UNIVERSITEIT
GENT

FACULTEIT INGENIEURSWETENSCHAPPEN
EN ARCHITECTUUR

VAKGROEP TELECOMMUNICATIE EN INFORMATIEVERWERKING
ONDERZOEKSGROEP IMAGE PROCESSING AND INTERPRETATION

Multimedia – Labo 4

Gianni Allebosch, Martin Dimitrievski
{gianni.allebosch, martin.dimitrievski}@ugent.be

Master of Science in de industriële wetenschappen: elektronica-ICT

Campus Ardoyen,
BE - 9000 Gent

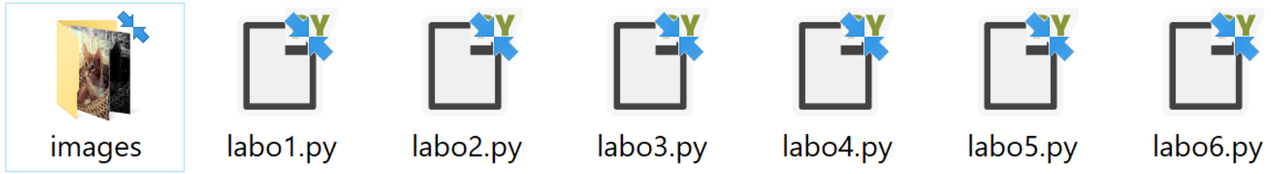
UNIVERSITEIT
GENT

TIPS & TRICKS

- **Keep your program user friendly!**
  - Provide a README.txt
  - Execute each exercise sequentially in 1 script
  - Make clear which exercise is currently being executed
  - Name your windows when showing images
  - If input is required from the user, make sure the instructions are clear

TIPS & TRICKS

- **Be sure the code is executable as is!**
  - We will not change your code!
  - Functions in comments will not be executed
  - Store all input images in a folder named “images” next to your code



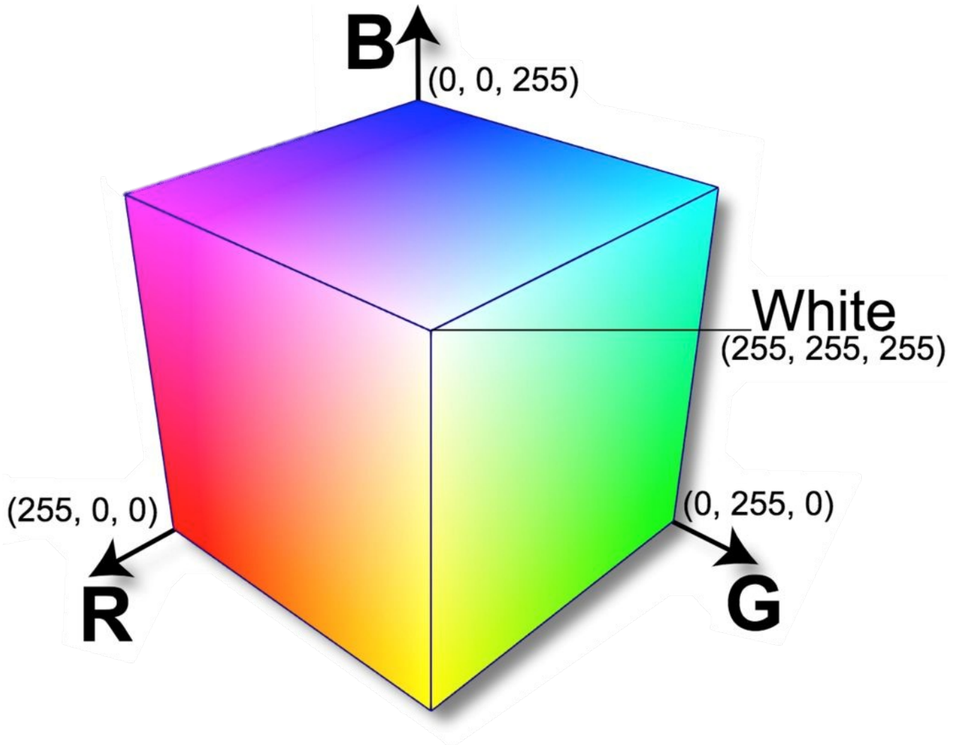
  - Do not use absolute paths but use relative paths (portability!):

✗ “C:/users/admin/multimedia/labo1/images/lena_color.jpg”

✔ “images/lena_color.jpg”

