

# Computer Vision

## Gradient and Edge Detection

Fátima Leal

DCT DEPARTAMENTO CIÊNCIA  
E TECNOLOGIA

# What we have learnt

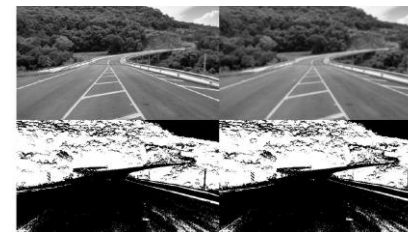
- Low and High pass filters

- Average Filtering
- Gaussian Filtering
- Non-linear Filtering
- Laplacian Filter



- Image segmentation: Thresholding

- Simple Thresholding
- Adaptive Thresholding
- Otsu



# Content

- Gradient detection
- Edge detection

# Gradients and Edge detection

- One of the most important tasks in computer vision is to recognize objects
- There are a set of techniques which allow to detect edges and identify objects
- Edge detector
  - Sobel gradient representation
  - Prewitt gradient representation
  - Laplacian gradient representation
  - Canny edge detector

# Feature detection

- What kinds of features should we detect?
  - Specific locations in the images: mountain peaks, building corners, doorways, or interestingly shaped patches of snow.



(a)



(b)



(c)



(d)

How to do that?

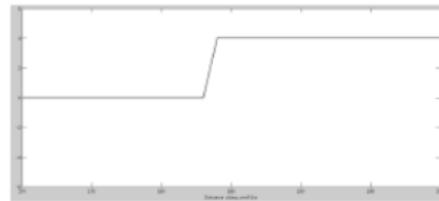
We are interested in points, lines or edges detection

# Gradient

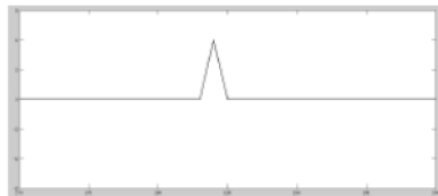
- Formally, points, lines or edges are an abrupt transition in intensity values between two regions.
- From calculus, we know that the changes in intensity can be measured by using the first or second derivative.



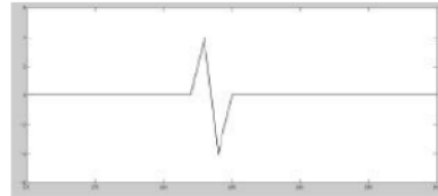
(a) Input image.



(b) Intensity profile.



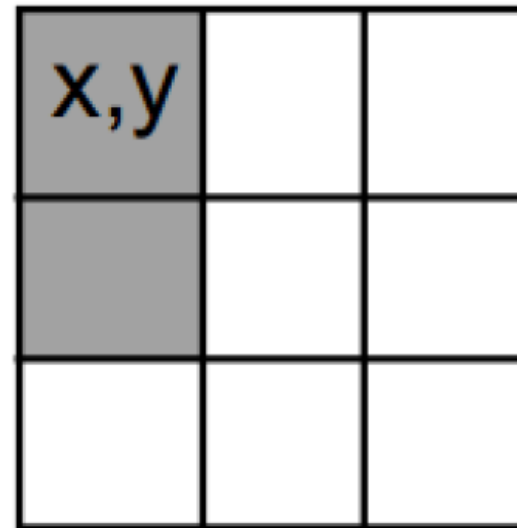
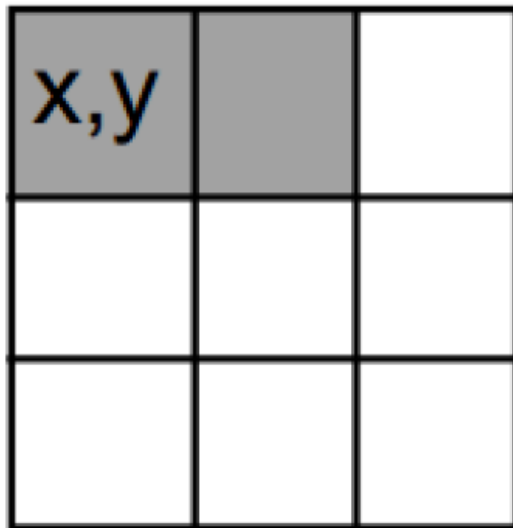
(c) First derivative profile.



(d) Second derivative profile.

