UNIVERSITY OF Southampton
School of Electronics and Computer Science

COMP3204 Computer Vision

Revision – Part 1

Xiaohao Cai

University of Southampton UK

Exam Format

Time: 19th January 2022, 9:30am-12:00am

3 hours duration

Please double check the latest timetable

Format: Choose 3 questions from 6

- Each question worth 33 marks (99 marks total)
- Jon wrote (the last) 3; I wrote (the first) 3
  - Expect that mine will typically cover lower level vision & Jon's will cover higher level vision
  - but there will probably be some overlap!

Typical Mark Scheme

- 80% from lectures and notes; 20% from independent reading
  - Each question will be broken down into sub-parts
  - The marks for each question are there to help you
  - Use them to judge how much to write and how much time to spend

- Questions are more focusing on problem solving and applying your knowledge

- Some mathematical/calculation questions to be expected

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Typical Mark Scheme

few

more

design and explain

4.
Feature morphology and segmentation

(a) ~~**Describe**~~ with the aid of sketches three different morphologies of image feature. [6 marks]

(b) **Describe** two strategies for segmenting an image. [13 marks]

(c) **Compare** the two strategies from the previous answer, highlighting advantages and disadvantages of one with respect to the other. [8 marks]

(d) **Identify** parameters that will need to be set, either manually or automatically for the segmentation methods you described and strategies to choose them. [6 marks]

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Past Exam Papers

UNIVERSITY OF Southampton
School of Electronics and Computer Science

University of Southampton > ECS > Intranet > Modules > 2020-2021 > COMP6223 > **Past Papers**

COMP6223: Computer Vision (MSc) (2020-2021)

Overview | Resources | Past Papers | Syllabus | Evaluation | Send Message | Students | Help

You are a lecturer on this module.

- COMP6223_rubric_201516.pdf
- COMP6223_rubric_201415.pdf
- COMP6223-201617-exam.pdf
- 201516_COMP6223.pdf
- 201415_COMP6223W1 Computer Vision (MSC).pdf

Is this page inaccurate, incomplete or out of date? Do you have a suggestion?

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Questions?

Title	Slides	Handouts	Recording
A Framework for Computer Vision	PDF	N/A	N/A
Lecture 1: Eye and Human Vision	PDF	Animations	link
Lecture 2: Image Formation	PDF	PDF	link
Lecture 3: Image Sampling	PDF	link; link	link
Lecture 4: Point Operators	PDF	N/A	link
Lecture 5: Group Operators	PDF	N/A	link
Lecture 6: Edge Detection	PDF	N/A	link
Lecture 7: Further Edge Detection	PDF	link	link
Lecture 8: Finding Shapes	PDF	N/A	link
Lecture 9: Finding More Shapes	PDF	link	link
Lecture 10: Data-driven & Model-driven Methods in Computer Vision	PDF	N/A	link
Bonus Lecture by Mark (20-21): State of Art/Deep Learning	PDF	N/A	N/A
Bonus Lecture by Mark (20-21): Survey of Gait	PDF	N/A	link

> *Demonstration material (download only): Worksheets (zipped Matlab)*

Don't forget

A Framework for Computer Vision

Grey level image

↓ Spatial gradient operators

..... Edge points .....

↓ Thinning and linking

Lines and boundaries

↓ Feature extraction

Shapes

↓

Organisation

↓ Real world knowledge

Recognition

Tony!
(CBE)

UNIVERSITY OF Southampton
School of Electronics and Computer Science

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Lecture 1 Eye and Human Vision

COMP3204 Computer Vision

Is human vision a good model for computer vision?

Book pp 1-28

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Lecture 2 Image Formation

Lecture 3 Image Sampling

COMP3204 Computer Vision

Book pp
29-66

Effects of differing image resolution

(a) 64×64 (b) 128×128 (c) 256×256

Low resolution lose information but *N* by *N* points implies much storage

How do we choose an appropriate value for *N*?

Step up Fourier…

A time−variant signal

j: the complex number $j = \sqrt{-1}$

Fourier transform of signal p

at angular frequency ω

$$Fp(\omega) = \int_{-\infty}^{\infty} p(t)\, e^{-j\omega t}\, dt$$

$$e^{-j\omega t} = \cos(\omega t) - j\sin(\omega t)$$

ω: angular frequency;

ω = 2 π ξ, where frequency ξ is $1/t$

Inverse Fourier…

Fourier coefficients

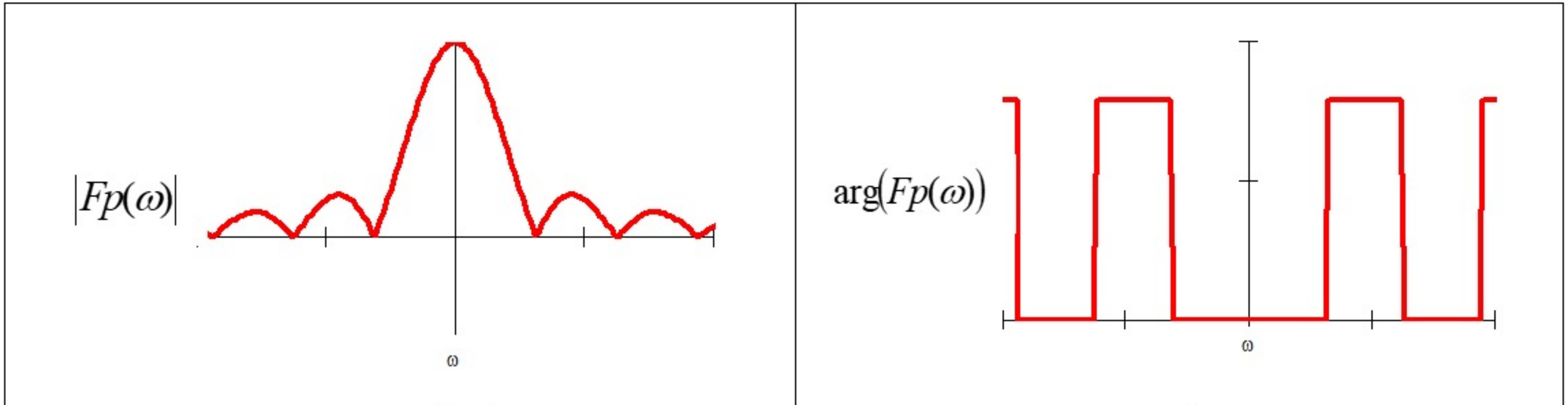
Original signal in time domain

$$p(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} Fp(\omega)\, e^{j\omega t}\, d\omega$$

$$e^{j\omega t} = \cos(\omega t) + j\sin(\omega t)$$

Magnitude and phase of Fourier transform of a pulse

$$Fp(\omega) = \int_{-\infty}^{\infty} p(t) e^{-j\omega t} dt = \mathrm{Re}\left(Fp(\omega)\right) + j\,\mathrm{Im}\left(Fp(\omega)\right)$$