UNIVERSITEIT GENT

Color spaces - RGB

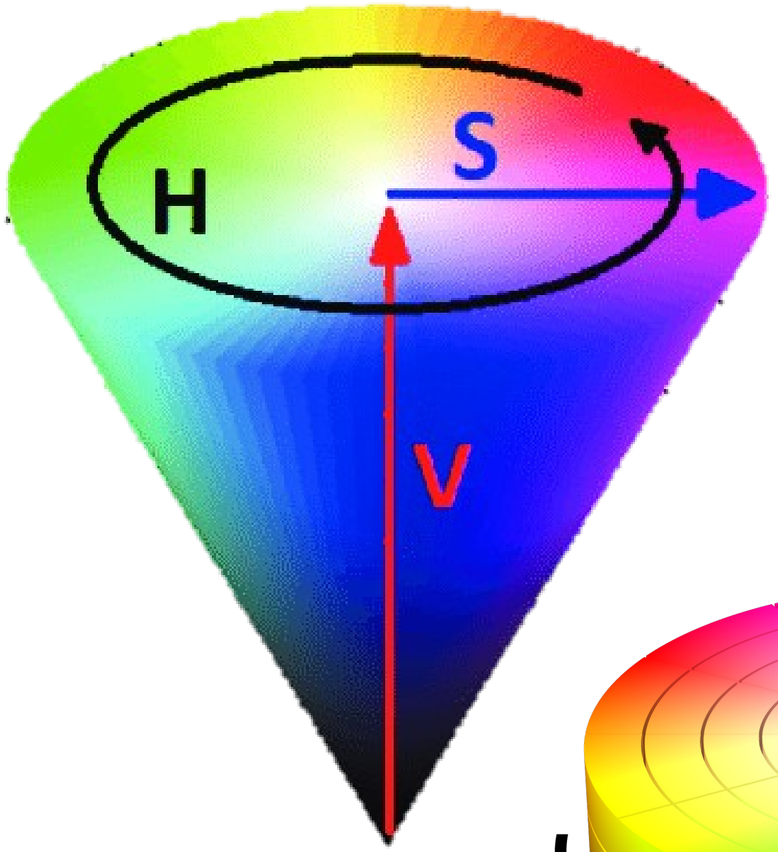


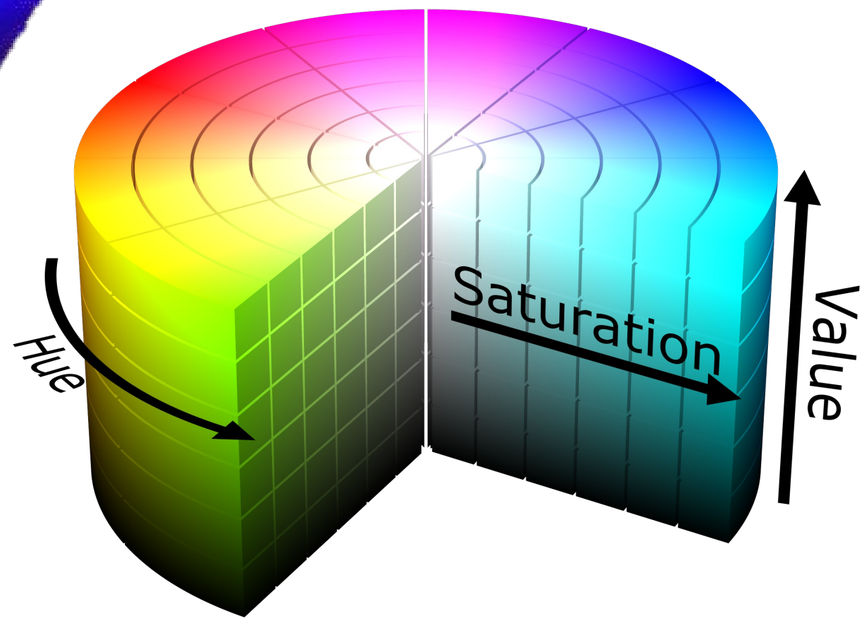
- **Additive** color mixing:
  - Red, green and blue are primary colors
  - Cyan, magenta and yellow are secondary colors
  - Any point inside the cube represents a color
- Computer: 3 color channels with typically 8 bits/color component (24 bits/pixel)
- Total number of colors:

$$(2^8)^3 = 16{,}777{,}216$$

UNIVERSITEIT GENT

Color spaces - HSV

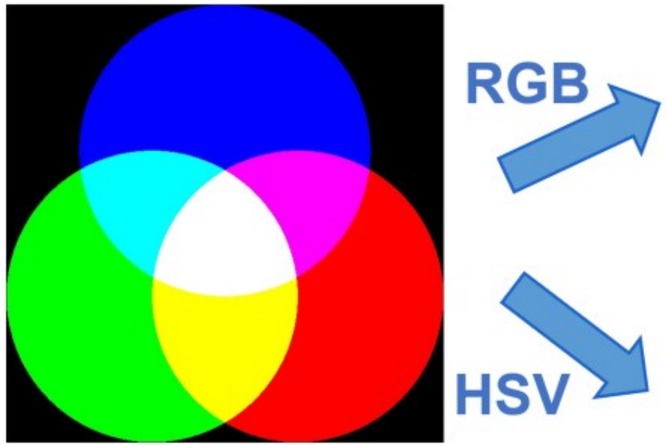




- Closer to the way human vision perceives color
- a color is represented by:
  - **Hue**: color type
  - **Value**: lightness of the color (amount of black/white)
  - **Saturation**: shade of the color (amount of gray)

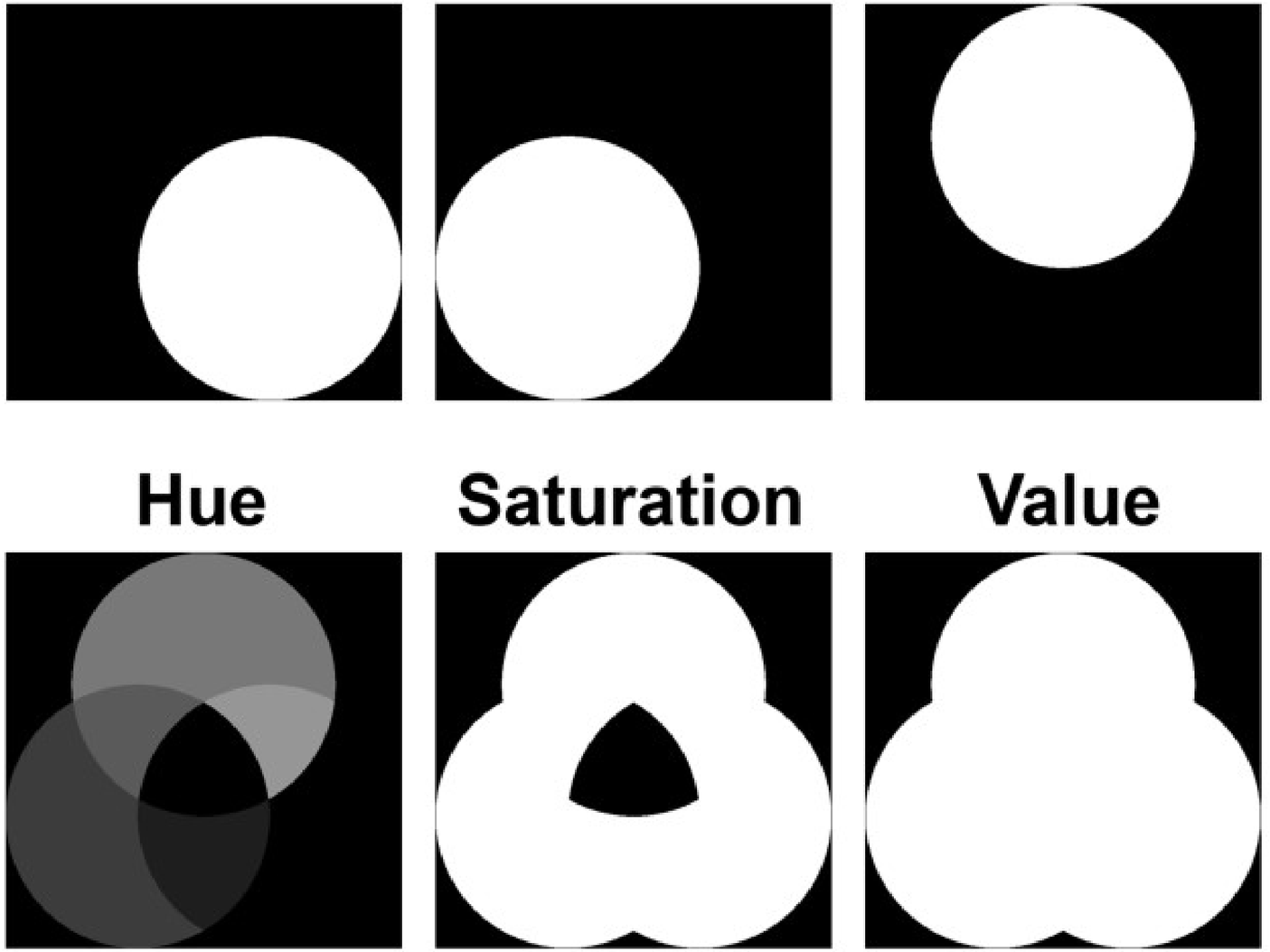
UNIVERSITEIT GENT

Color spaces – RGB vs. HSV



RGB

HSV



UNIVERSITEIT GENT

Color spaces – RGB to HSV

$$\mathrm{V} = \max(R, G, B)$$

$$S = \begin{cases} 0, & V = 0 \\ \dfrac{V - \min(R, G, B)}{V}, & V \neq 0 \end{cases}$$