# Gradient Detection

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y) \quad \frac{\partial f}{\partial y} = f(x, y+1) - f(x, y)$$



# Gradient Detection

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) + 2f(x, y)$$

$$\frac{\partial^2 f}{\partial y^2} = f(x, y-2) + f(x, y-1) + 2f(x, y)$$

x,y		

x,y		



# Gradient Detection - Noise

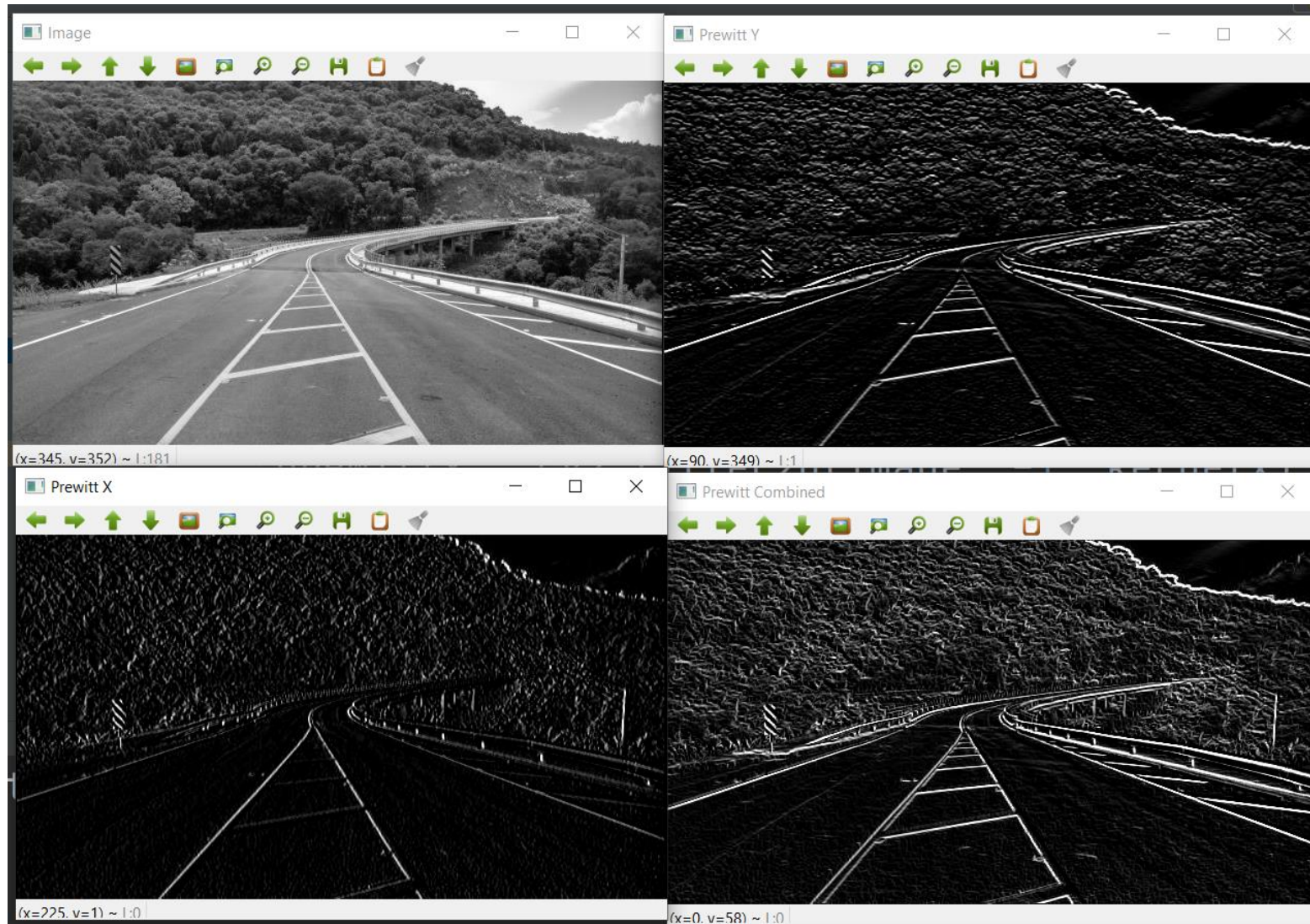
- The noise can have a lot of influence in the results
- The process of edge detection consists of three main steps:
  - Noise reduction: smoothing techniques.
  - Detection of edge points.
  - Edge localization

# Sobel Gradient Representation

- Using the Sobel operator, we can compute gradient magnitude representations along the x and y axis, allowing us to find both horizontal and vertical edge-like regions.



# Prewitt Gradient Representation





# Laplacian Gradient Representation

- The Laplacian gradient does not require the processing horizontal and vertical.