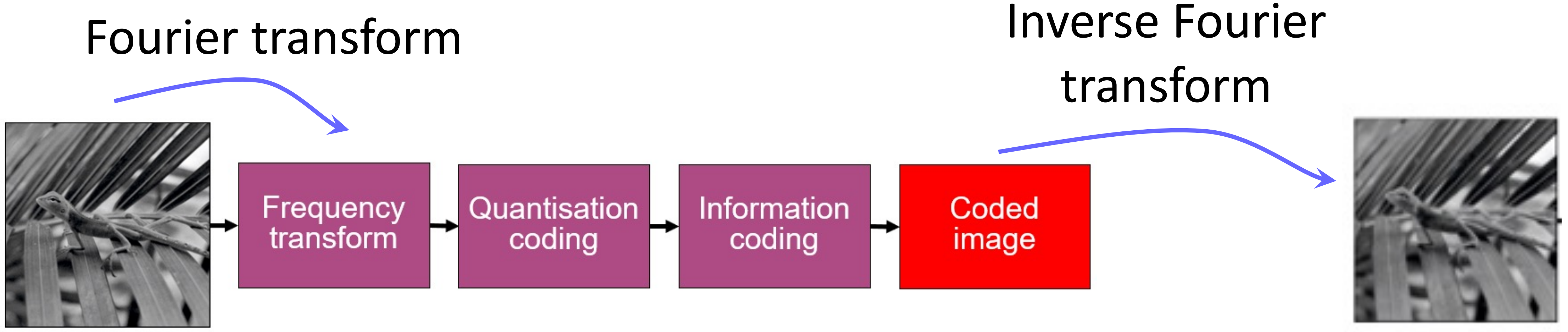


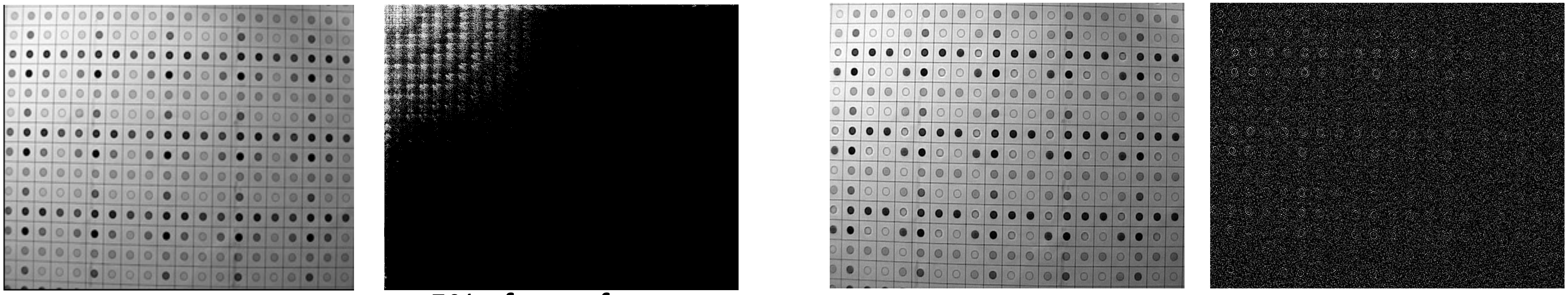
(a) magnitude **(b)** phase

$$|Fp(\omega)| = \sqrt{\mathrm{Re}\left(Fp(\omega)\right)^2 + \mathrm{Im}\left(Fp(\omega)\right)^2}$$

$$\arg\left(Fp(\omega)\right) = \tan^{-1}\left(\frac{\mathrm{Im}\left(Fp(\omega)\right)}{\mathrm{Re}\left(Fp(\omega)\right)}\right)$$

Inverse Fourier transform is used for reconstruction





image

5% of transform components

reconstructed image

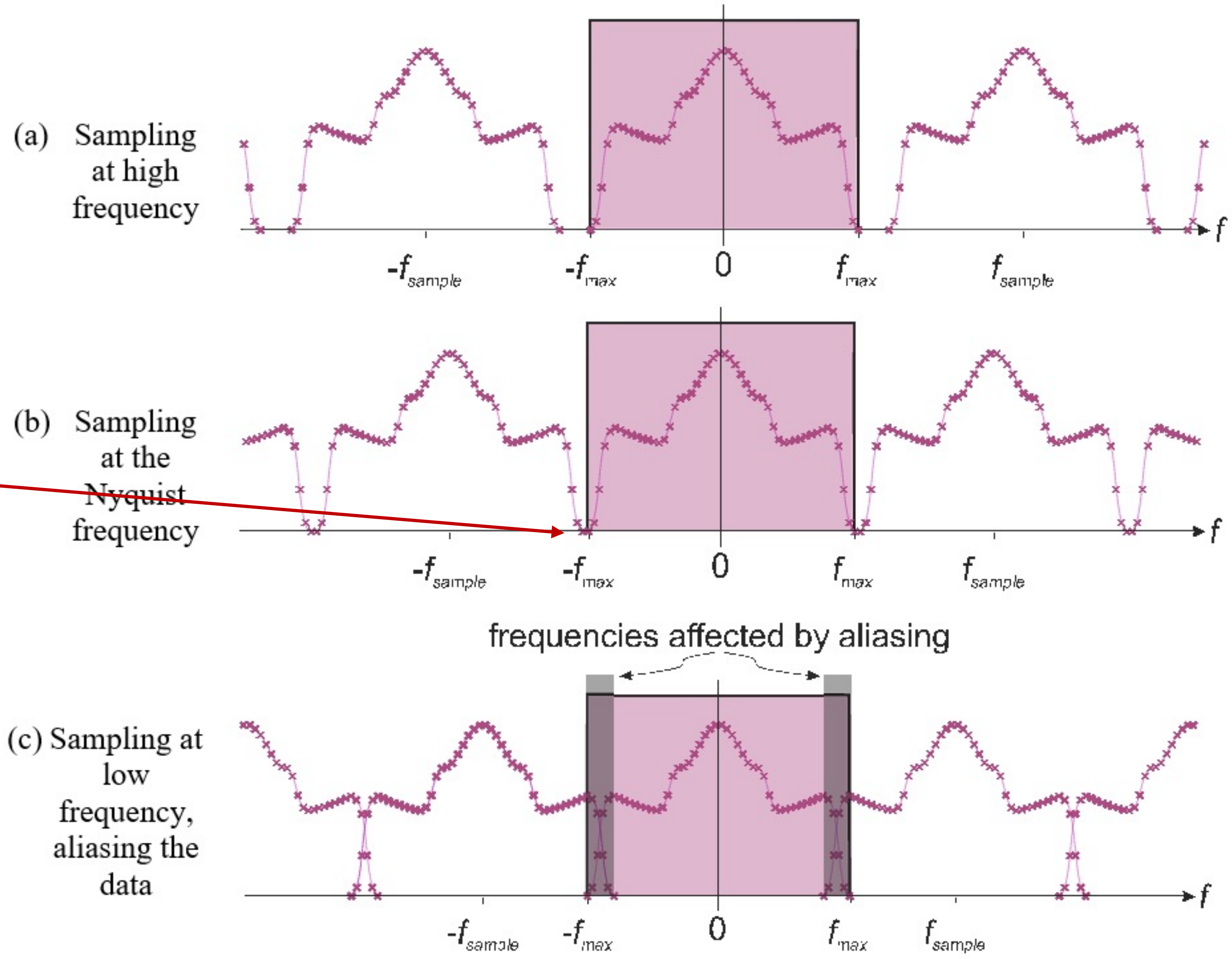
error

In the frequency domain

Spectra repeat

If sampling is just right, spectra just touch

Minimum sampling frequency = 2 × max



Sampling process in the frequency domain

UNIVERSITY OF Southampton
School of Electronics
and Computer Science

Lecture 4 Point Operators
Lecture 5 Group Operators

COMP3204 Computer Vision

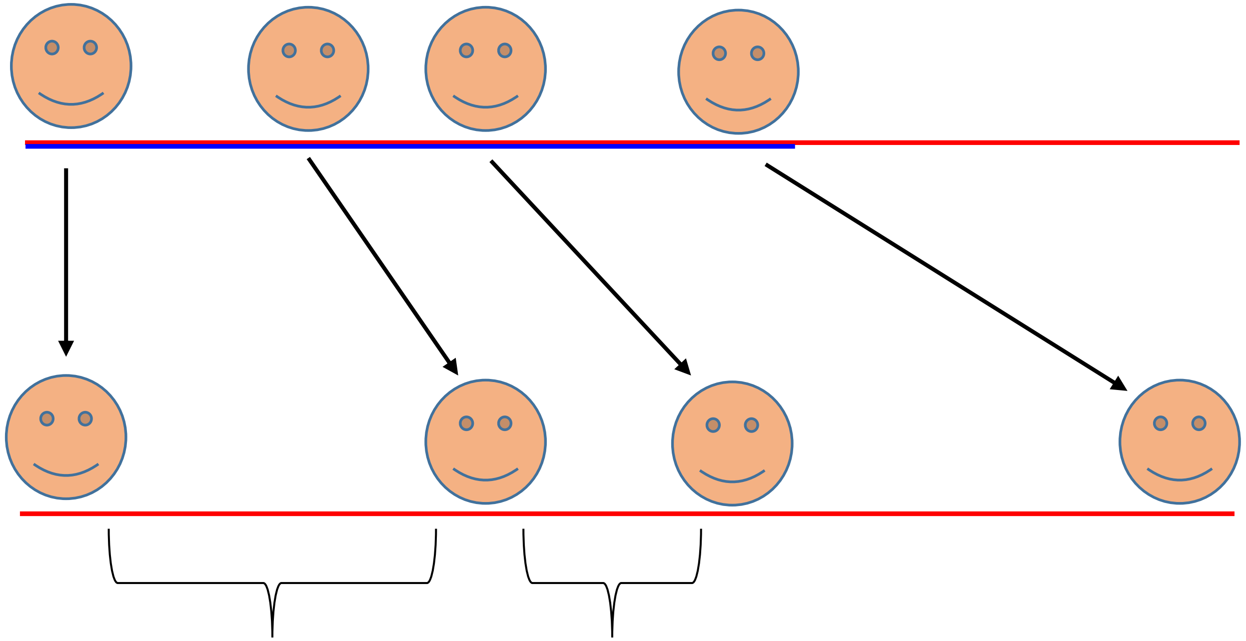
Book
pp
72-114

Intensity normalisation

Toy example

Before normalization:

After normalization
(whole range used):



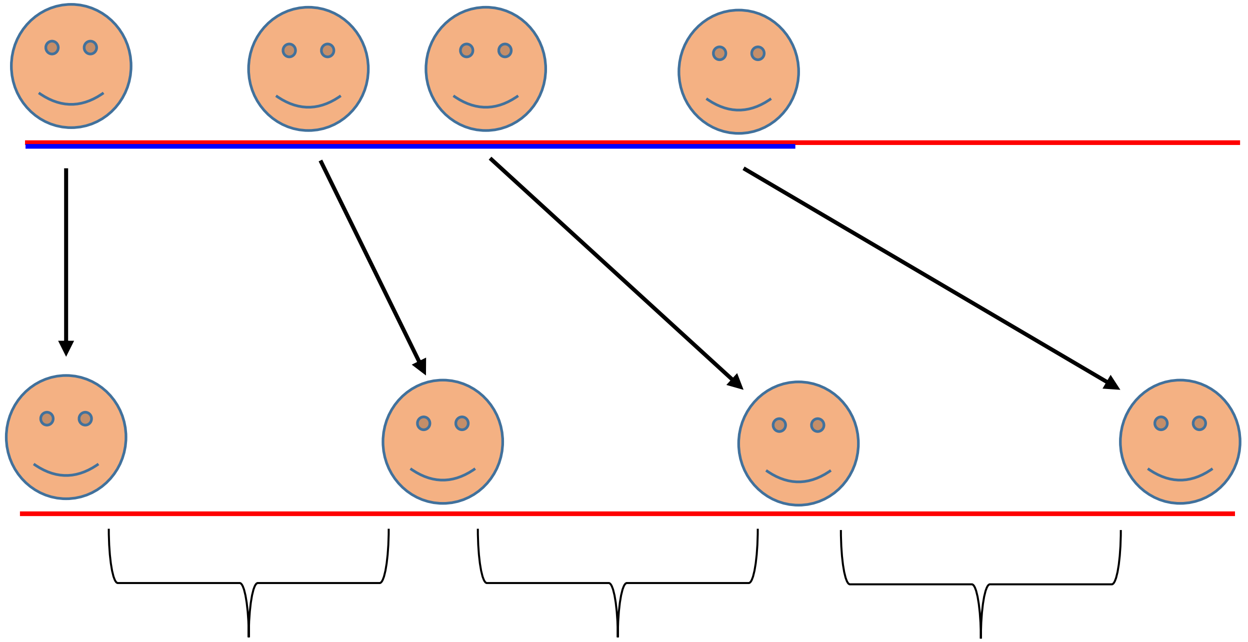
The ratio between the distances of these smileys is kept

Histogram Equalisation

Toy example

Before normalization:

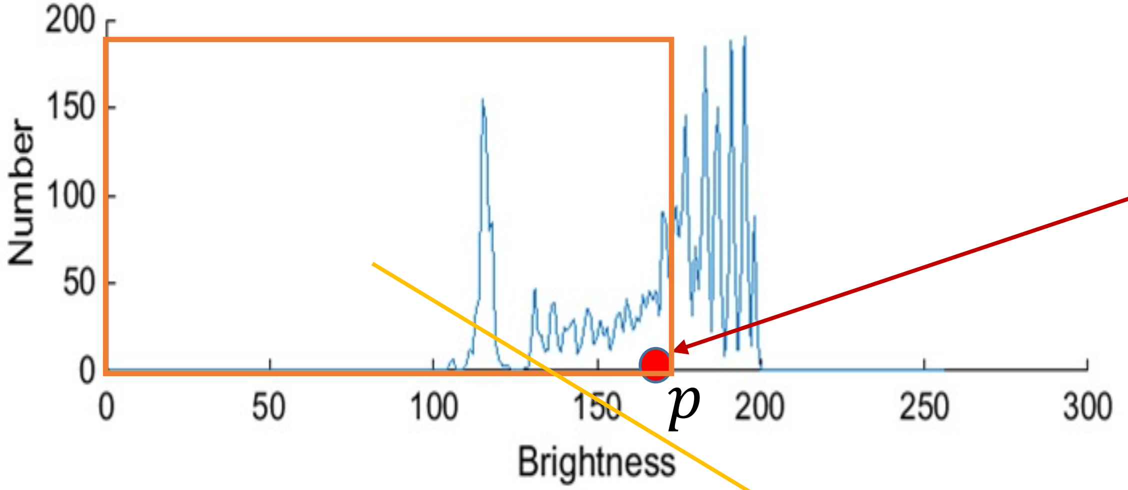
After equalisation
(whole range used):



Same distance between
two adjacent smileys

Histogram Equalisation – aim is a flat histogram

Original histogram



N^2 points in the original and equalised images

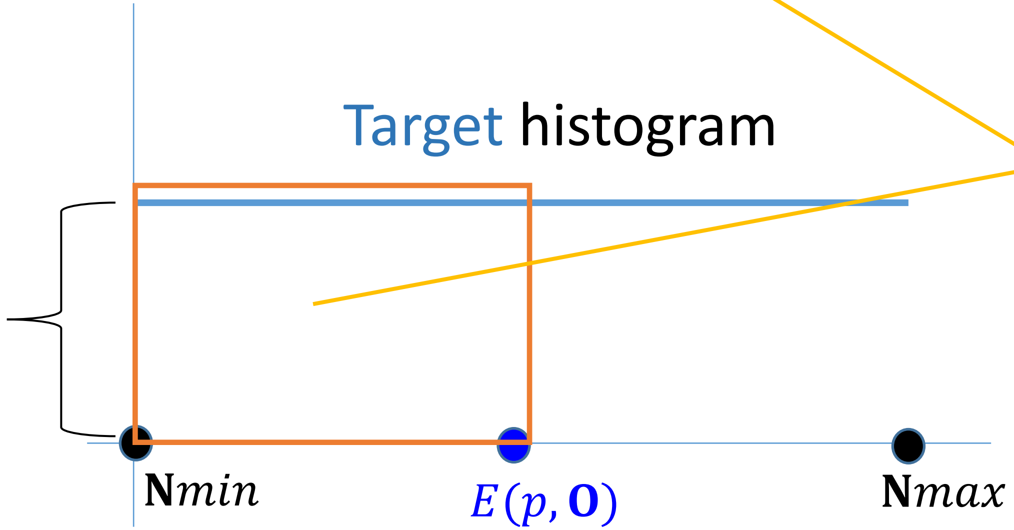
$\mathbf{O}(p)$: the histogram at level p

$\sum_{l=0}^{p} \mathbf{O}(l)$: the cumulative histogram up to level p

$E(p, \mathbf{O})$: the level after equalisation

$$\frac{N^2}{\mathbf{N}max - \mathbf{N}min}$$

Target histogram



$$\frac{N^2}{\mathbf{N}max - \mathbf{N}min}(E(p, \mathbf{O}) - \mathbf{N}min) = \sum_{l=0}^{p} \mathbf{O}(l)$$

$$E(p, \mathbf{O}) = \frac{\mathbf{N}max - \mathbf{N}min}{N^2}\sum_{l=0}^{p} \mathbf{O}(l) + \mathbf{N}min$$

