Image derived contractor and its application in robot localization

Jan SLIWKA
PhD student at ENSIETA

"**Robust solving** of localization, mapping and SLAM problems in the field of underwater robotics using **set membership methods**"

My website: www.ensieta.fr/sliwka
M Luc JAULIN's website: www.ensieta.fr/jaulin

ENSIETA
ÉCOLE NATIONALE SUPÉRIEURE D'INGÉNIEURS

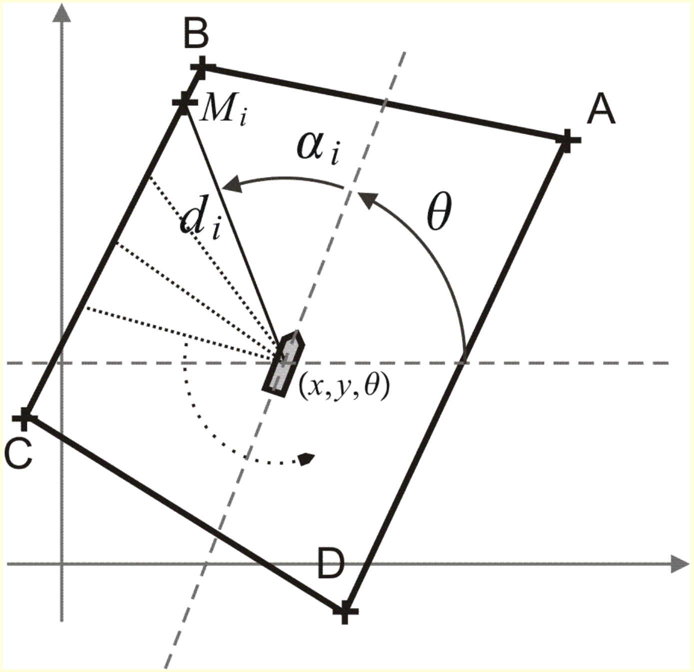
Context

Localization :

→ Map (structured environment, **binary image**)

→ Position (X, Y, Heading)

→ Measures (Sonar measures and compass)



$$\forall (d_i, \alpha_i, \theta_i) \in Correct\_Measures$$

$$(x, y, \theta) \in Pose, \exists (x_{map}, y_{map}) \in Map$$

$$(x, y, \theta) = H(d_i, \alpha_i, \theta_i, x_{map}, y_{map})$$

Context

Localization :

→ Map (structured environment, **binary image**)

→ Position (X, Y, Heading)

→ Measures (Sonar measures and compass)

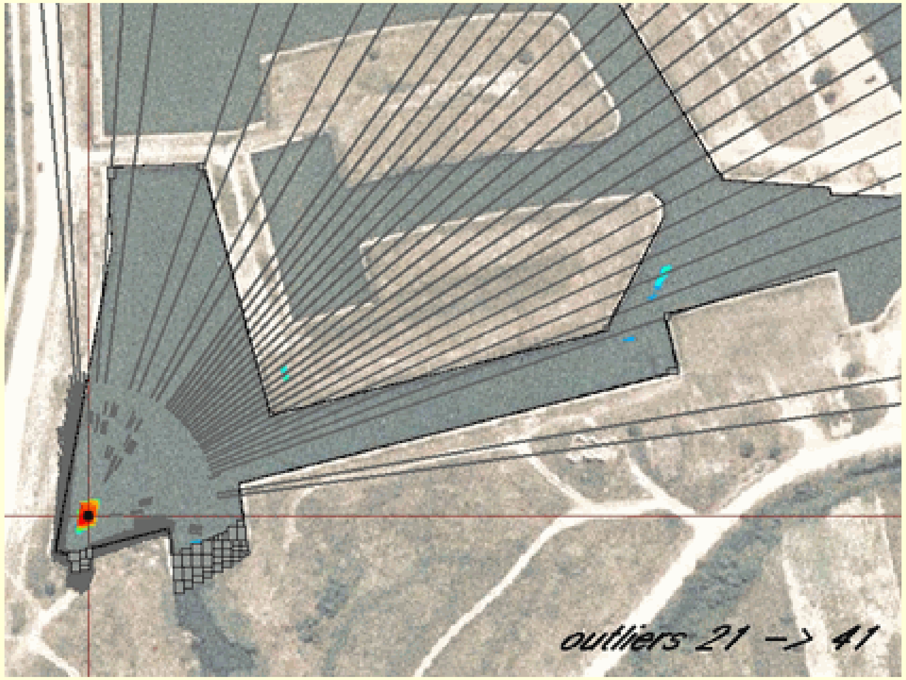
Context

Localization :

→ Map (structured environment, **binary image**)

→ Position (X, Y, Heading)

→ Measures (Sonar measures and compass)



Contractor on binary image

- **Goal** :

→ **Not** limit oneself to **structured** environments

→ Image representation : use **image processing** optimized algorithms (OpenCV)

→ Use of integers : Implement on hardware for **parallel computing** as the FPGA.

What is an image contractor

Consider a binary image

> Continuous representation :