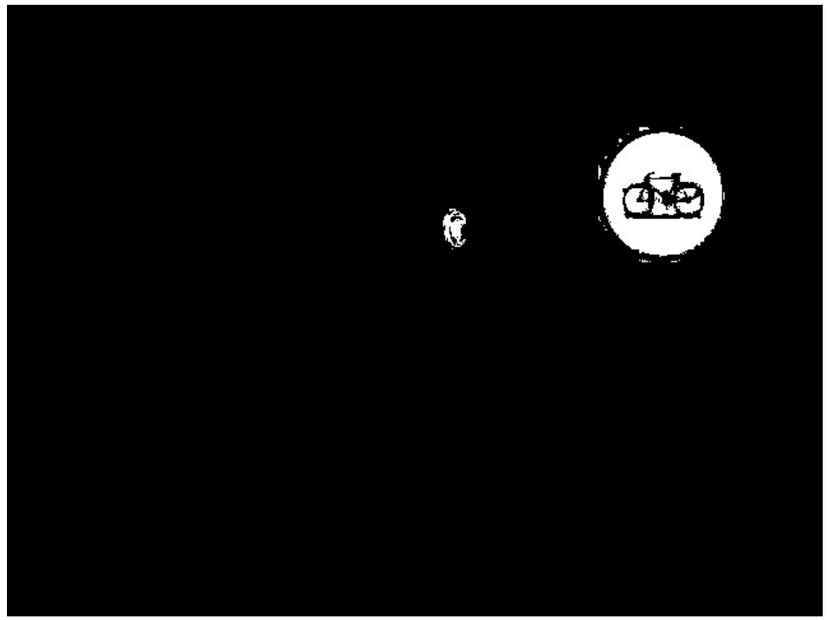
$$H = \begin{cases} \dfrac{30(G - B)}{V - \min(R, G, B)}, & V = R \\ 60 + \dfrac{30(B - R)}{V - \min(R, G, B)}, & V = G \\ 120 + \dfrac{30(R - G)}{V - \min(R, G, B)}, & V = B \end{cases}$$