Thresholding

Thresholding selects points that exceed a chosen threshold τ

$$\mathbf{N}_{x,y} = \begin{cases} 255, & \text{if } \mathbf{O}_{x,y} > \tau \\ 0, & \text{otherwise} \end{cases}$$

coordinates

An eye image

After thresholding

3×3 template and its inverse

Original template

w_{00}	w_{01}	w_{02}
w_{10}	w_{11}	w_{12}
w_{20}	w_{21}	w_{22}

Its inverse

w_{22}	w_{21}	w_{20}
w_{12}	w_{11}	w_{10}
w_{02}	w_{01}	w_{00}

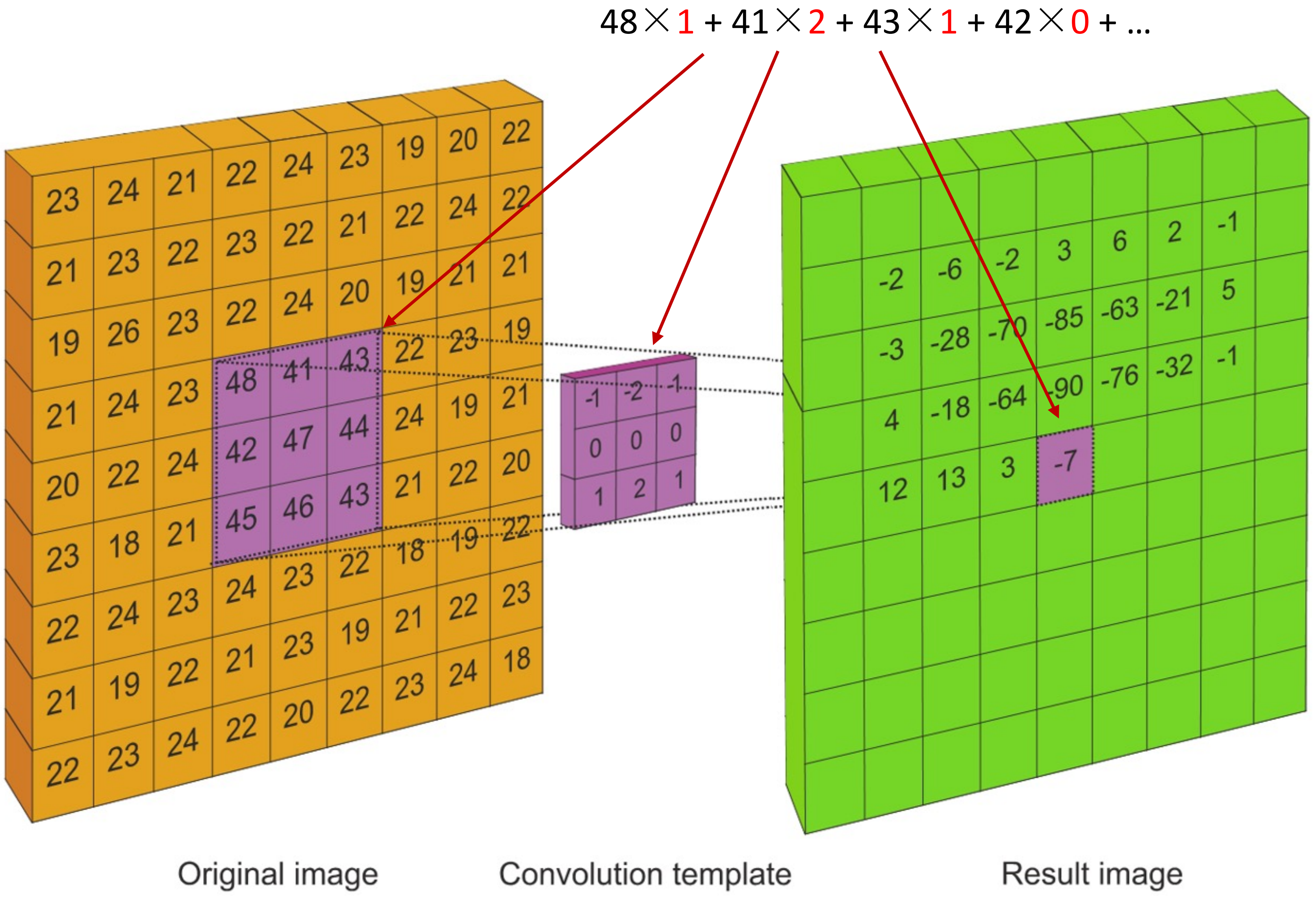
Flip corresponding to
both directions

Template convolution

Calculate a new image from the original

Template is inverted for convolution

Template is convolved in a raster fashion



Template convolution via the Fourier transform

Convolution theorem allows for **fast** computation via FFT for template size ≥ 7×7

Fourier transform of the picture, $\Im(\mathbf{P})$

Template convolution $*$

Fourier transform of the template, $\Im(\mathbf{T})$

$$\mathbf{P} * \mathbf{T} = \Im^{-1}\left(\Im(\mathbf{P}) .\times \Im(\mathbf{T})\right)$$

Picture and the template

Inverse Fourier

Point by point multiplication

The inversion is implicit in Fourier
The theory is at end, for information only

Border?

Options:

- Set border to black

- Padding the original image
  1. constant
  2. wrap-around (periodic or cyclic)
  3. symmetric

- Make template smaller near edges

Normally we assume object of interest is near centre so set border to black

Question 2.

UNIVERSITY OF Southampton
School of Electronics
and Computer Science

(a) **Explain** the difference between the spatial domain (that the above image is in) and the frequency domain after applying the Fourier transform on the above image. **List** some advantages and disadvantages of analysing the image in its frequency domain.

[8 marks]

(b) Suppose you are not satisfied with the noise present in the image above. **Design** a system which can remove the noise using the *frequency* information.

[4 marks]

(c) Suppose the noise type in the above image is Gaussian. **Design** a noise removal filter which works in the *spatial domain* of the image. When the filter size is large, **design** a way to improve the noise removal efficiency, and **explain** why this works.

[12 marks]

(d) **Explain** how the image boundaries impact the size of the image with noise removed in the noise removal techniques you developed in (c).

[9 marks]

UNIVERSITY OF Southampton
School of Electronics
and Computer Science

Lecture 6 Edge Detection

Lecture 7 Further Edge Detection

COMP3204 Computer Vision

Book
pp
118 -
171

Templates for Sobel operator

Sobel is **most popular basic operator**
Double the centre coefficients of Prewitt

1	0	-1
2	0	-2
1	0	-1

(a) Mx

1	2	1
0	0	0
-1	-2	-1

(b) My

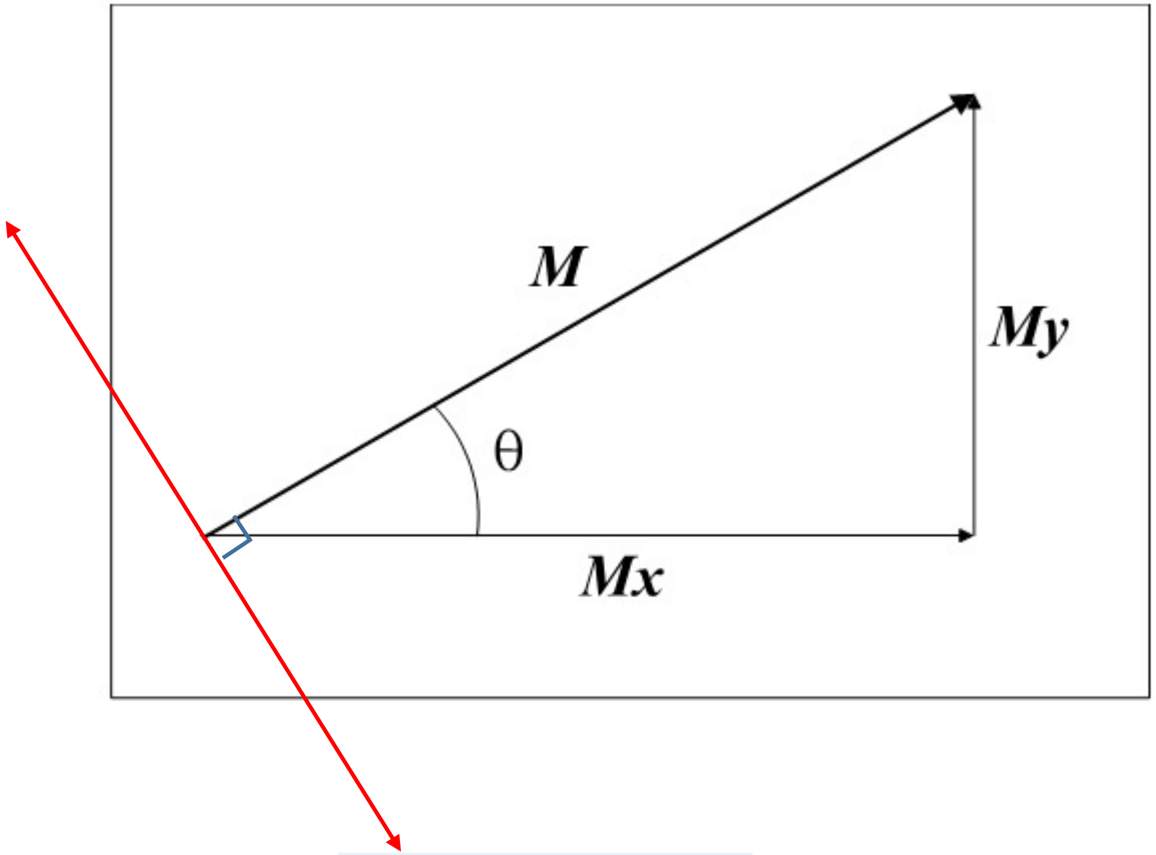
WHY?