$$f : \mathbb{R}^2 \rightarrow \{0,1\}$$

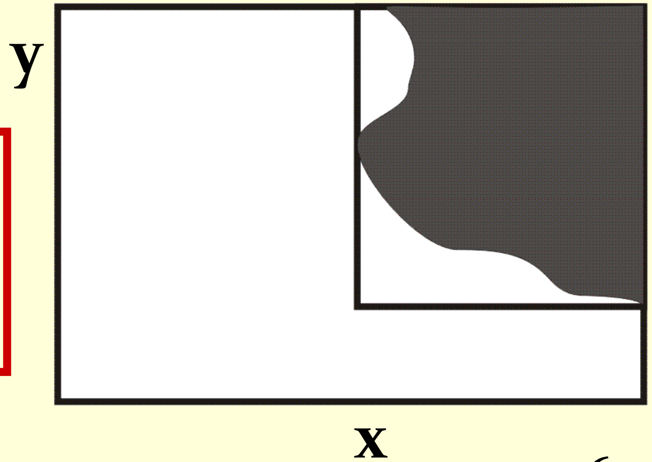
> Discrete representation

$$f : [1..Width] \times [1..Height] \rightarrow \{0,1\}$$

Contraction of a box P

$$E = \{(x,y) \in P, f(x,y) = 1\}$$

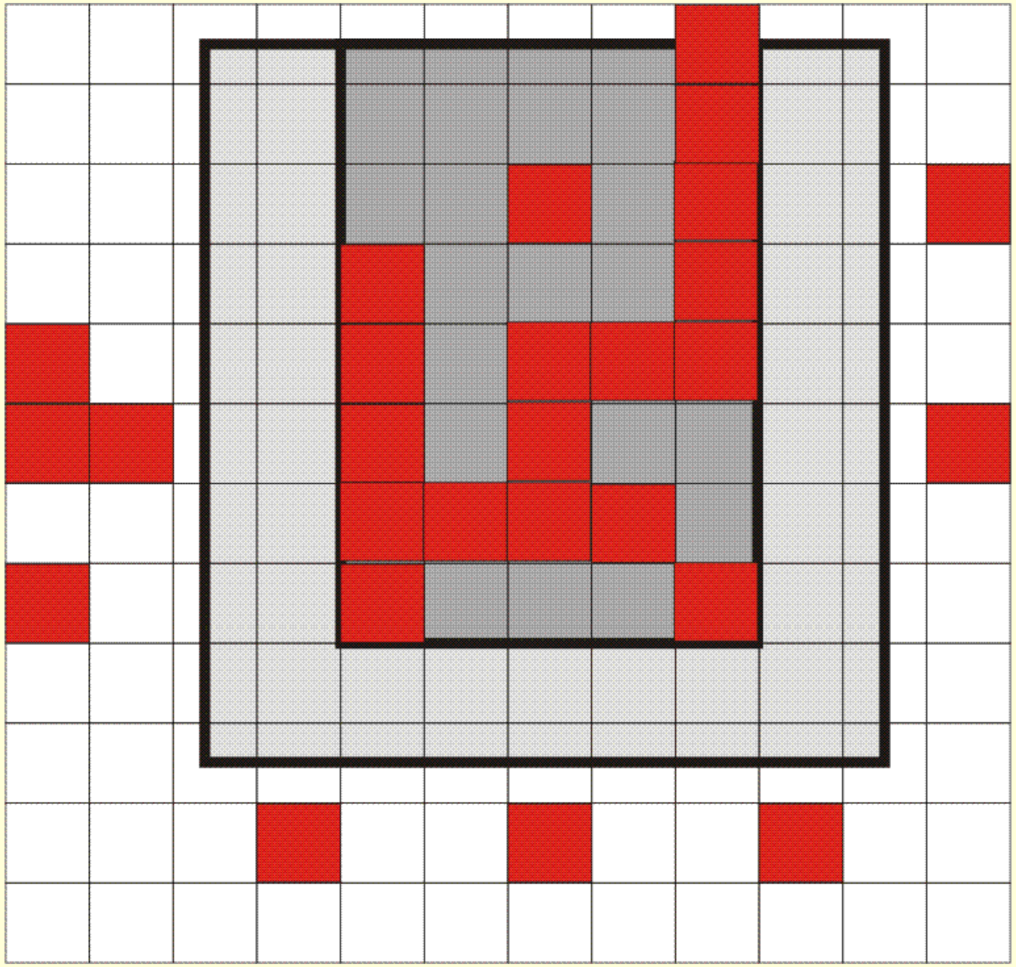
$$Contracted\_P = enclosing\_box\_of(E)$$