UNIVERSITEIT GENT

Exercise 1 – Color Segmentation - RED

UNIVERSITEIT GENT

Exercise 1 – Color Segmentation - BLUE



UNIVERSITEIT GENT

Exercise 1 – Color Segmentation - WHITE

UNIVERSITEIT GENT

Exercise 1 – Color Segmentation – BLACK

UNIVERSITEIT GENT

Edge Detection

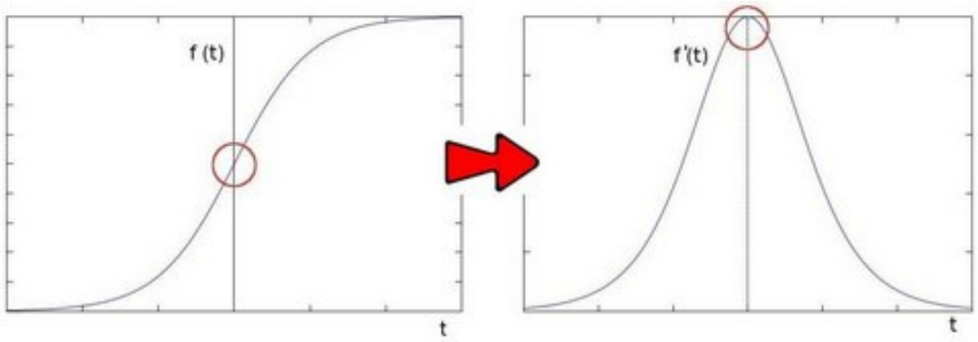
- Edges = discontinuities in intensity

- Detect horizontal, vertical and diagonal edges with spatial filters

UNIVERSITEIT GENT

Edge Detection

- Search-based: edge strength with first order derivative



-1	-1	-1
0	0	0
1	1	1

Prewitt

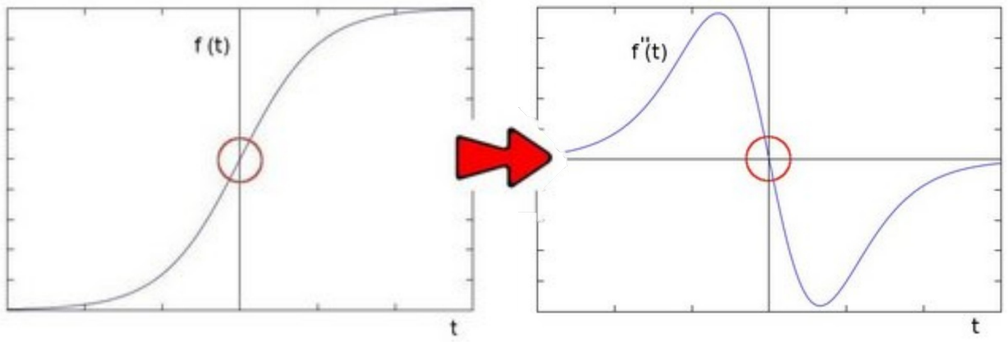
-1	-2	-1
0	0	0
1	2	1

Sobel

-1	0
0	1

Roberts

- Zero-crossing based: zero crossings in second order derivative



0	1	0
1	-4	1
0	1	0

Laplacian

UNIVERSITEIT GENT

Edge Detection - Canny

1) Noise Reduction

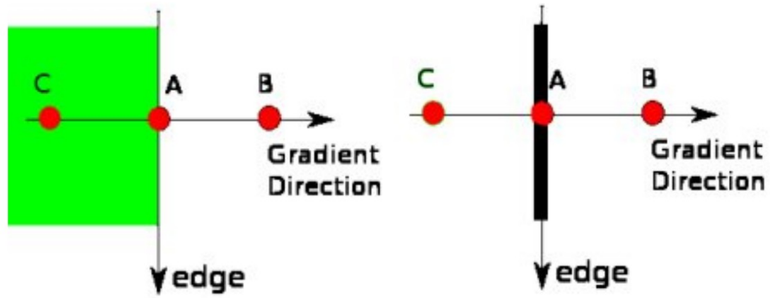
→ smoothing with Gaussian filter

2) Finding Intensity Gradient of the Image

→ edge gradient and direction with Sobel filter

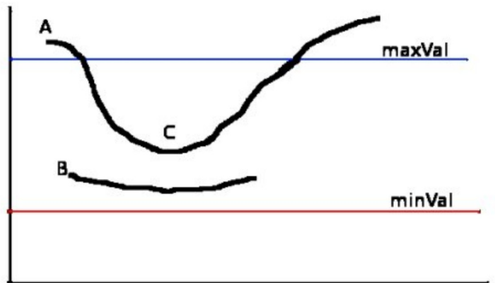
3) Non-maximum Suppression

→ edge thinning



4) Hysteresis Thresholding

→ edge linking



UNIVERSITEIT GENT

Question – Edge Detection

- **Solve the question in your README.txt**
  - Why does the Sobel edge detector have better noise suppression than the Prewitt edge detector?

Line detection – Hough transform

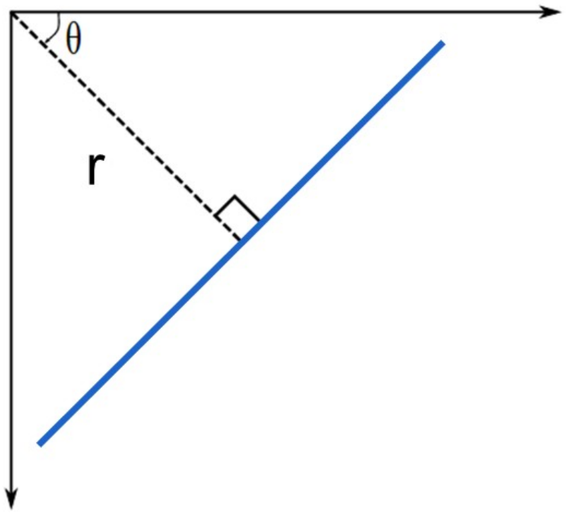
Line representation:

$$y = mx + b \rightarrow (m, b)$$

however, vertical lines $\rightarrow m = \infty$

Alternative line representation:

$$r = x\cos\theta + y\sin\theta \rightarrow (r, \theta)$$



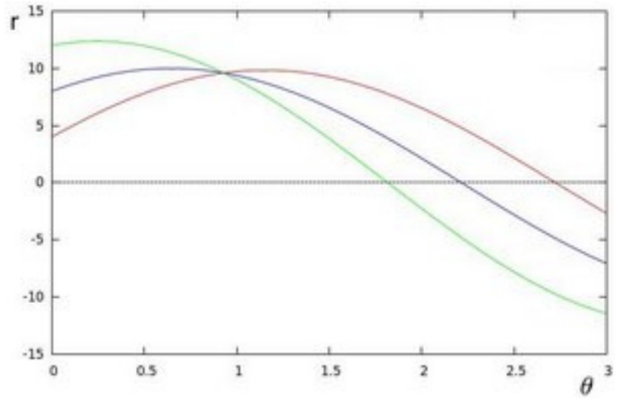
UNIVERSITEIT GENT

Line detection – Hough transform

Accumulator for every (r,θ):

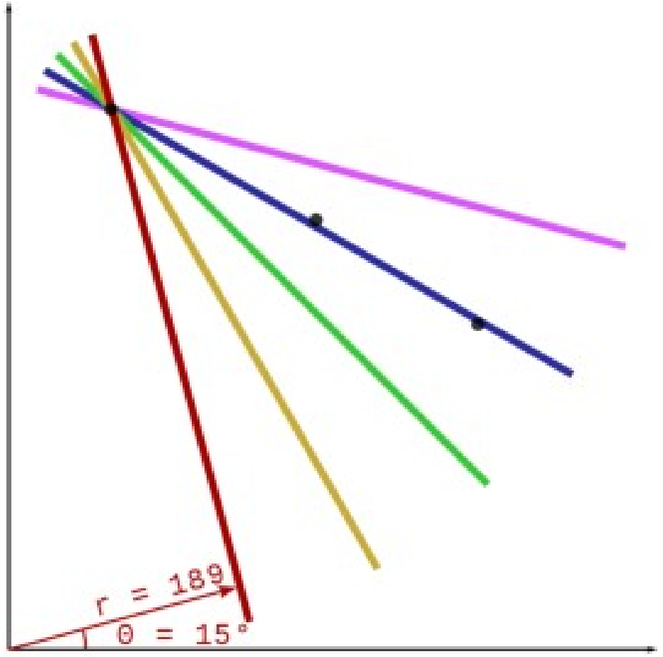
- Initialised with zeros
- rows = r ◊ $\#rows = \frac{\sqrt{W^2+H^2}}{pixel\ accuracy}$
- columns = θ ◊ $\#columns = \frac{180°}{angle\ accuracy}$
- Vote for every possible (r,θ) for every edge pixel (x,y):

$$r = x\cos\theta + y\sin\theta\ , with\ \theta\epsilon[0°, 180°[$$



UNIVERSITEIT GENT

Line detection – Hough transform

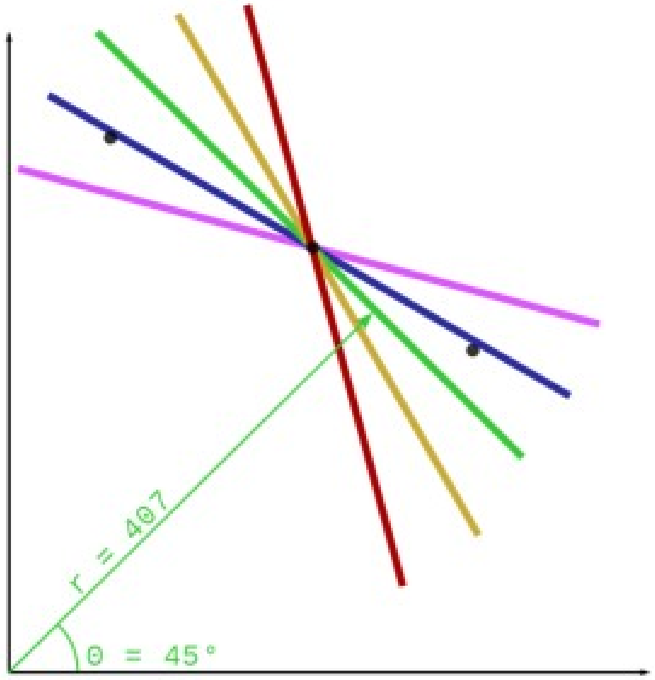


θ	r
15	189.0
30	282.0
45	355.7
60	407.3
75	429.4

R \ θ	15	30	45	60	75
200	1				
225					
250					
275		1			
300					
325					
350			1		
375					
400				1	
425					1
450					