Edge Detection in Vector Format

Vectors have magnitude (strength) and direction

Magnitude: $M = \sqrt{M_x^2 + M_y^2}$

Direction: $\theta = \tan^{-1}\left(\frac{M_y}{M_x}\right)$



Edge direction

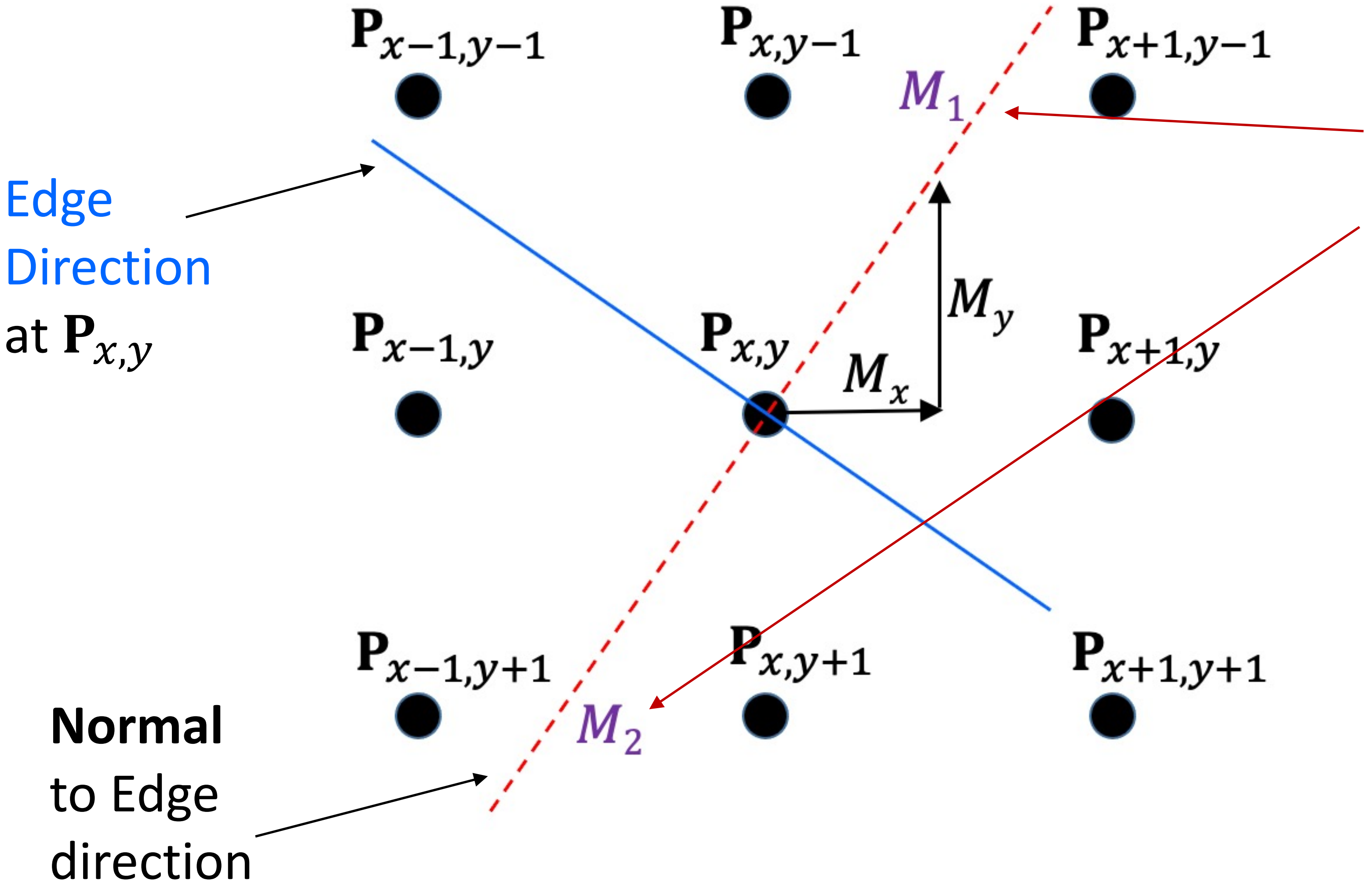
Stages in Canny edge detection operator

(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

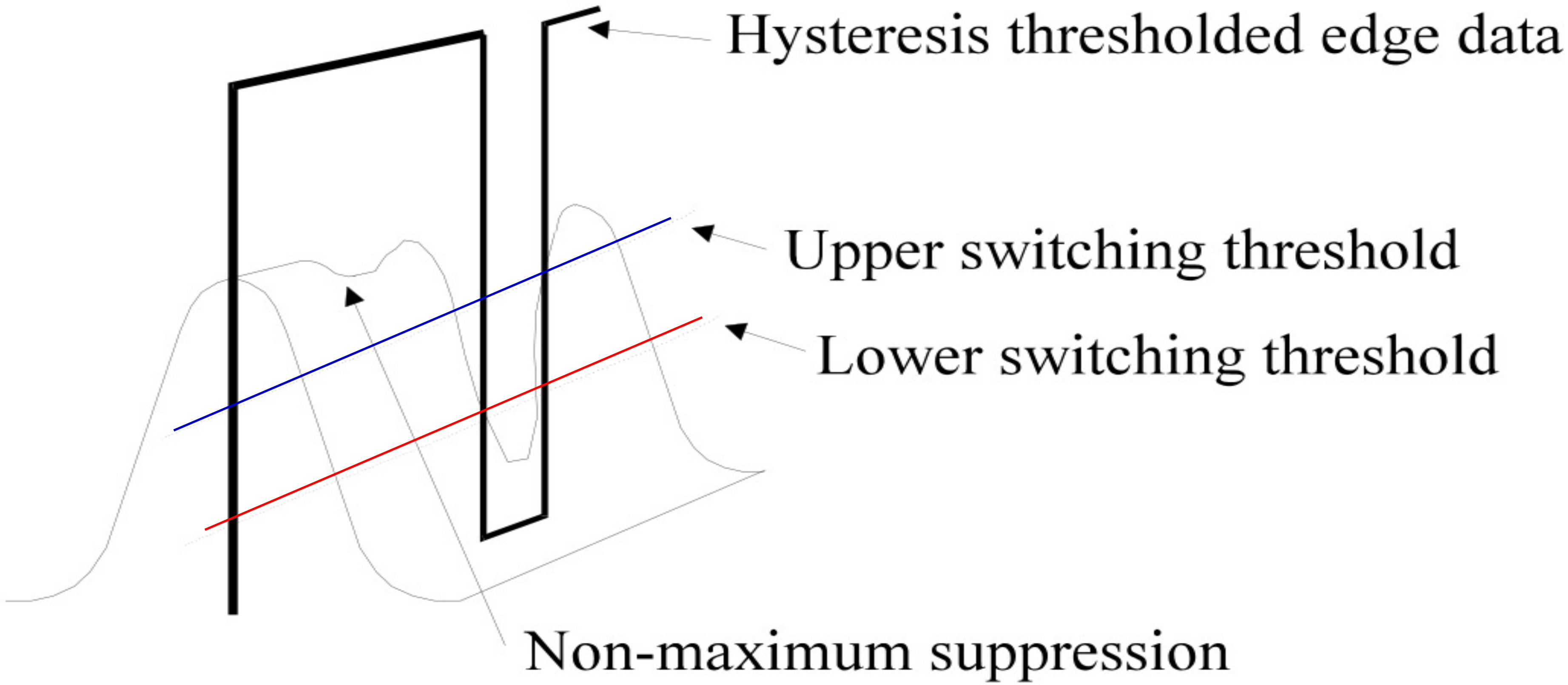


Need to use points which are not on the image grid

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.