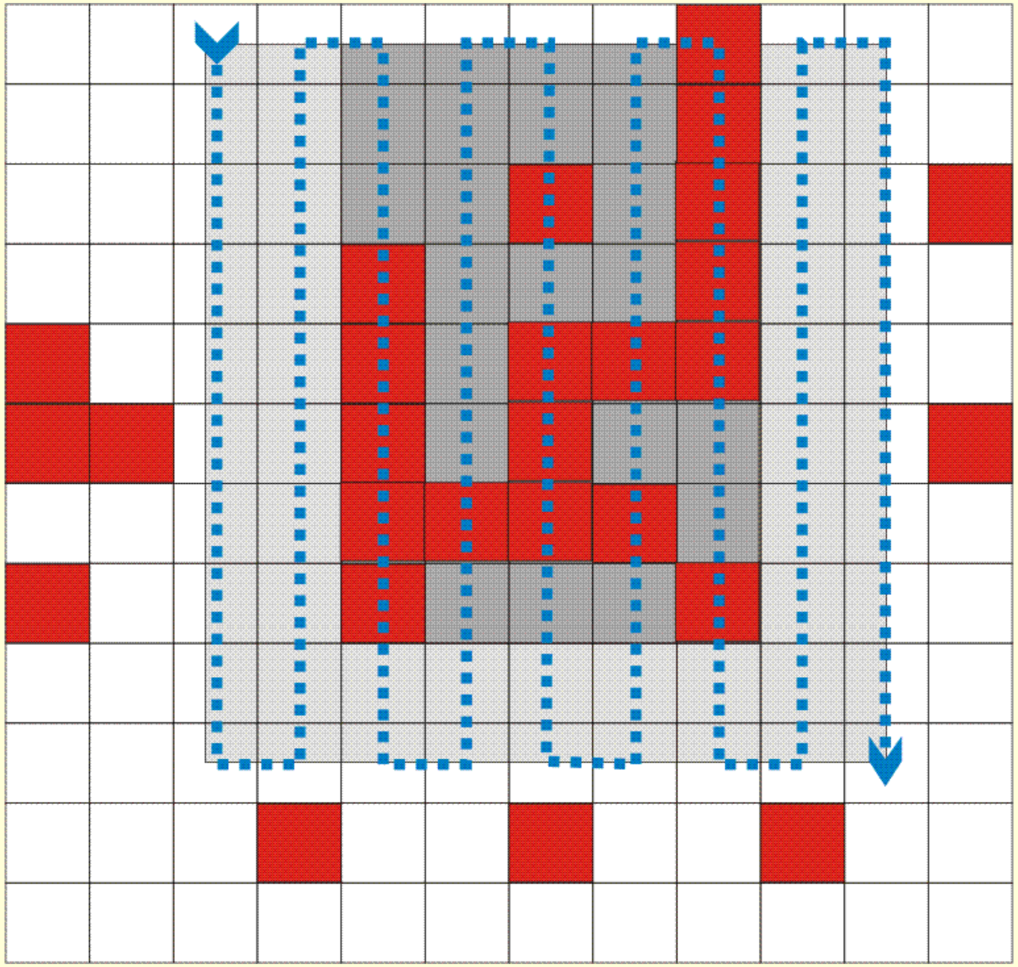
Plan

- Show 4 methods to implement Image Contractor
- Some applications

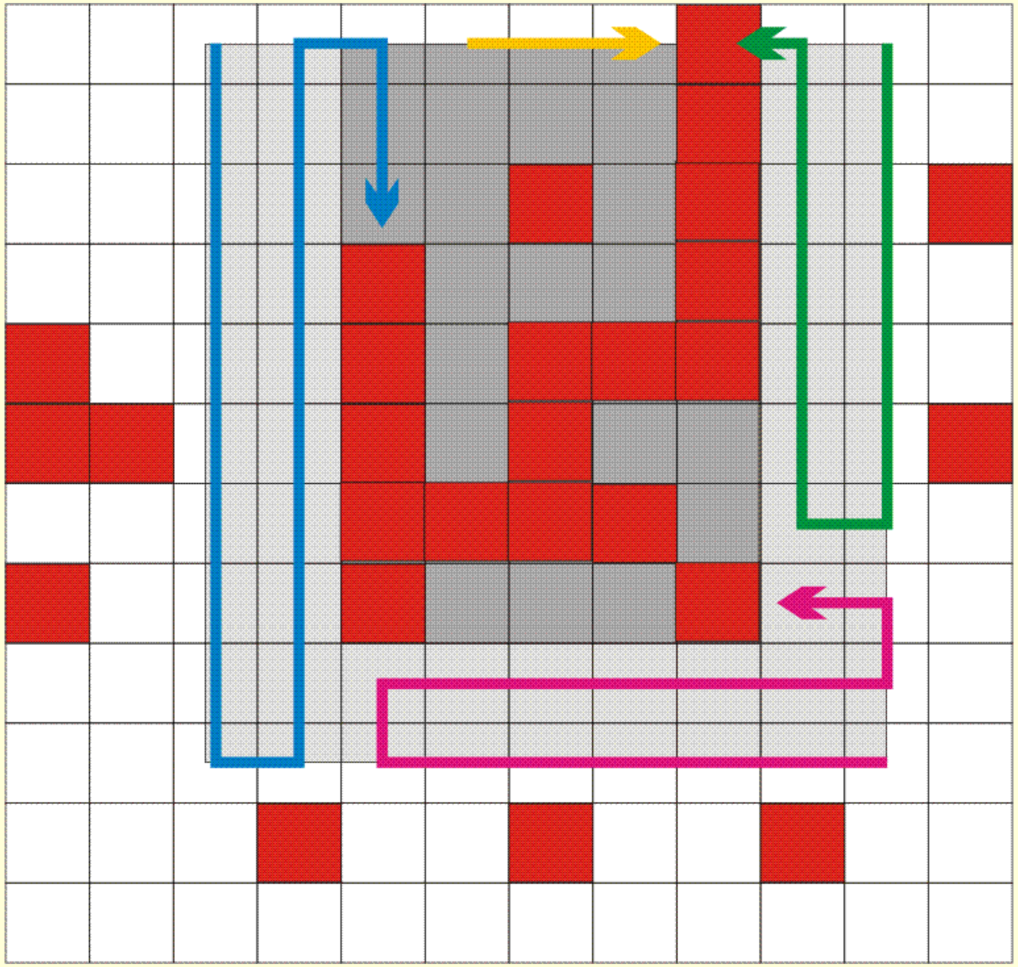
Example



naïve method (directly on image)



Less naïve method (directly on image)



Second method : 2D structure

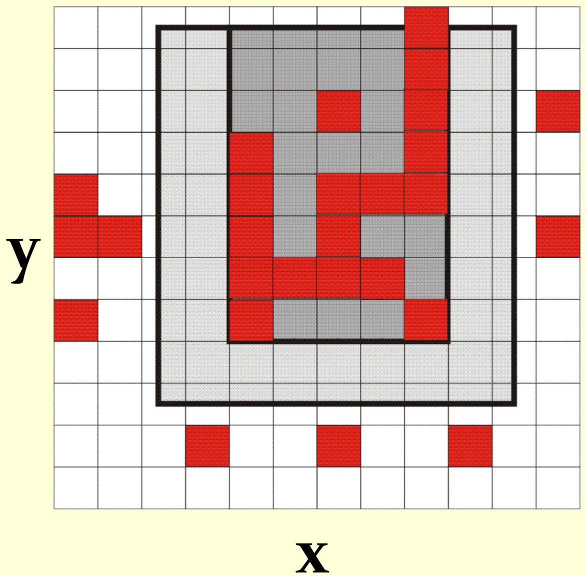
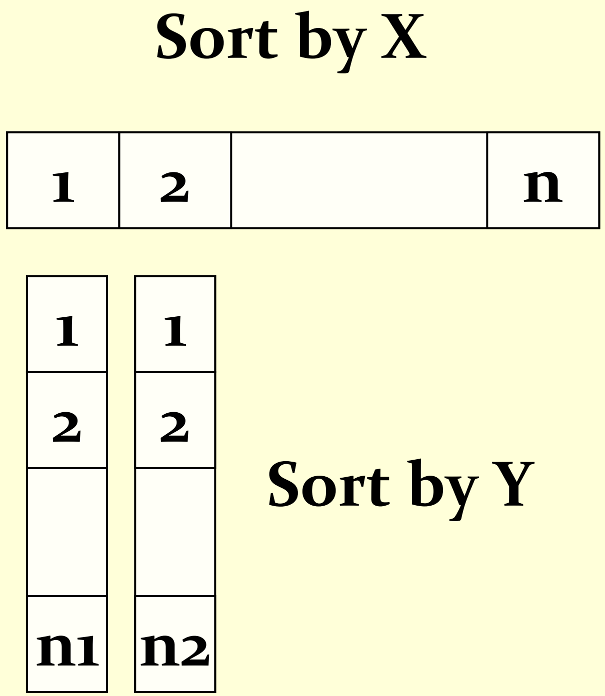


