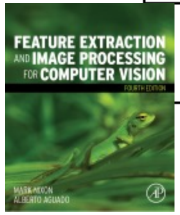
Advanced: Phase Congruency

(a) modified cameraman image

(b) edges by the Canny operator

(c) phase congruency

Figure 4.34 Edge Detection by Canny and by Phase Congruency



Advanced: localised feature extraction

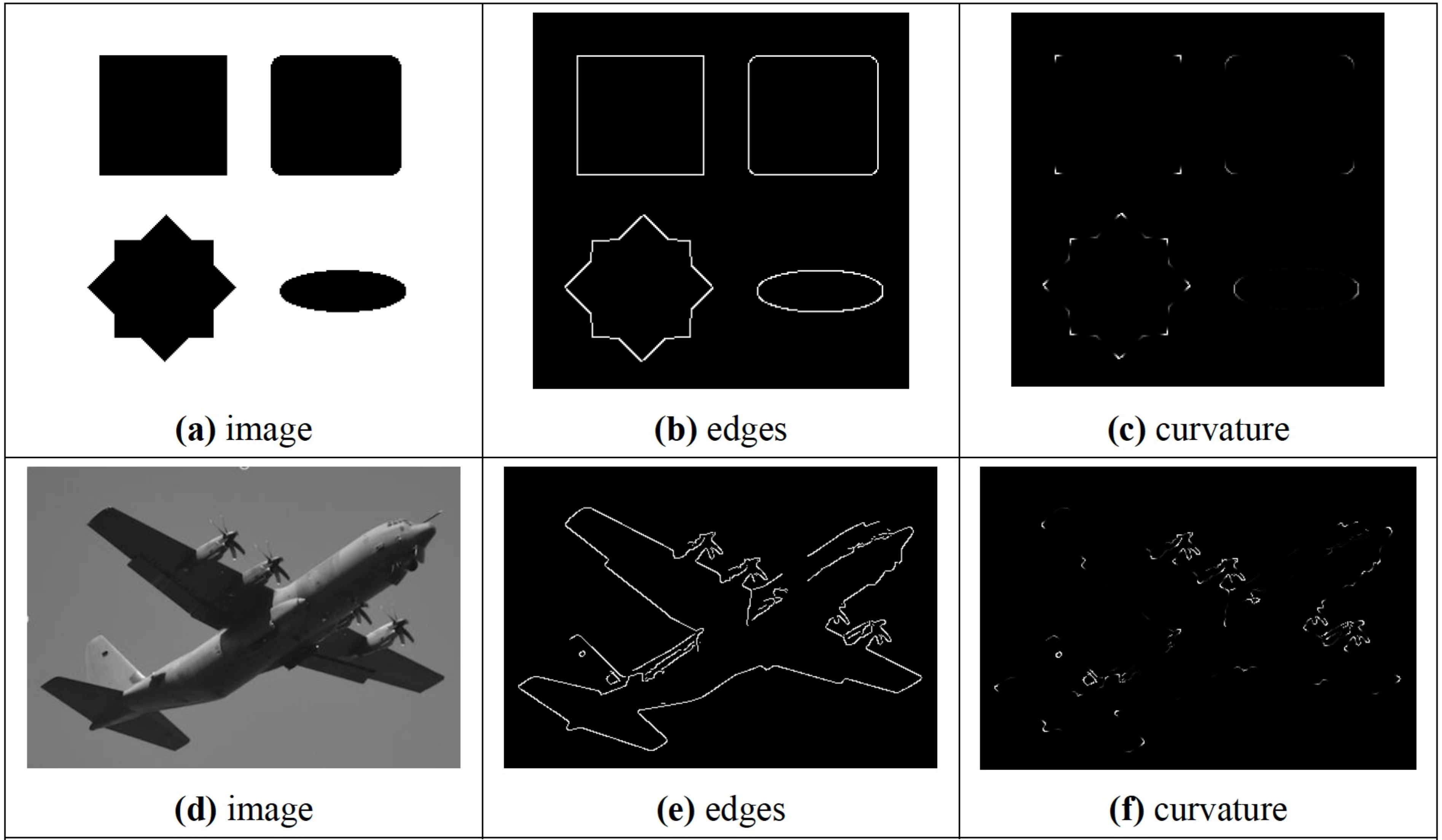
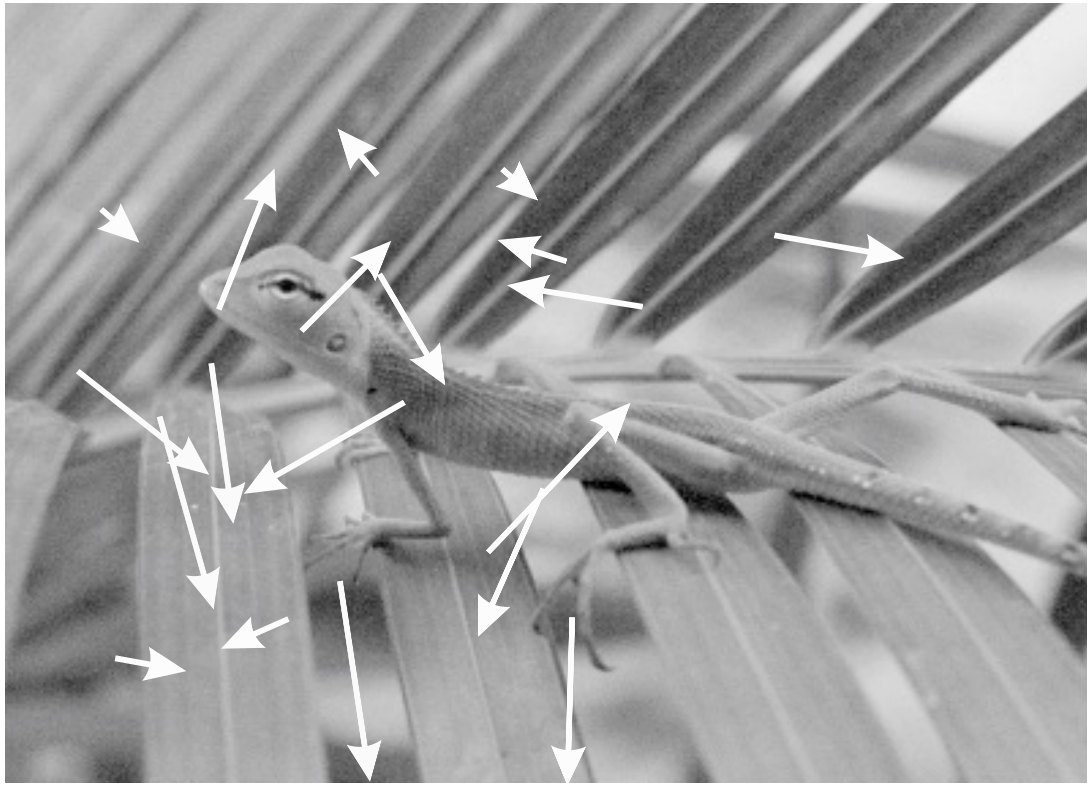