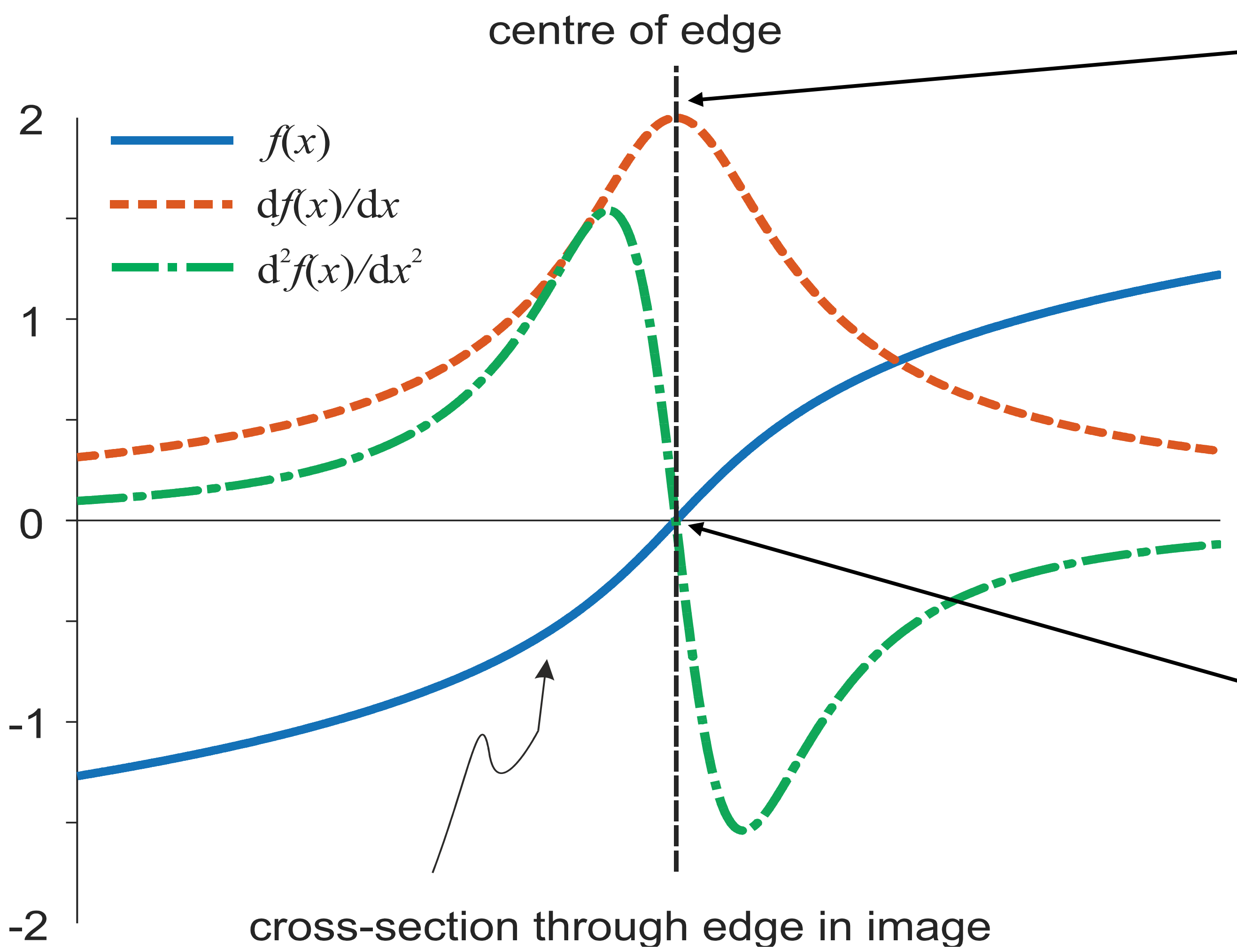
Action of non-maximum suppression and hysteresis thresholding



Walk along top of ridge

Gives thin edges in the right place

First and second order edge detection



First order:
single differentiation
with **thresholding**

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

Google: “Laplacian of Gaussian”

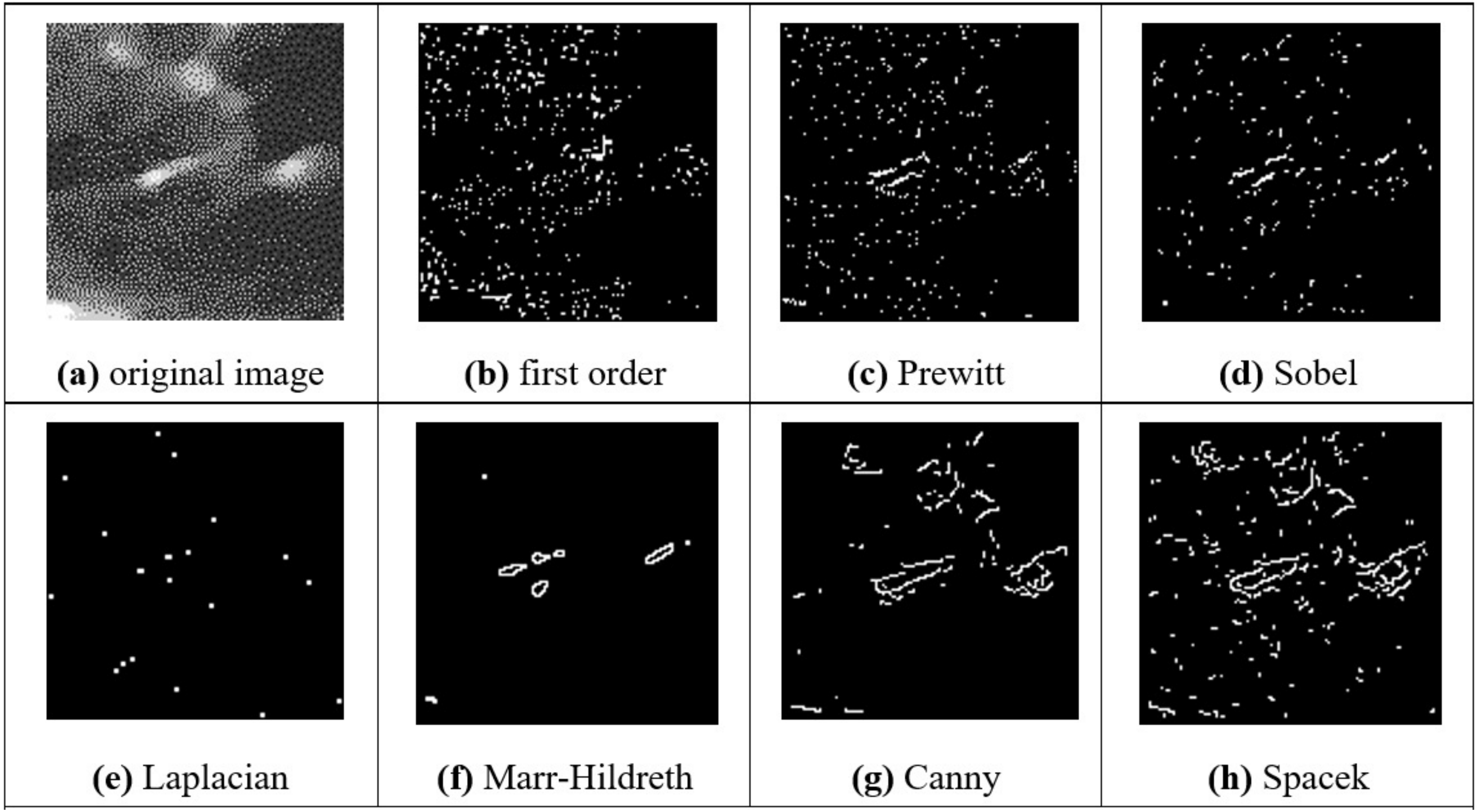
$$LoG \triangleq \Delta G_\sigma(x,y) = \frac{\partial^2}{\partial x^2}G_\sigma(x,y)+\frac{\partial^2}{\partial y^2}G_\sigma(x,y) = \frac{x^2+y^2-2\sigma^2}{\sigma^4}e^{-(x^2+y^2)/2\sigma^2}$$

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

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

Comparison of edge detection operators



(a) original image
(b) first order
(c) Prewitt
(d) Sobel
(e) Laplacian
(f) Marr-Hildreth
(g) Canny
(h) Spacek

University of Southampton
School of Electronics and Computer Science

Lecture 8 Finding Shapes
Lecture 9 Finding More Shapes

COMP3204 Computer Vision

Book
pp
187-201;
208-246

Template Matching - basis

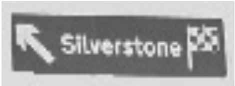
$$\| \mathbf{I}_{\Omega} - \mathbf{T} \|^{2}$$

Process of template matching

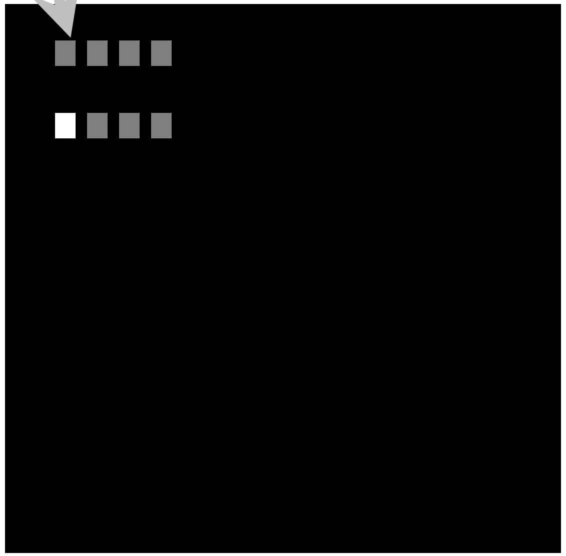
count of matching points

$\boldsymbol{\Omega}$

Image $\mathbf{I}$



Template $\mathbf{T}$



accumulator space

Suggestions for improving the process?