Image and example of contraction

re-organize once for all



New data structure

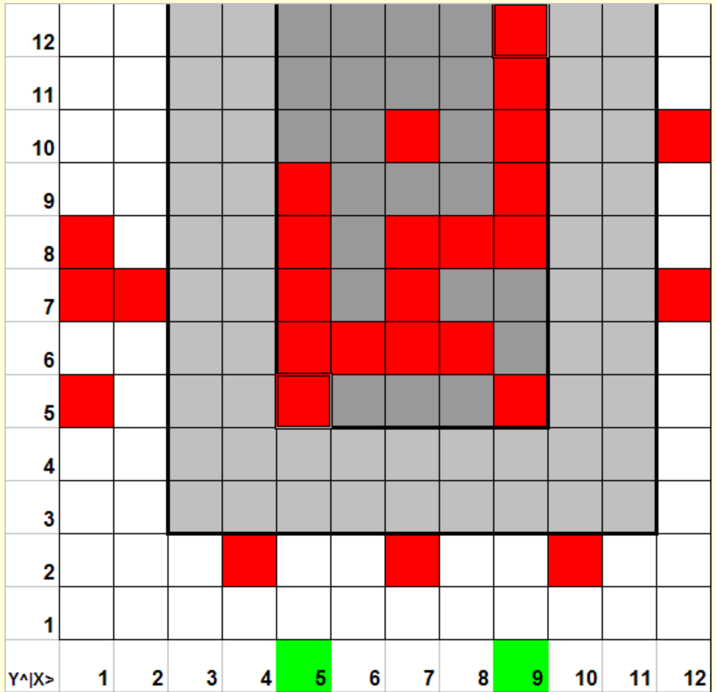
Second method : 2D structure

12												
11												
10												
9												
8												
7												
6												
5												
4												
3												
2												
1												
Y^|X>	1	2	3	4	5	6	7	8	9	10	11	12
	5	9		2	5	6	2	6	5	2		7
	7				6		6	8	8			10
	8				7		7		9			
					8		8		10			
Ymin	3	Xmin	3		9		10		11			
Ymax	12	Xmax	11						12			

Second method : 2D structure

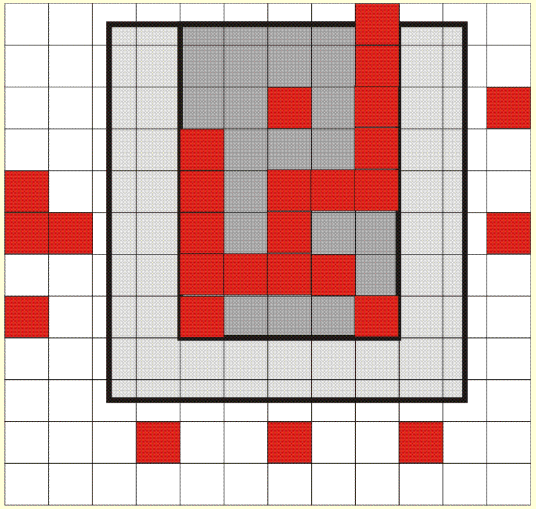
4												
3												
2												
1												
Y^|X>	1	2	3	4	5	6	7	8	9	10	11	12
	5	9		2	5	6	2	6	5	2		7
	7				6		6	8	8			10
	8				7		7		9			
					8		8		10			
Ymin	3	Xmin	3		9		10		11			
Ymax	12	Xmax	11						12			

Second method : 2D structure



	5	9		2	5	6	2	6	5	2		7
	7				6		6	8	8			10
	8				7		7		9			
					8		8		10			
Ymin	3	Xmin	3		9		10		11			
Ymax	12	Xmax	11						12			