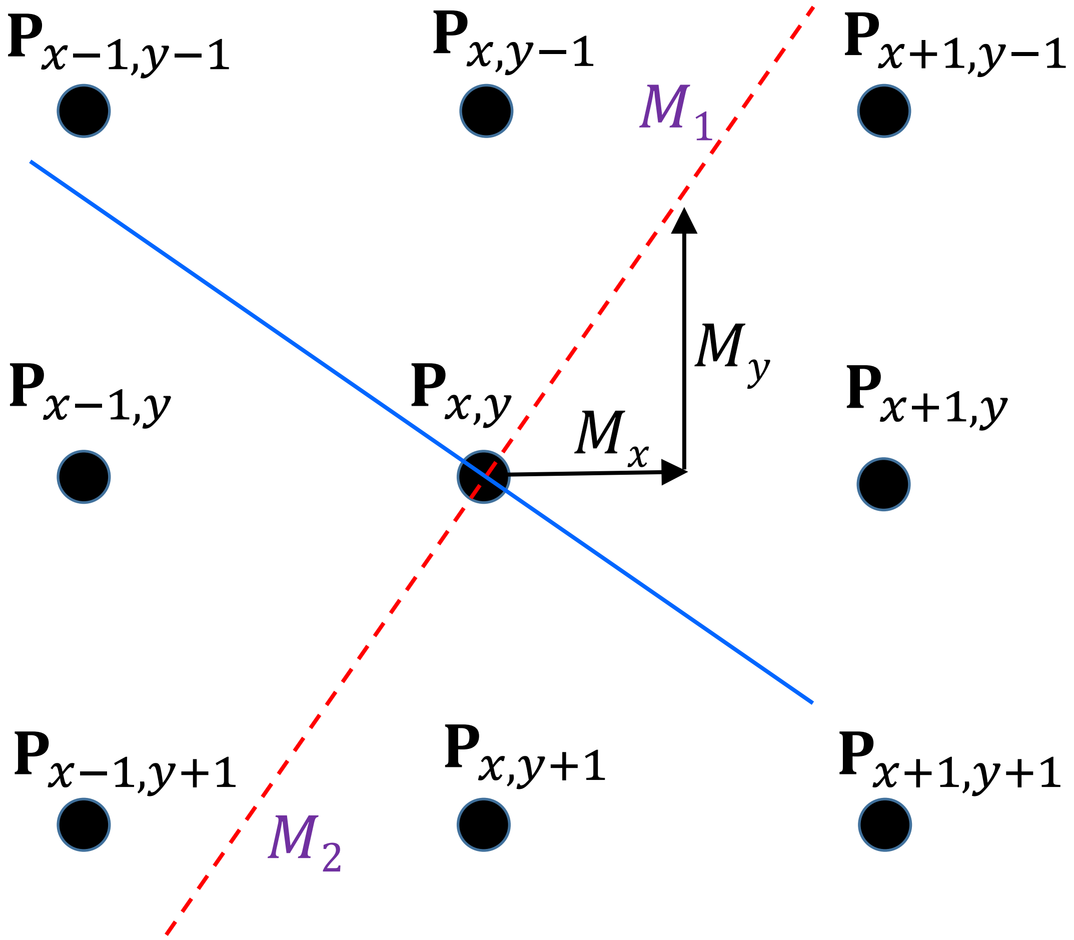
Figure 4.38 Curvature Detection by Angle Differences

Advanced: localised feature extraction



SIFT (mega famous) feature points

Others: SURF, FAST, ORB, FREAK, LOCKY, etc.


$\mathbf{P}_{x-1,y-1}$
$\mathbf{P}_{x,y-1}$
$\mathbf{P}_{x+1,y-1}$
M_1
M_y
$\mathbf{P}_{x-1,y}$
$\mathbf{P}_{x,y}$
M_x
$\mathbf{P}_{x+1,y}$
$\mathbf{P}_{x-1,y+1}$
$\mathbf{P}_{x,y+1}$
$\mathbf{P}_{x+1,y+1}$
M_2