Use edges!

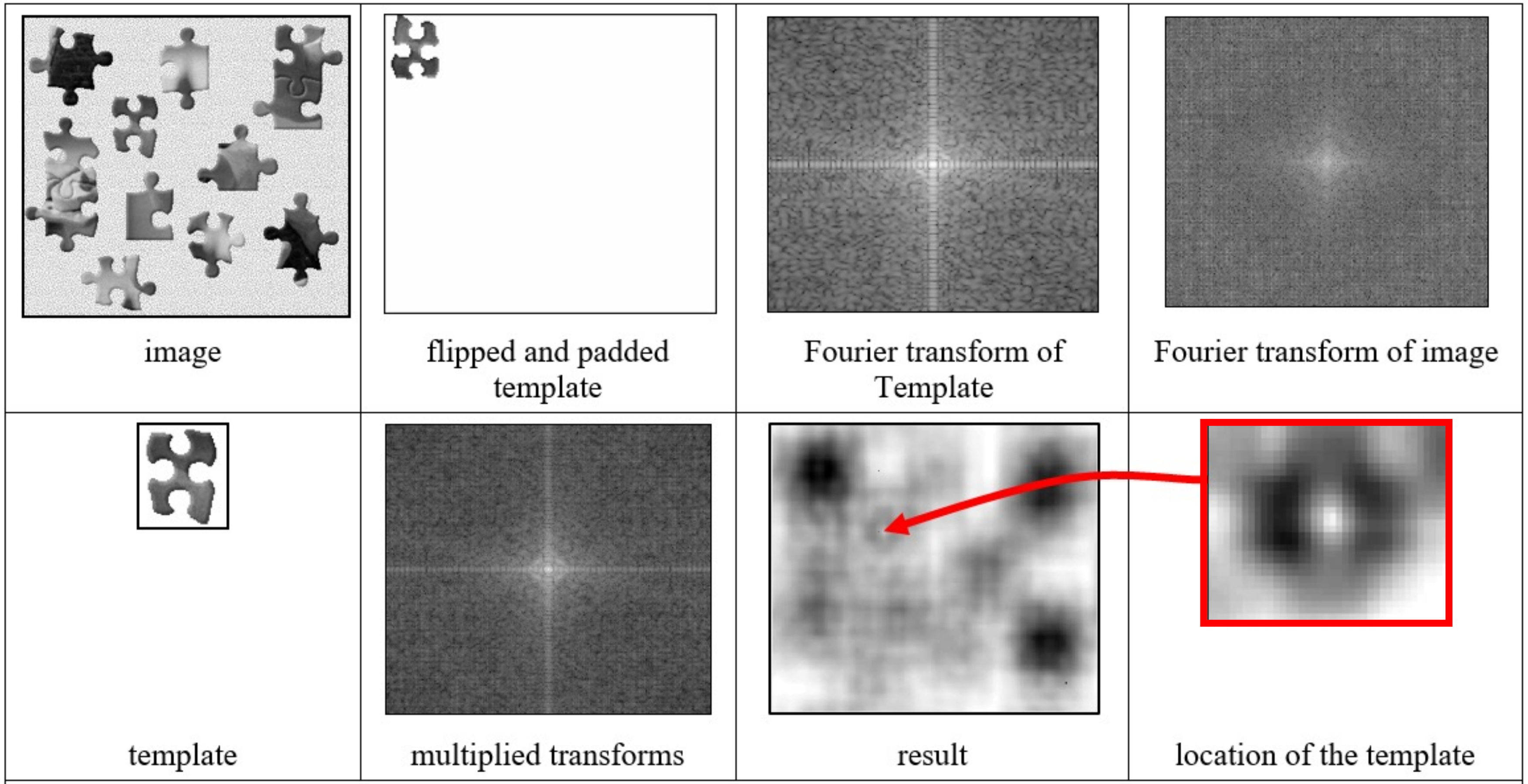
Encore, Baron Fourier!

Template matching is slow, so use FFT

$$(\mathbf{I} \otimes \mathbf{T})(i,j) = \sum_{(x,y)\in W} \mathbf{I}_{x,y}\mathbf{T}_{x+i,y+j}$$

$$\Updownarrow$$

$$F^{-1}(F(\mathbf{I}).\times F(\mathbf{T}_{\text{flip}}))$$



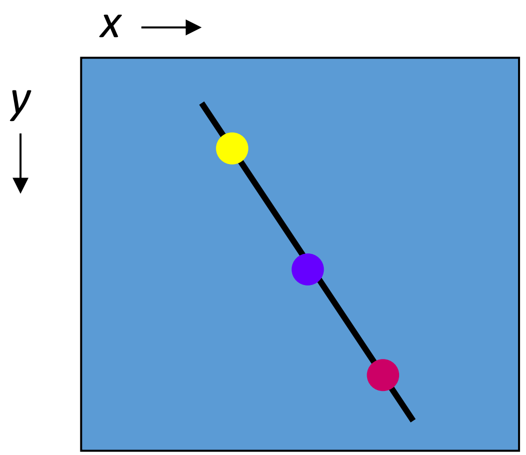
Template Matching via Fourier Transform

No sliding of templates here

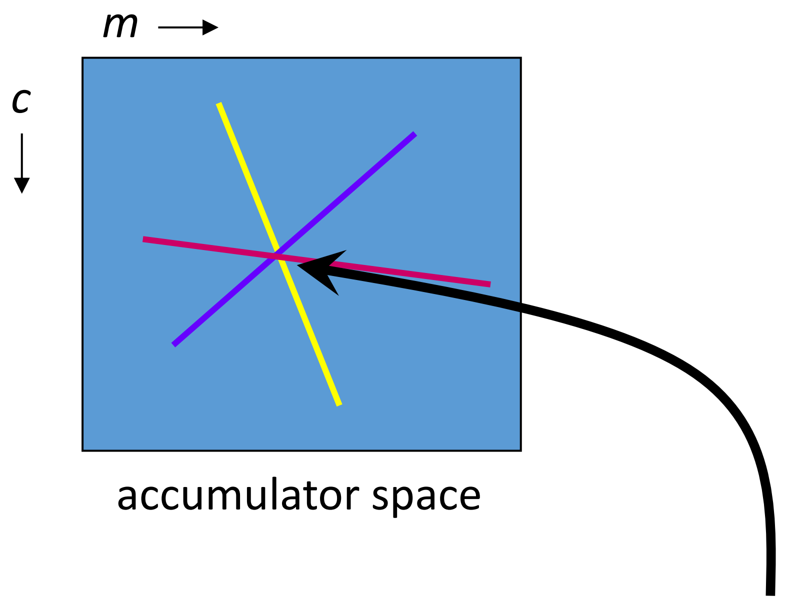
Cost is 2×FFT plus multiplication

Hough Transform

- Performance same as template matching, but faster
- A line is points *x*, *y* gradient *m* intercept *c* $y = m \times x + c$
- and is points *m*, *c* gradient -*x* intercept *y* $c = -x \times m + y$