UNIVERSITEIT GENT

Line detection – Hough transform

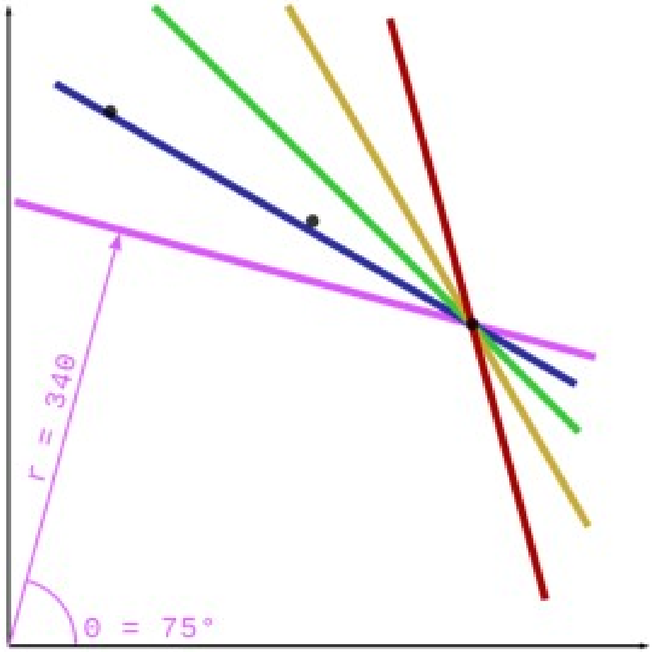


θ	r
15	318.5
30	376.8
45	407.3
60	409.8
75	385.3

R \ θ	15	30	45	60	75
200	1				
225					
250					
275		1			
300					
325	1				
350			1		
375		1			1
400			1	2	
425					1
450					

UNIVERSITEIT GENT

Line detection – Hough transform

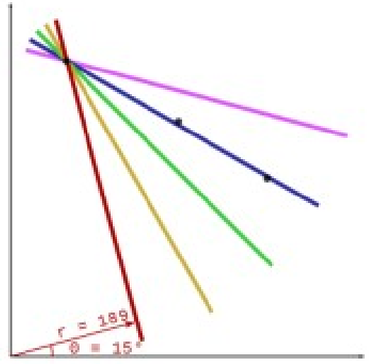


θ	r
15	419.0
30	443.6
45	438.4
60	402.9
75	340.1

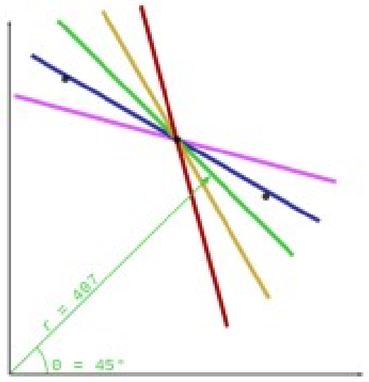
R \ θ	15	30	45	60	75
200	1				
225					
250					
275		1			
300					
325	2				
350			1		1
375		1			1
400			1	3	
425	1				1
450		1	1		