# Sobel, Prewitt and Laplacian

## Sobel Filter

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

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

## Prewitt Filter

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

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

## Laplacian Filter

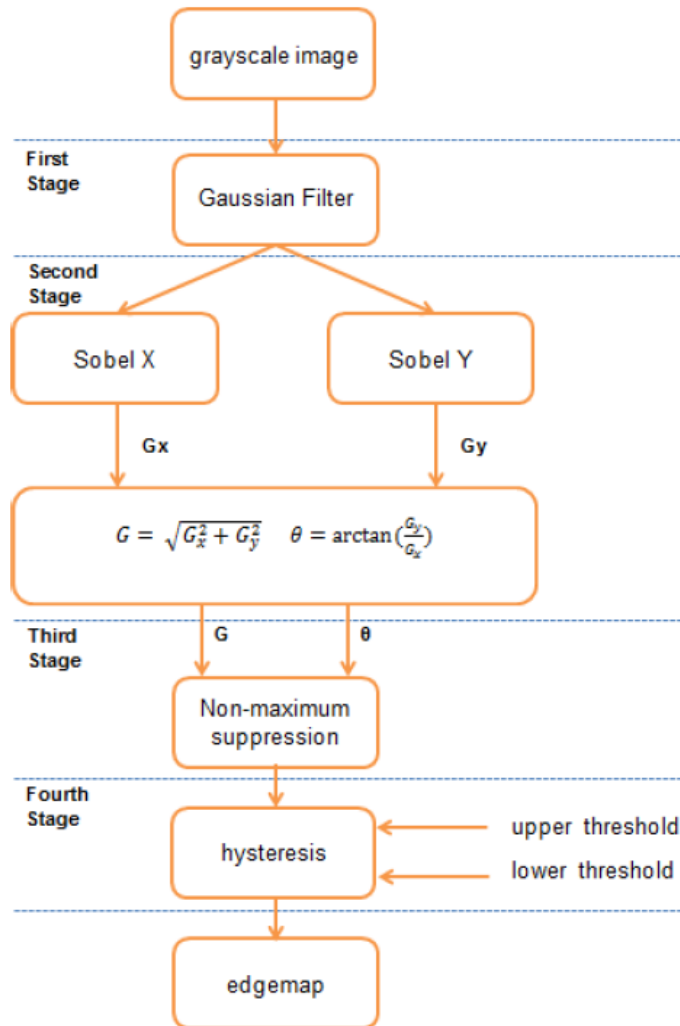
0	1	0
-1	4	-1
0	1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

# Canny edge detection

- One thing you'll notice in the previous subsections is that the edges are very “noisy”. They are not clean and crisp.
- We'll remedy that by using the Canny edge detector.
- The Canny edge detector is a multi-step process:
  - blurring the image to remove noise,
  - computing Sobel gradient images in the x and y direction
  - suppressing edges
  - Hysteresis thresholding stage that determines if a pixel is “edge-like” or not

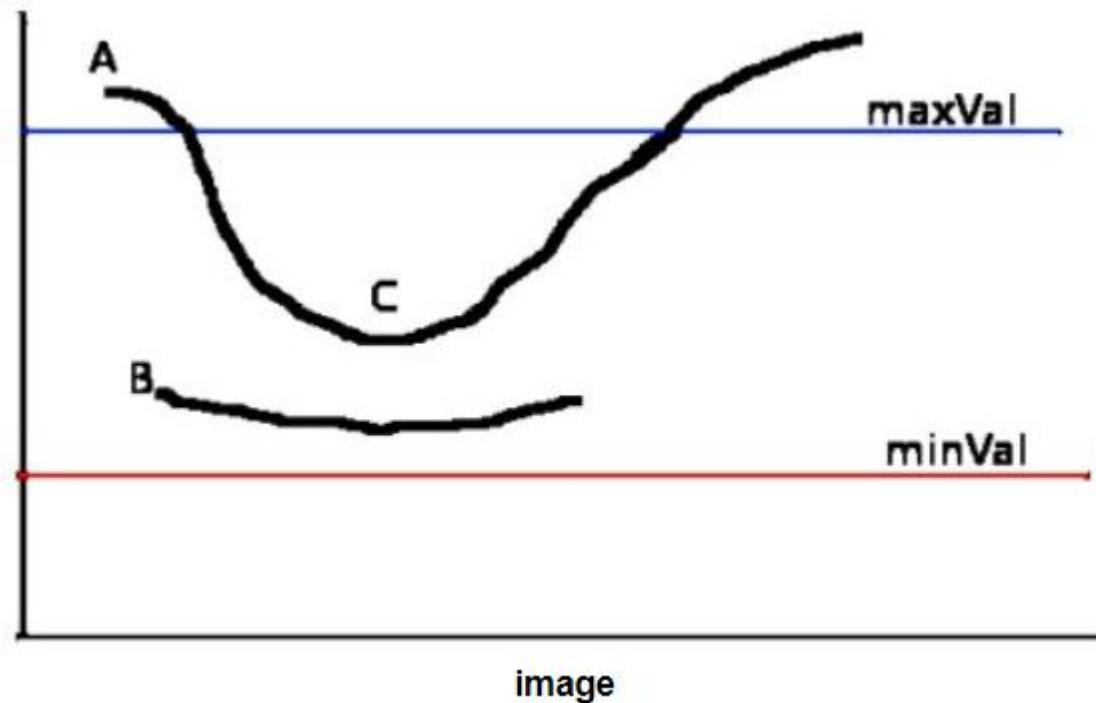
# Canny edge detection



1. Any pixel in the image that has a value greater than T1 is presumed to be an edge pixel, and is marked as such immediately.