third method : restructure to binary tree



re-organize once for all

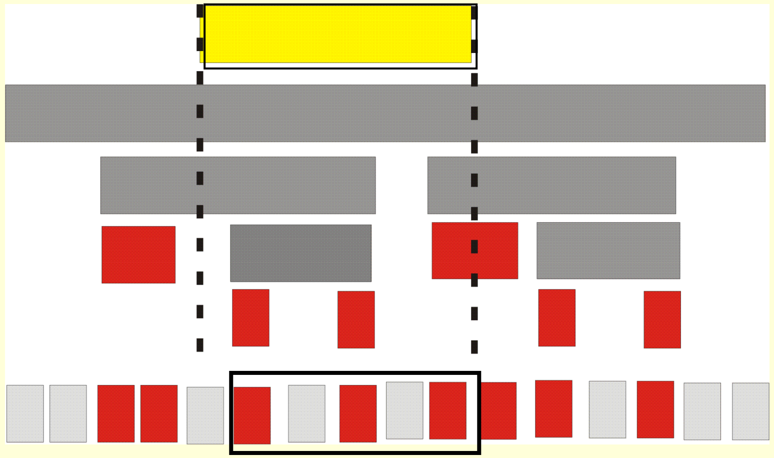
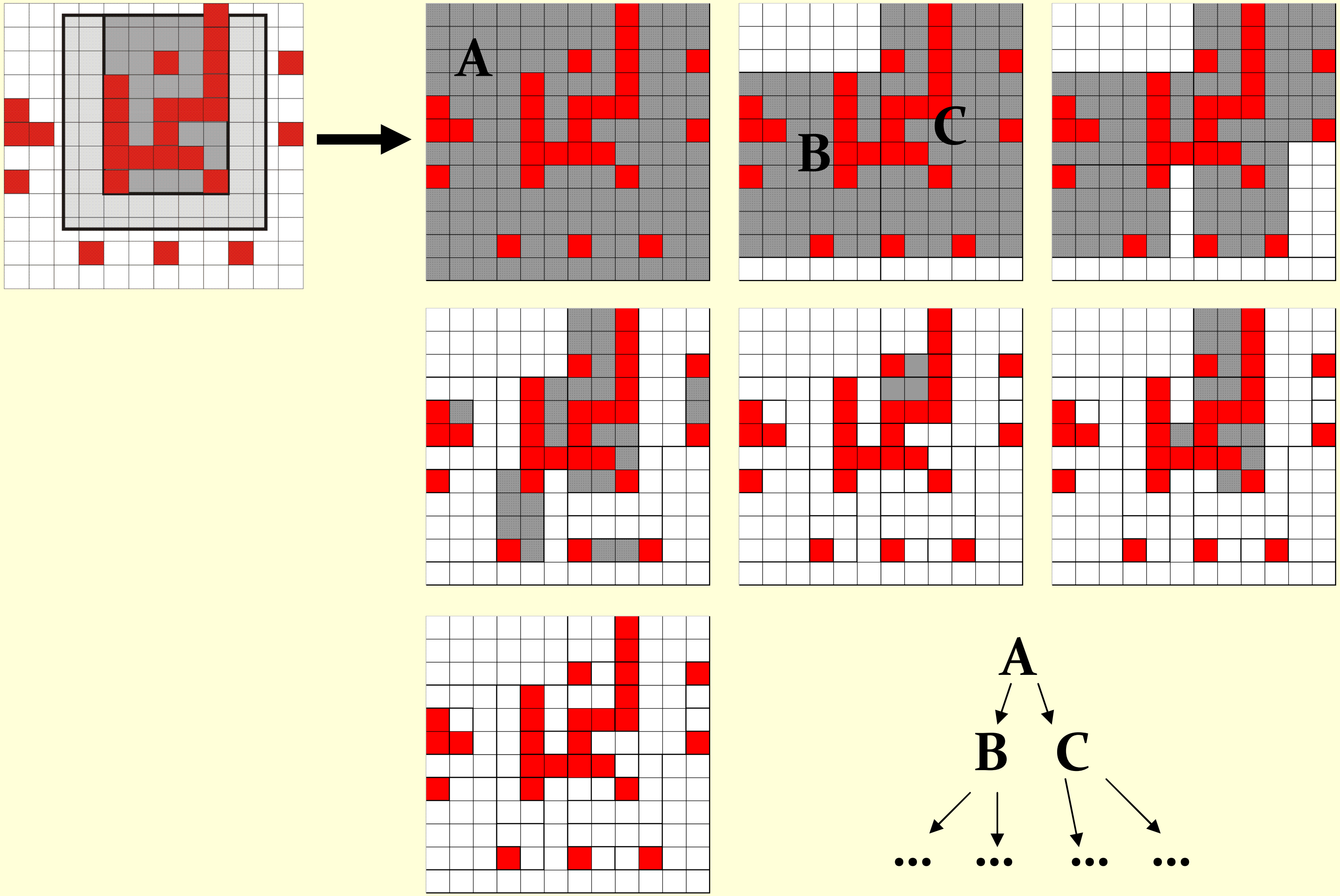


Image and example of contraction

Binary tree representation of an image and example of contraction

third method : binary tree

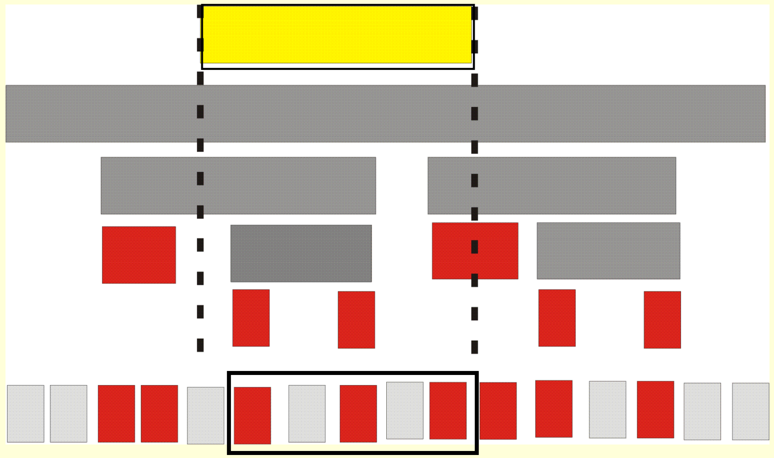


third method : binary tree

The algorithm

Output : Union of boxes

- consecutive inclusion / intersection tests on nodes
- If **inclusion** true add the node / leaf to a list of boxes
- If **intersection** true :
  - If leaf add intersection to list
  - If node apply algorithm on children of the node
- If **exclusion** stop
- End : The result will be the union of those boxes



Binary tree representation of an image and example of contraction

Fourth method :

Consider a binary image

> Continuous representation for easier explanation :

$$f : \mathbb{R}^2 \to \{0,1\}$$

Consider the function

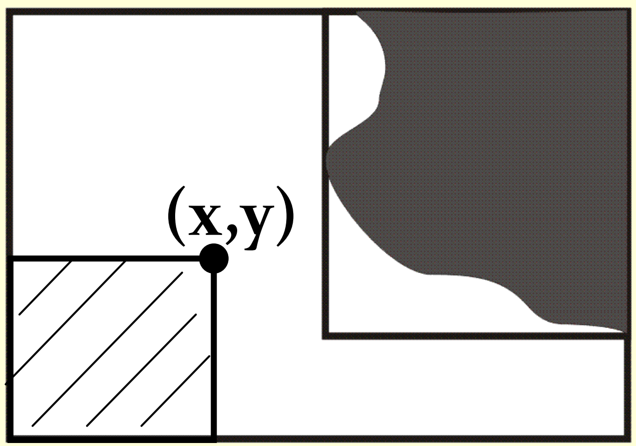
$$\psi : \mathbb{R}^2 \to \mathbb{R}$$

$$\psi(x,y) = \int_0^x \int_0^y f(x,y)dxdy$$

↓

Number of 1-pixels in the box

$$[0,x] \times [0,y]$$



Fourth method :