UNIVERSITEIT GENT

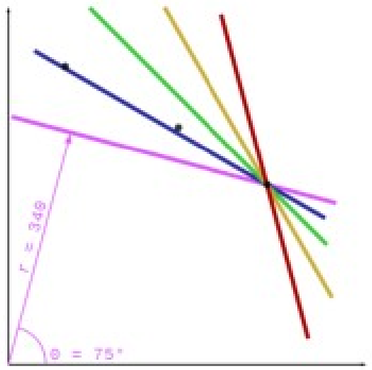
Line detection – Hough transform



θ	r
15	189.0
30	282.0
45	355.7
60	407.3
75	429.4



θ	r
15	318.5
30	376.8
45	407.3
60	409.8
75	385.3

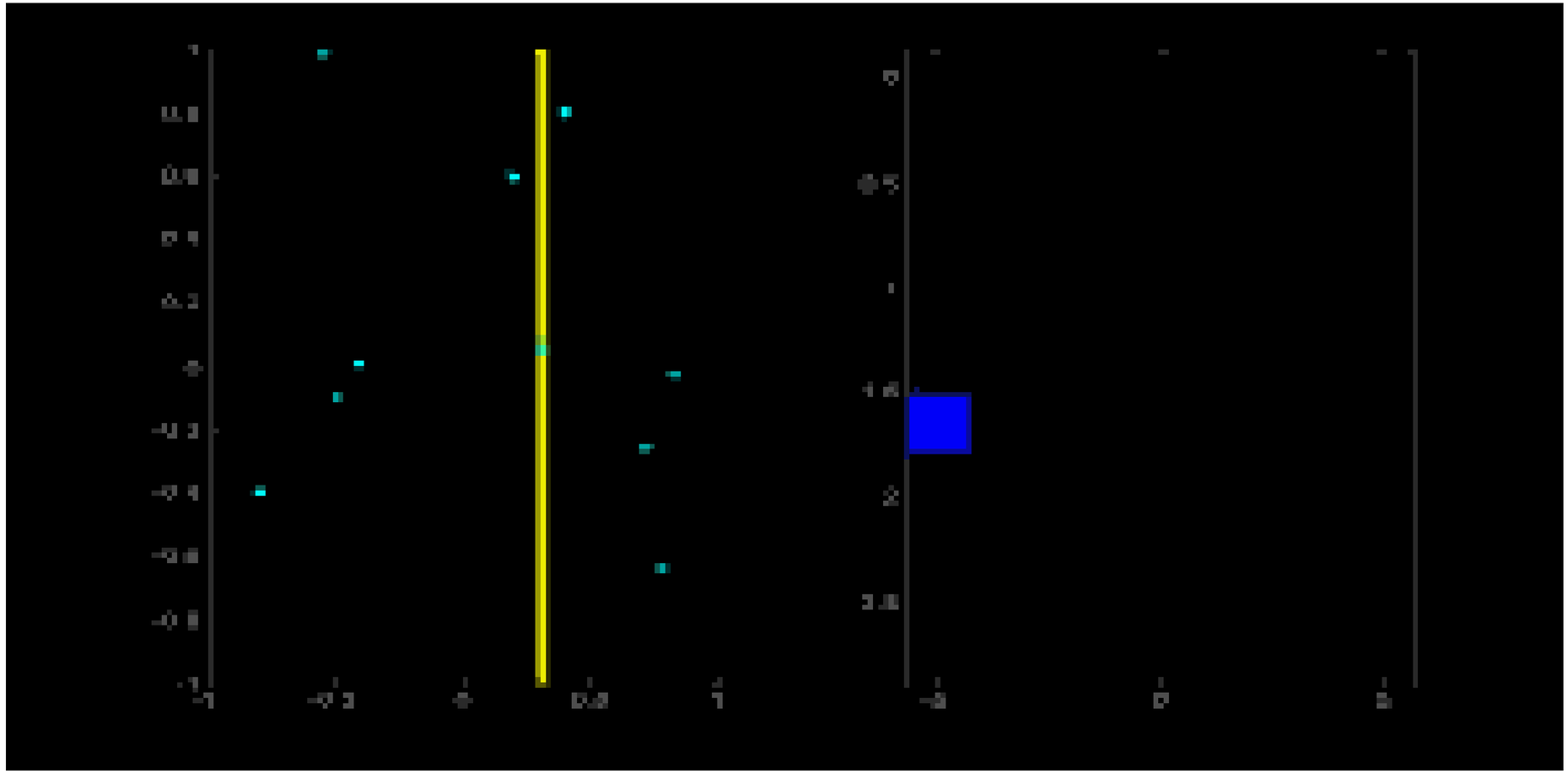


θ	r
15	419.0
30	443.6
45	438.4
60	402.9
75	340.1

R \ θ	15	30	45	60	75
200	1				
225					
250					
275		1			
300					
325	2				
350			1		1
375		1			1
400			1	3	
425	1				1
450		1	1		