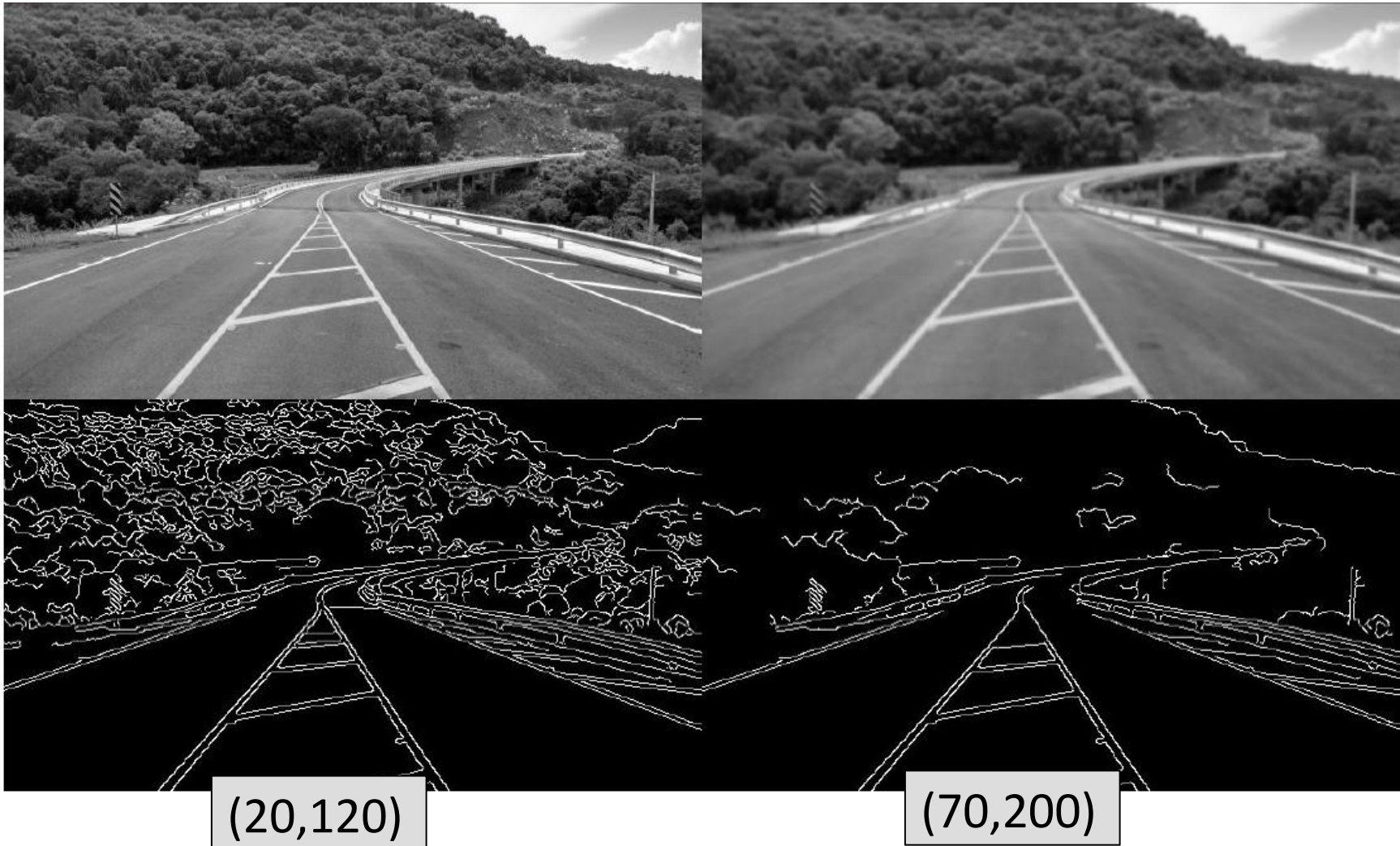
2. Any pixels that are connected to this edge pixel and that have a value greater than T2 are also selected as edge pixels.

- A and C are edges. C is because it is connected to A





# Canny edge detection



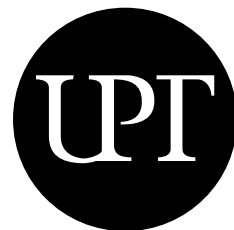
# Counters



`cv2.findContours()`

# Let's play with images!





UNIVERSIDADE  
PORTUCALENSE

Do conhecimento à prática.