Consider a binary image $f : \mathbb{R}^2 \rightarrow \{0,1\}$

Consider the function

$$\phi : [\mathbb{R}]^2 \rightarrow \mathbb{R},$$

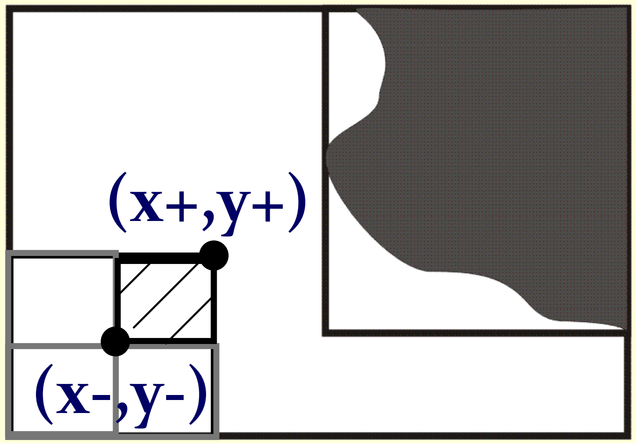
$$\phi([x],[y]) = \int_{(x,y)\in[x][y]} f(x,y)dxdy$$

$$[x] = [x^-,x^+], [y] = [y^-,y^+]$$

$$\phi([x],[y]) = \psi(x^+,y^+) - \psi(x^-,y^+) - \psi(x^+,y^-) + \psi(x^-,y^-)$$

Number of 1-pixels in the box

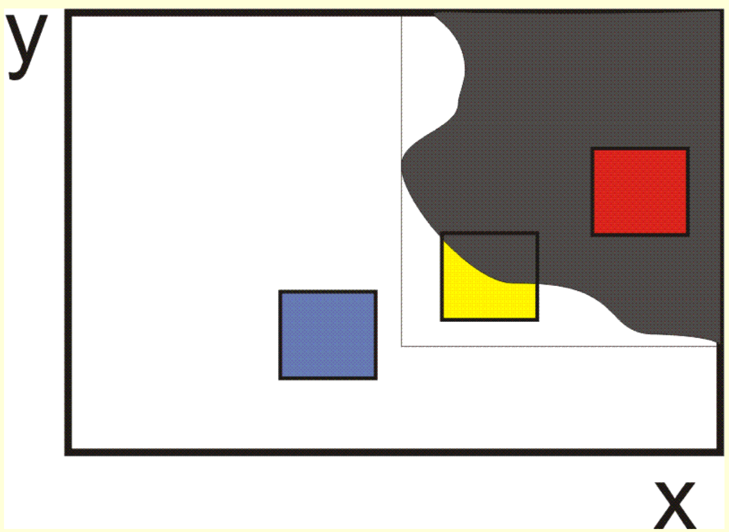
$$[x^-,x^+] \times [y^-,y^+]$$



Fourth method :