UNIVERSITEIT GENT

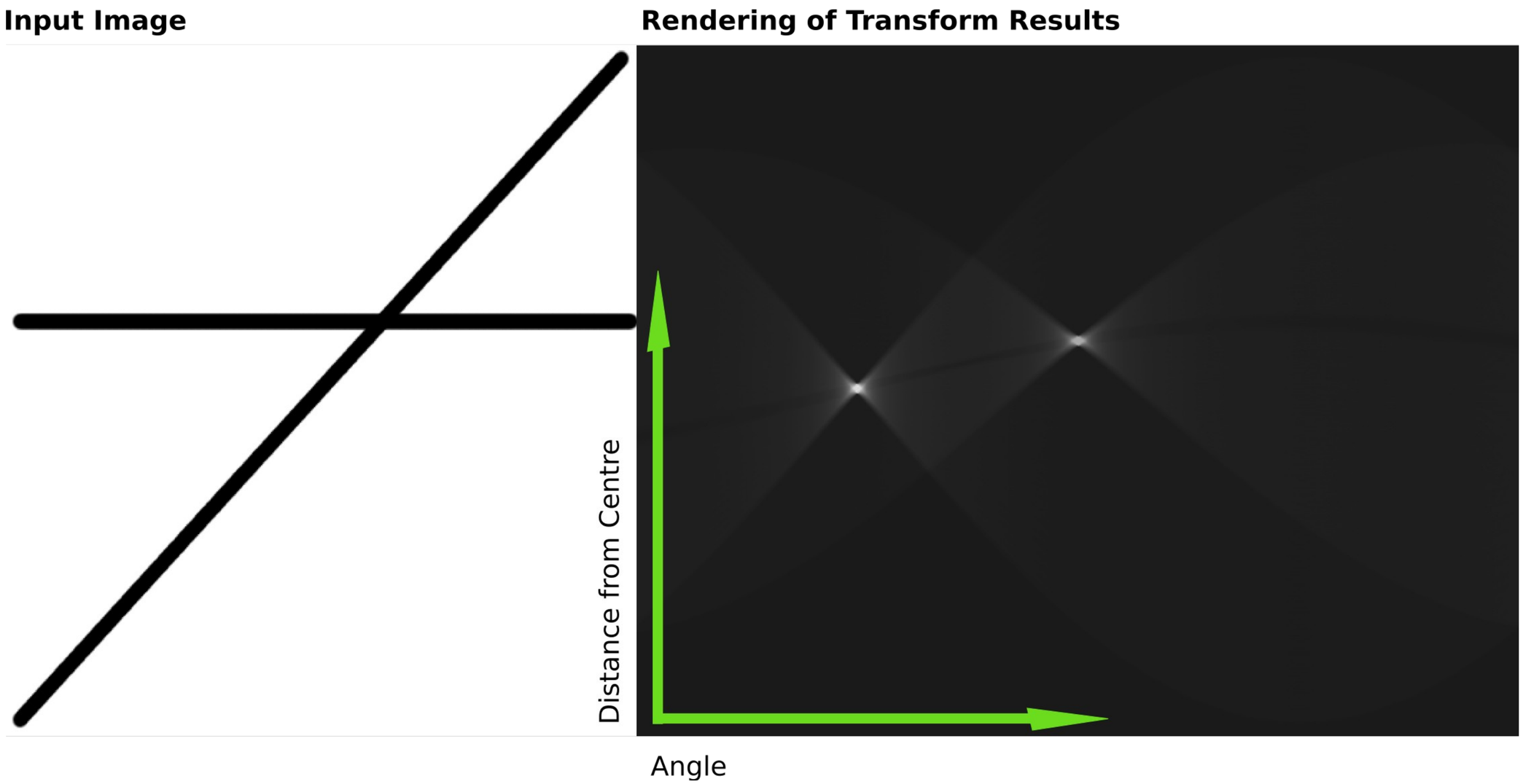
Line detection – Hough transform



Gif example from http://homepages.inf.ed.ac.uk/amos/demos/houghdemoweb.gif

UNIVERSITEIT GENT

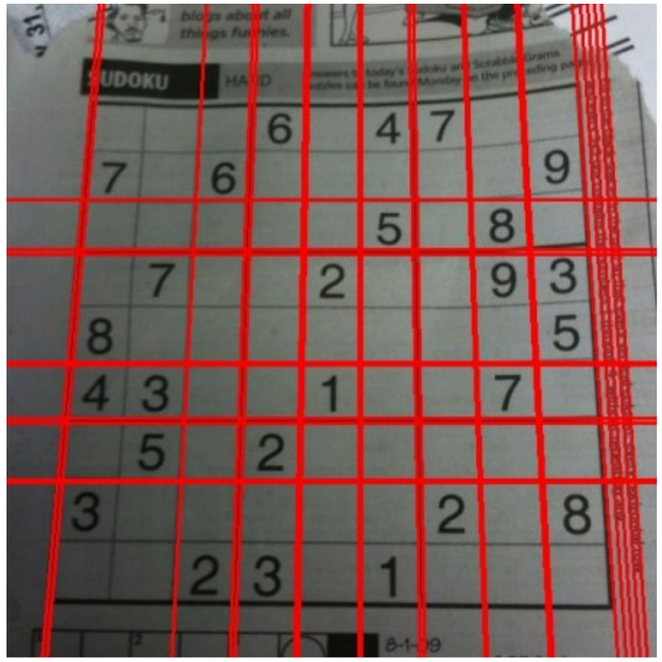
Line detection – Hough transform



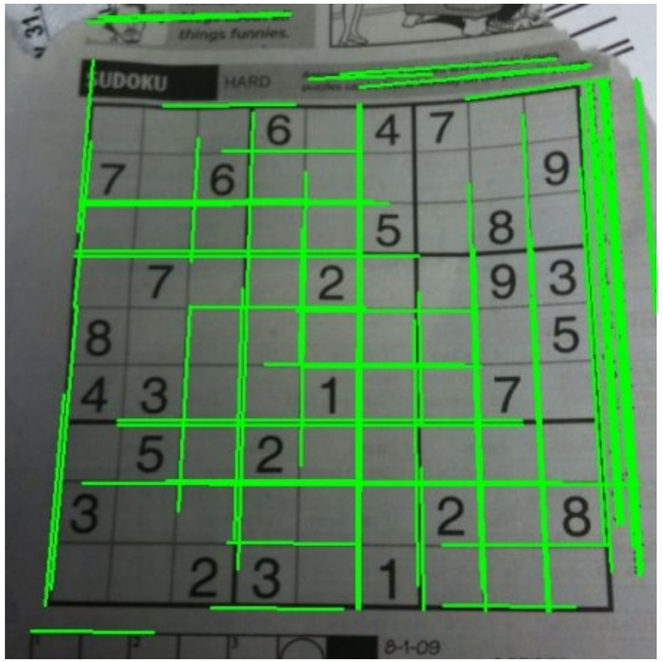
UNIVERSITEIT GENT

Line detection – Hough transform

- **Hough Transform:** all edge points
- **Probabilistic Hough Transform:** random subset of edge points



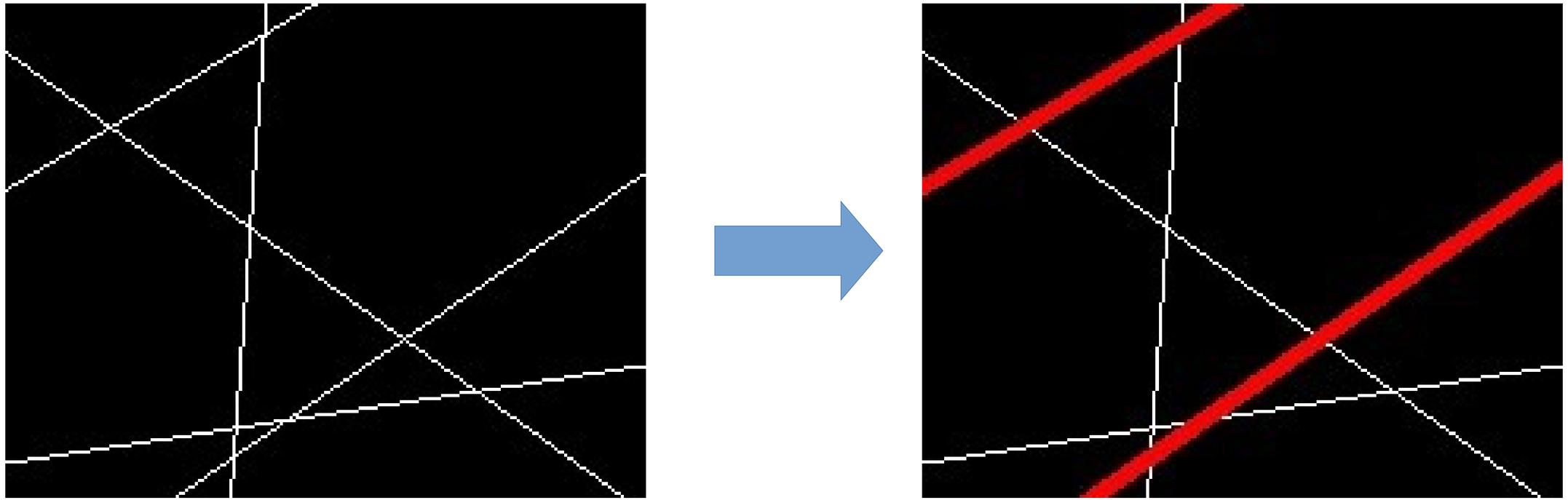
Hough Transform



Probabilistic Hough Transform

UNIVERSITEIT GENT

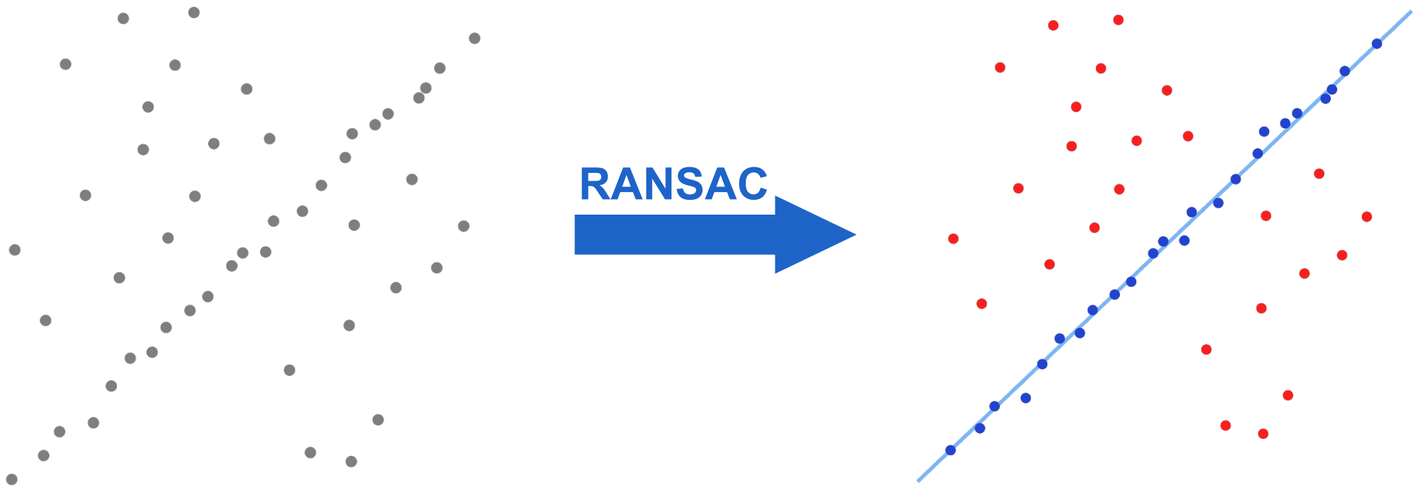
Exercise 2 – Line Detection



Line detection – RANSAC

RANdom **SA**mple **C**onsencus:

- iterative method of fitting a mathematical model to a collection of points with inliers and outliers
- non-deterministic: reasonable result only with a certain probability (rises with more iterations)



UNIVERSITEIT GENT

Line detection – RANSAC

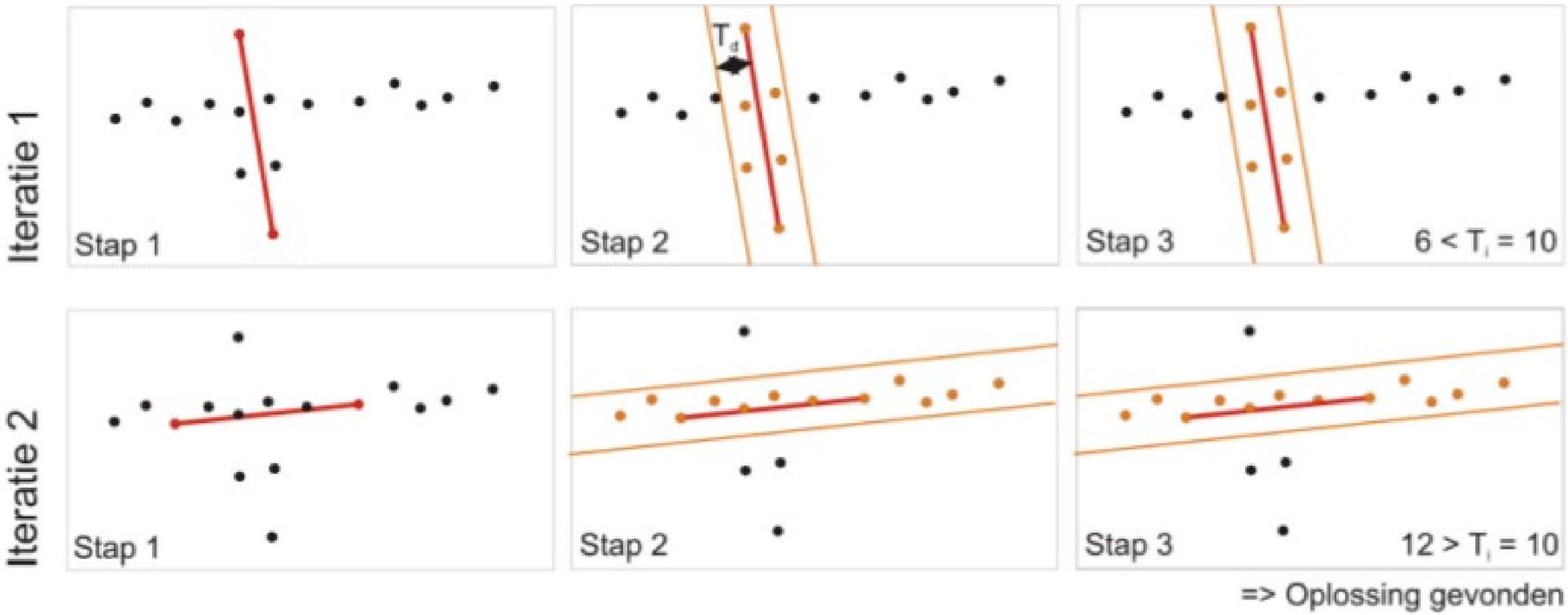
1) Select random sample of *s* points and construct its line

2) Find every point within distance T_d from this line = inliers

3) Compare the number of inliers *r* with threshold T_i

4) Repeat until $r \geq T_i$

5) If no solution is found after N iterations, select sample with largest number of inliers



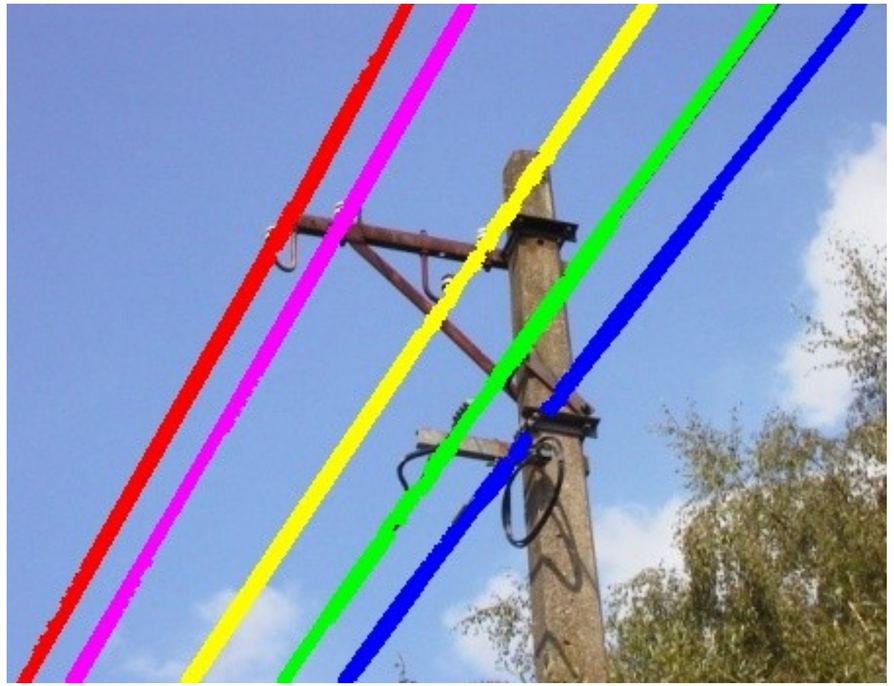
UNIVERSITEIT GENT

Question – RANSAC

- **Solve the question in your README.txt**
  - What is the minimum value of the parameter *s* if RANSAC is used to detect circles?

UNIVERSITEIT GENT

Exercise 3 – RANSAC



TIPS & TRICKS

- **Deliver your working code before next session**
  - *Opdrachten* on Ufora
  - 1 python script named **labo#.py**
  - README.txt (see example)
  - **output.zip** with every output image

TIPS & TRICKS

- **Questions or need help? Contact us at:**

  Gianni.Allebosch@UGent.be AND Martin.Dimitrievski@UGent.be

UNIVERSITEIT GENT

GOOD LUCK!