image



accumulator space

The coordinates of the peak are the parameters of the line

In maths it's the principle of duality

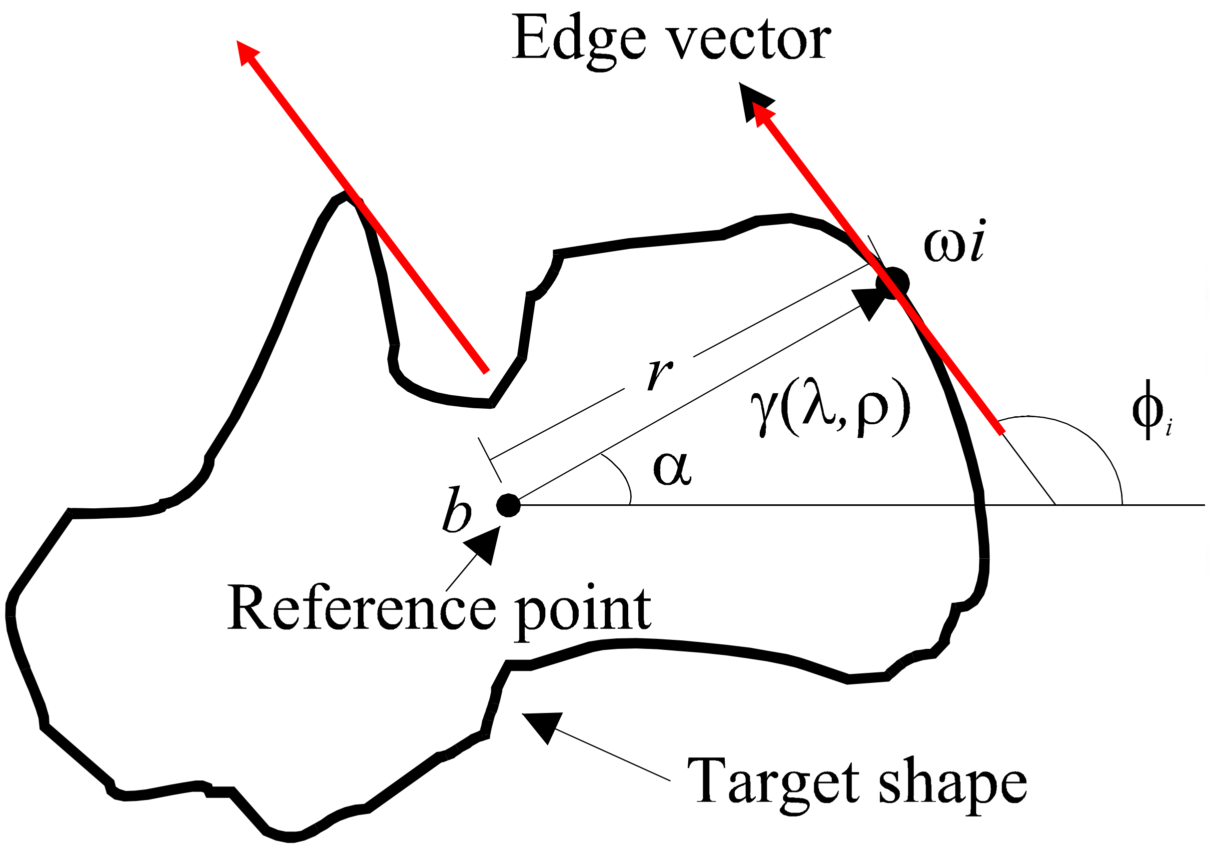
Pseudocode for HT

```
accum=0
for all x,y                              !look at all points
    if edge(y,x)>threshold               !check significance
        for m = -10 to +10               !if so, go thru m
            c = -x*m+y                   !calculate c
            accum(m,c) PLUS 1            !vote in accumulator
m,c = argmax(accum)                      !peak gives parameters
```

R-table Construction

$$x_c = x_i - r\cos(\alpha)$$

$$y_c = y_i - r\sin(\alpha)$$



$\hat{\varphi}'_i$	$\gamma = (r,\alpha)$
0	$(r_0,\alpha_0),(r_1,\alpha_1),(r_2,\alpha_2)$
$\Delta\phi$	⋮
$2\Delta\phi$	⋮
...	...

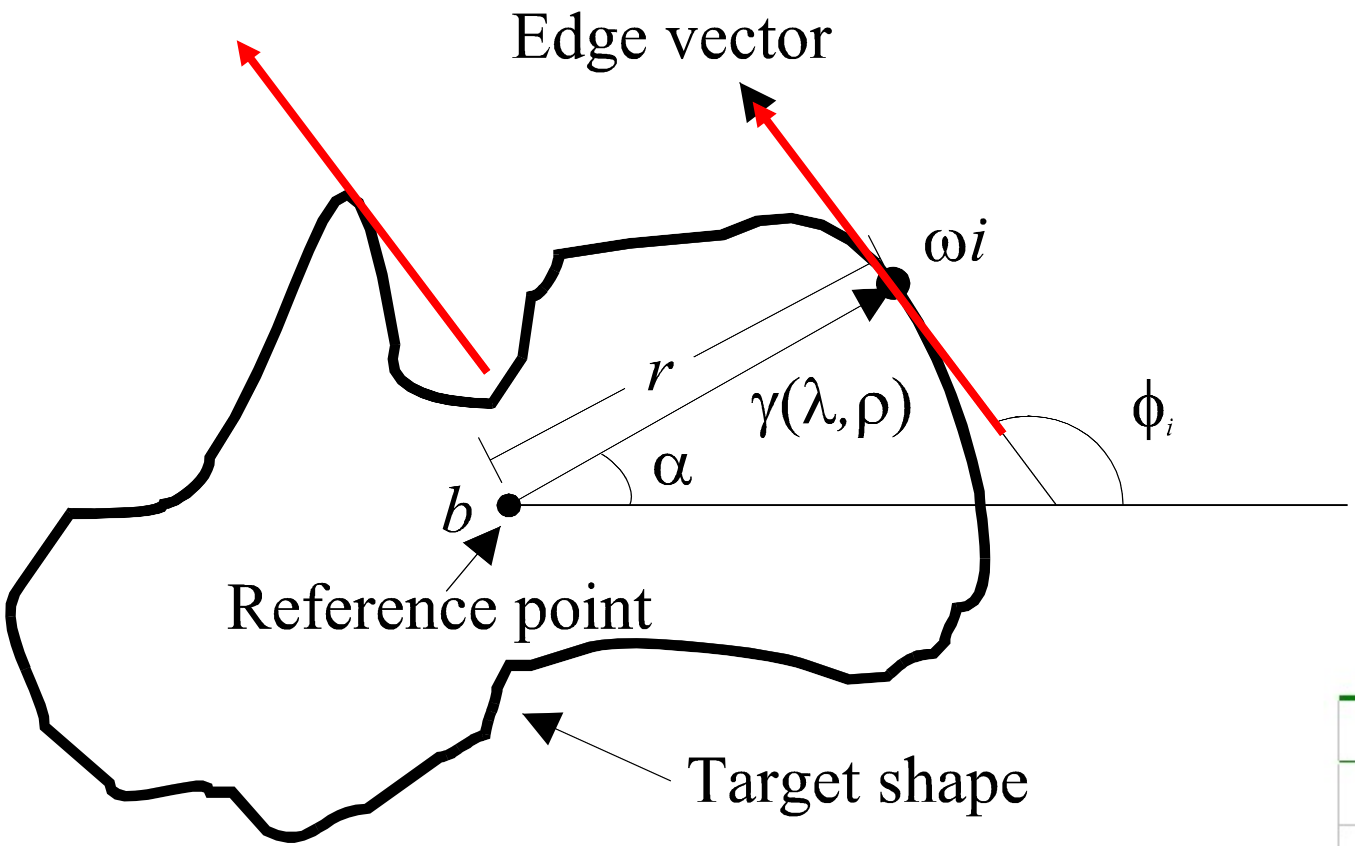
Edge direction is not a unique description

Gives noise in accumulator

R-table Construction

$$x_c = x_i - rS\cos(\alpha + \theta)$$
$$y_c = y_i - rS\sin(\alpha + \theta)$$

Scale Orientation



Important: for the case of rotation, before checking with the R-table, the edge direction of the edge point needs to minus the rotation θ

$\hat{\varphi}'_i$	$\gamma = (r, \alpha)$
0	$(r_0, \alpha_0), (r_1, \alpha_1), (r_2, \alpha_2)$
$\Delta\phi$	⋮
$2\Delta\phi$	⋮
...	...

Question 1.

UNIVERSITY OF Southampton
School of Electronics and Computer Science

First Image of a Black Hole (`https://www.eso.org/public/images/`)

(a) The above image is the first image of a black hole. As we can see, it looks quite blurry. **Provide** an explanation as to why it is so blurred.

[5 marks]

(b) **Design** an edge detection system to detect the edge of the black hole, and **explain** how you address the blurred edges within the image.

[12 marks]

(c) **Design** a shape detection system to detect as many shapes in the image as you can when you have *no template to use*. **Explain** how to *improve* your shape detection system if you are allowed to use templates.

[16 marks]

Thanks and Good luck!