Consider a binary image $f : \mathbb{R}^2 \to \{0,1\}$

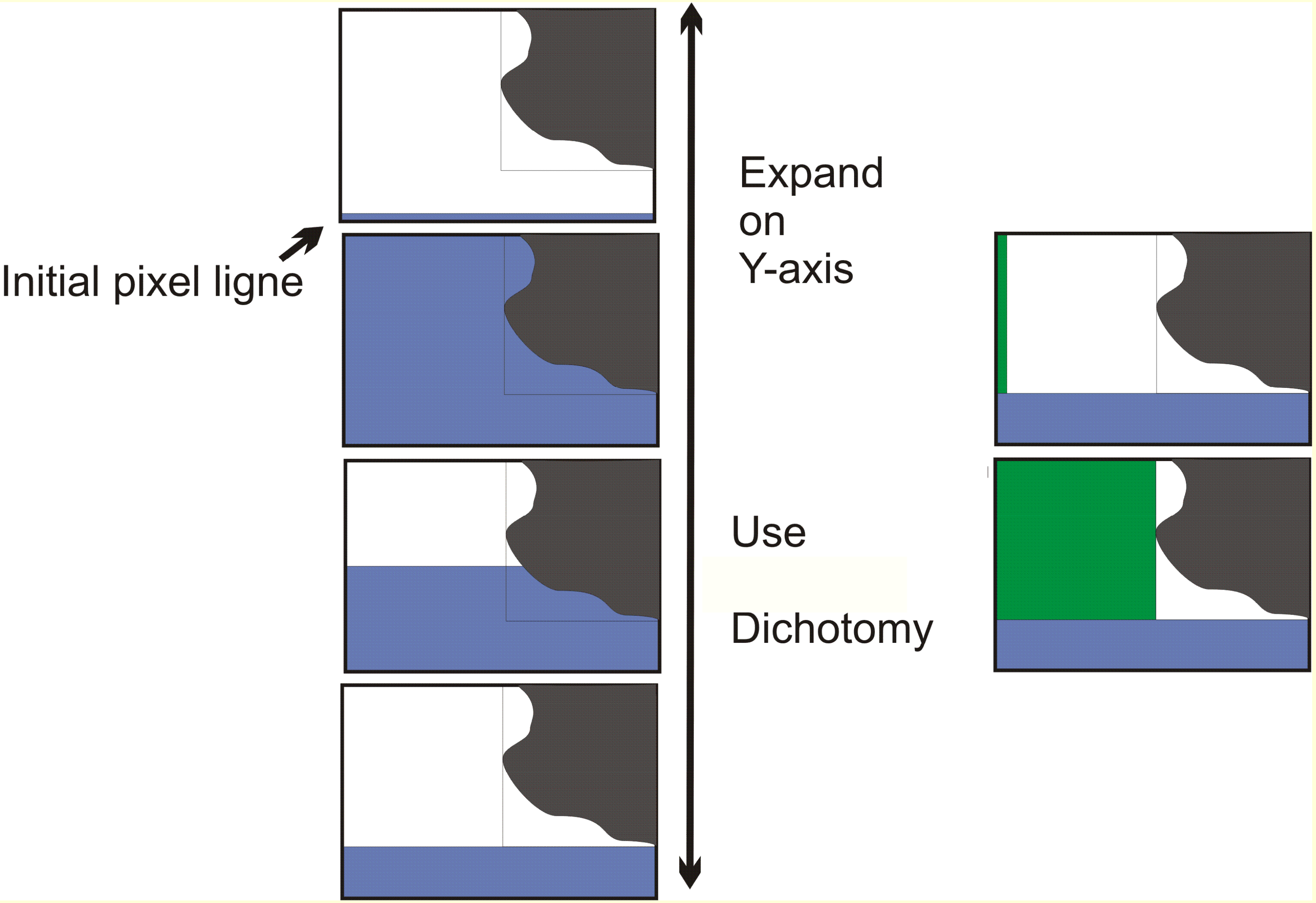
Compute ψ **Once!**



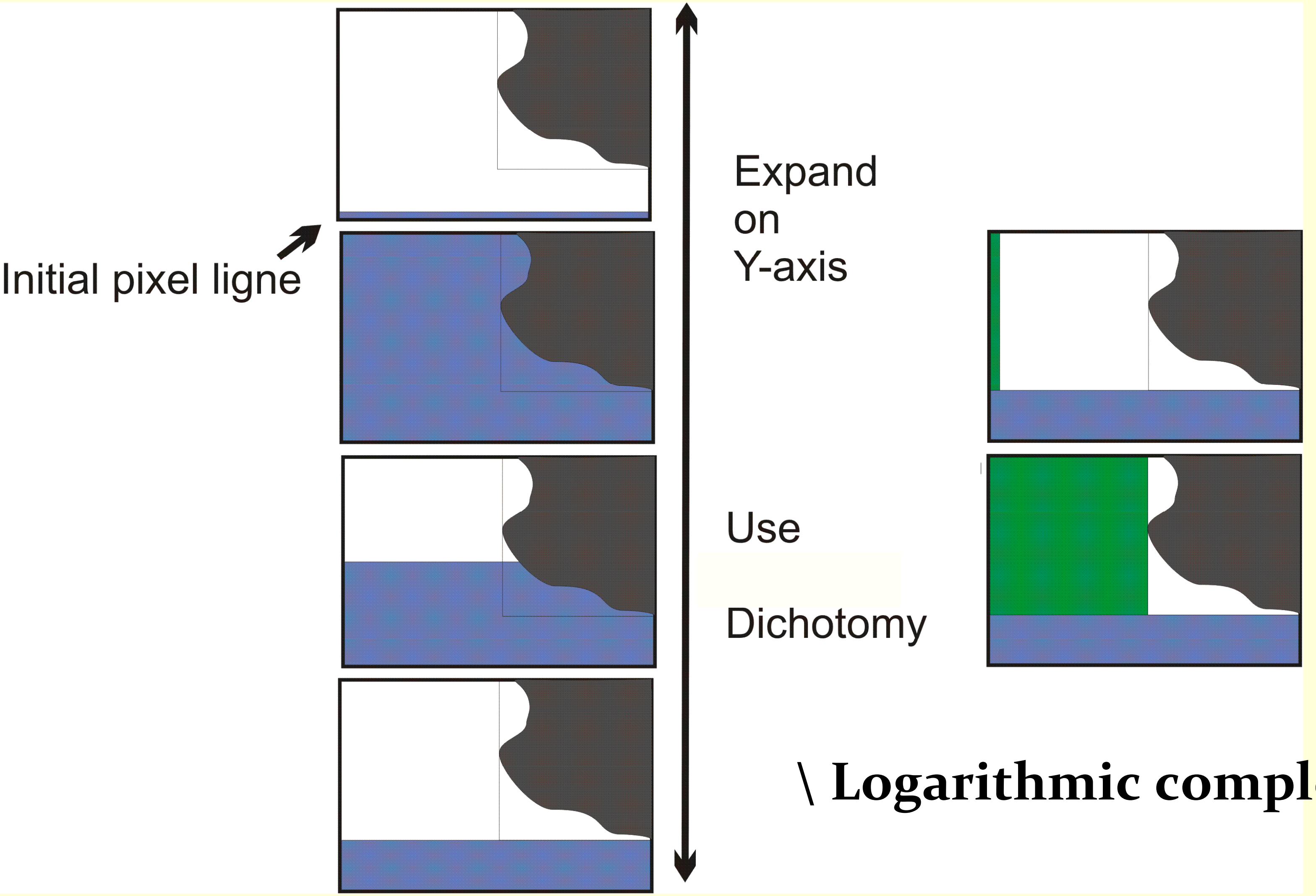
$$\phi([x],[y]) = \psi(x^+,y^+) - \psi(x^-,y^+) - \psi(x^+,y^-) + \psi(x^-,y^-)$$

Number of 1-pixels in the box $[x^-,x^+] \times [y^-,y^+]$

Fourth method : Algorithm



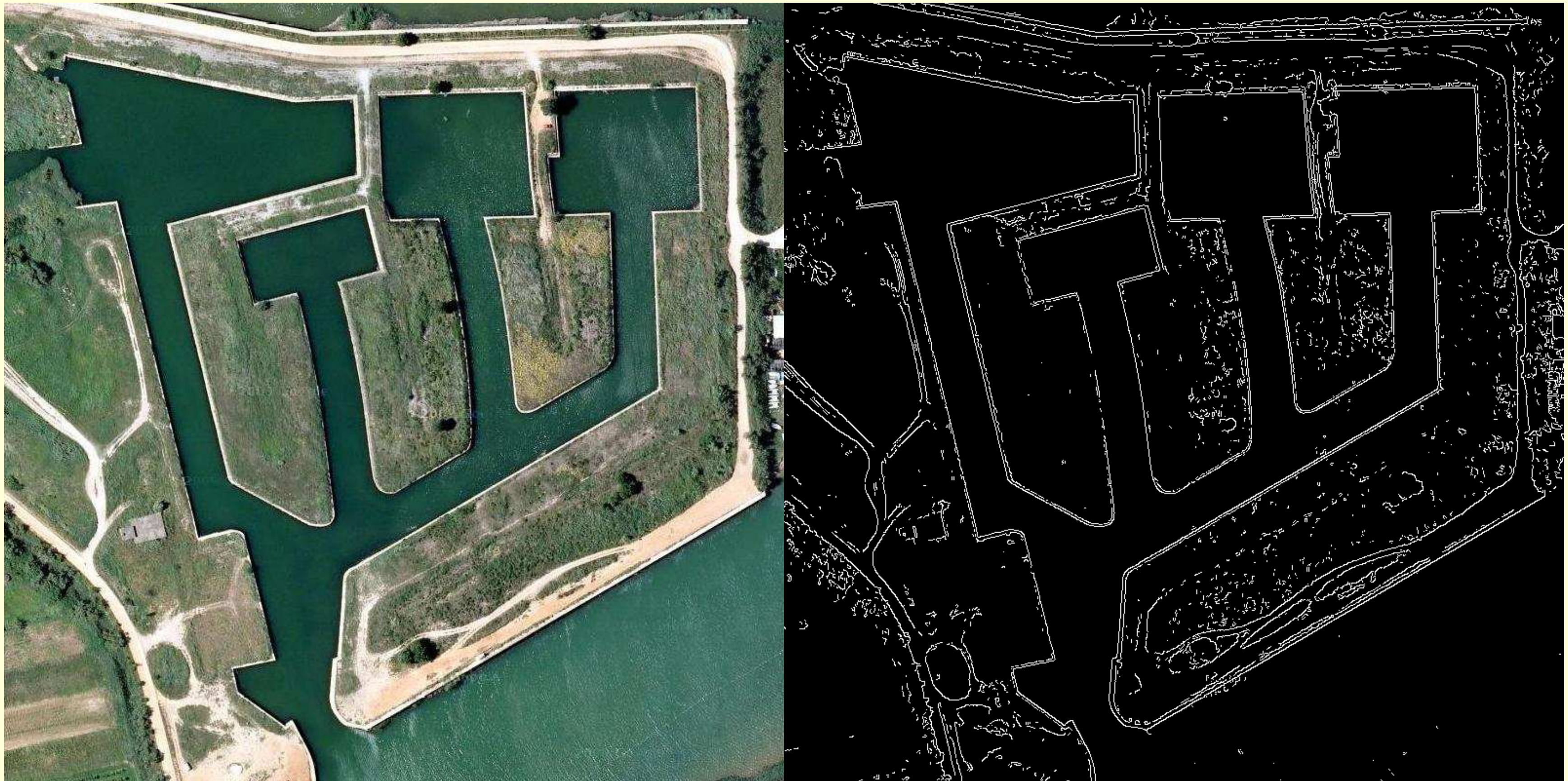
Fourth method : Algorithm



\ Logarithmic complexity /

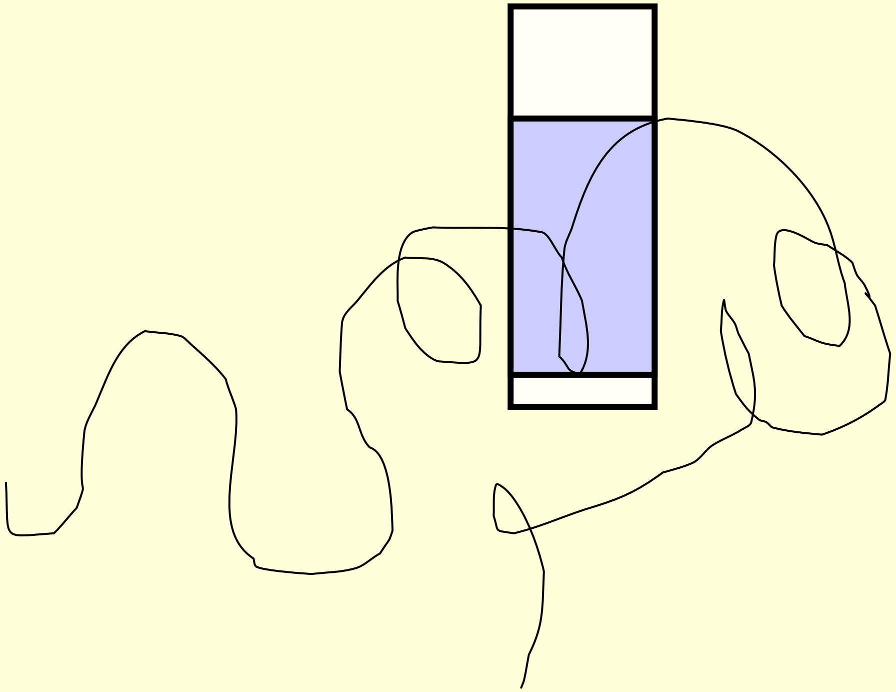
Applications

Localization (OpenCV Canny)



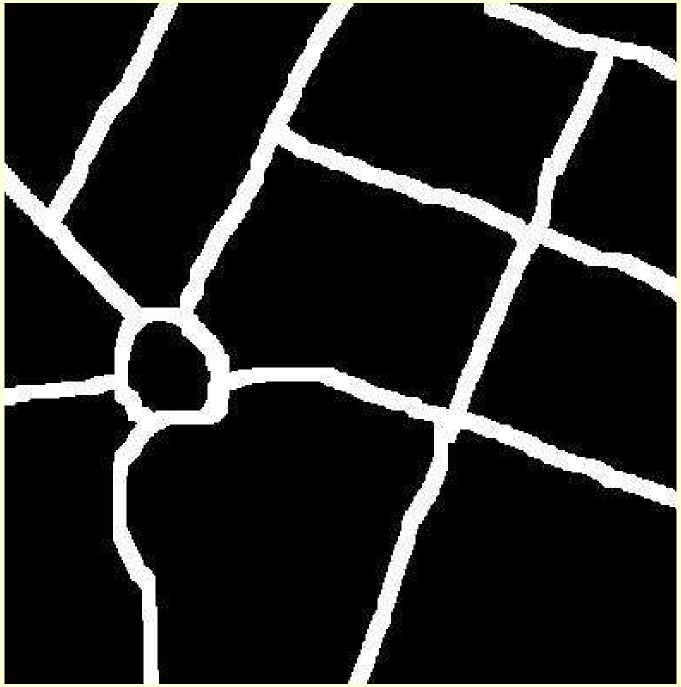
Applications

Normal contractions



Applications

Navigating in the city with **compass** and **speedometer only** (No GPS)



Conclusion

- Image Contractor
- Promising for parallel computing hardware implementation (use Integers)
- Logarithmic complexity (on PC)
- Some real applications in localization or simply any contractions.
- The main drawback is storage

?Questions?