



Tratamiento de Señales

Version 2023-2

PARTE 2

Detección de Bordes

[Capítulo 8]

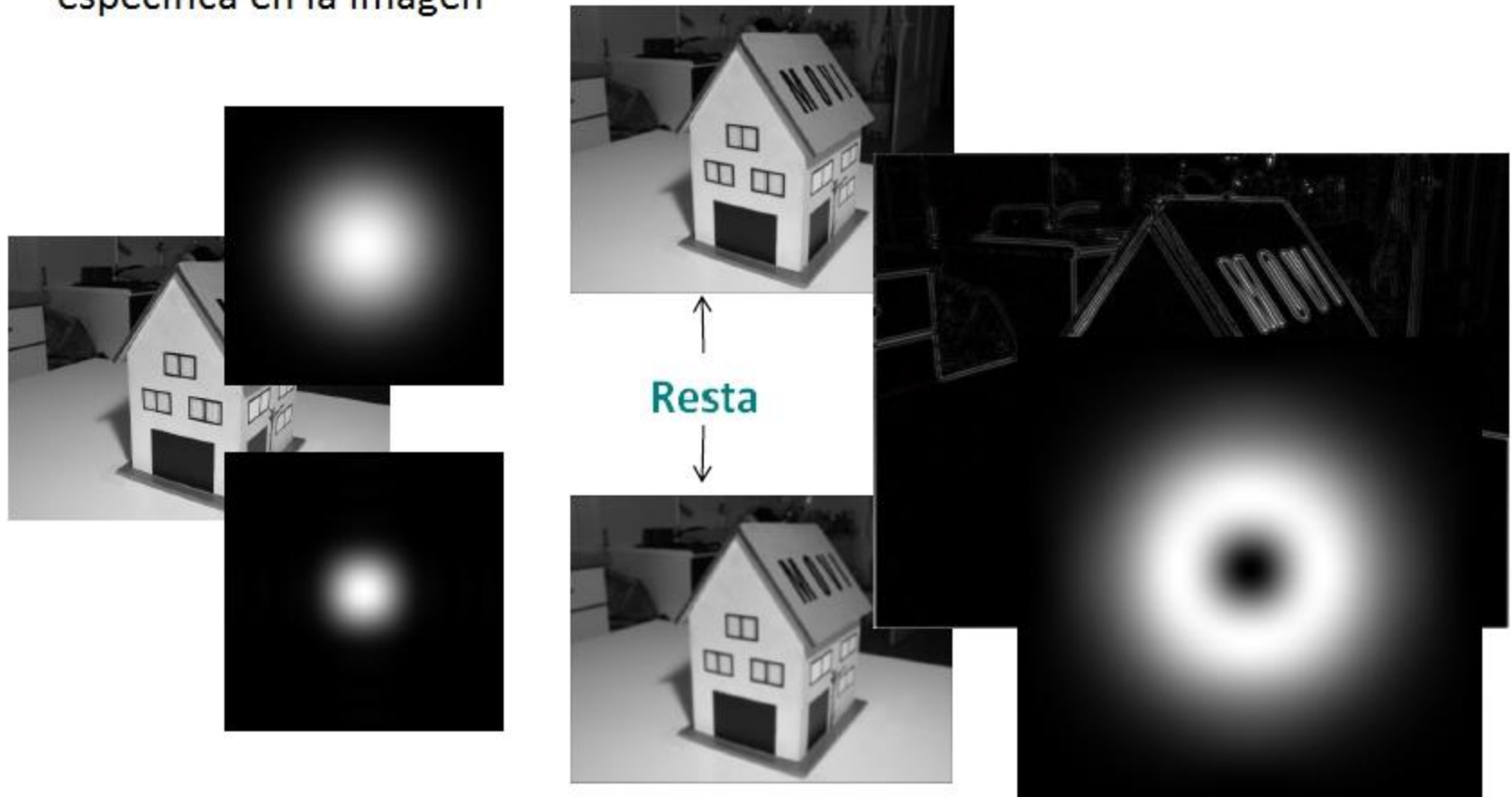
Dr. José Ramón Iglesias

DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingeniería Electronica
Universidad Popular del Cesar

Detector de Bordes usando
Laplacian of Gaussian (LoG)

Filtrado pasa-banda: DOG

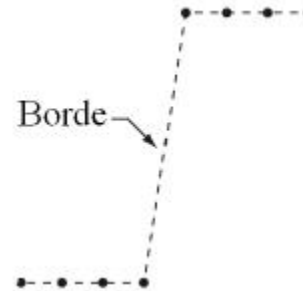
Idea: capturar información sobre una banda de frecuencia específica en la imagen



Filtrado pasa-altas (Cont.)

Detección de bordes

Ejemplo. Un borde vertical.



Como detectarlo?

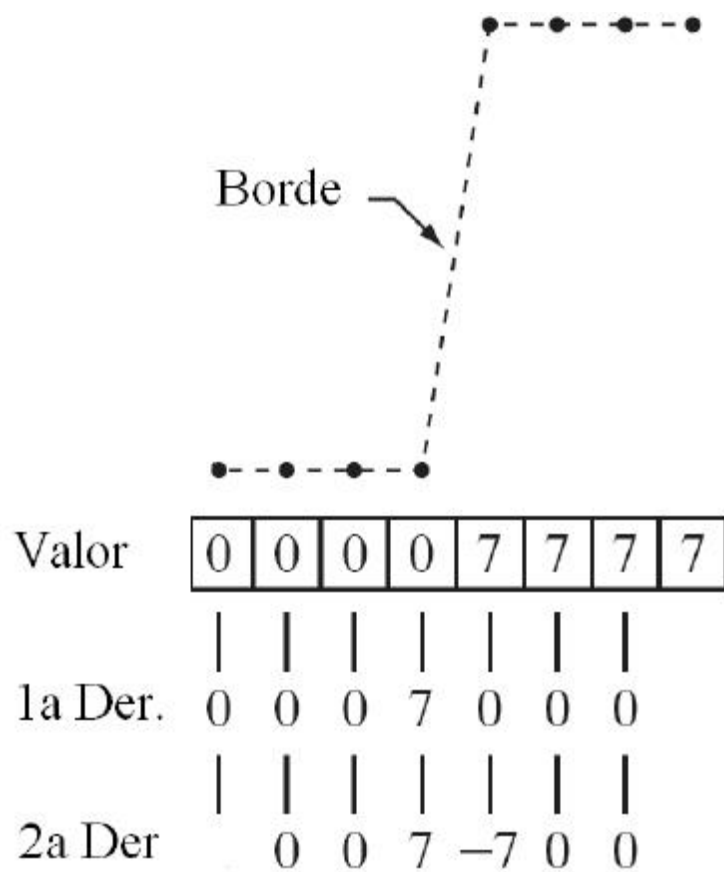
R/ Eliminando las componentes de baja frecuencia.

- Primera derivada

$$\frac{df(x)}{dx} = \underbrace{f(x+1) - f(x)}_{\text{Aprox. discreta}}$$

- Segunda derivada

$$\frac{d^2 f(x)}{dx^2} = \underbrace{f(x+1) - 2f(x) + f(x-1)}_{\text{Aprox. discreta}}$$



Máscaras para detectar bordes

Derivada	Bordes	Máscara
1	Verticales	$\begin{bmatrix} -1 & 1 \end{bmatrix}$
1	Horizontales	$\begin{bmatrix} -1 & 1 \end{bmatrix}^T$
2	Verticales	$\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$
2	Horizontales	$\begin{bmatrix} 1 & -2 & 1 \end{bmatrix}^T$

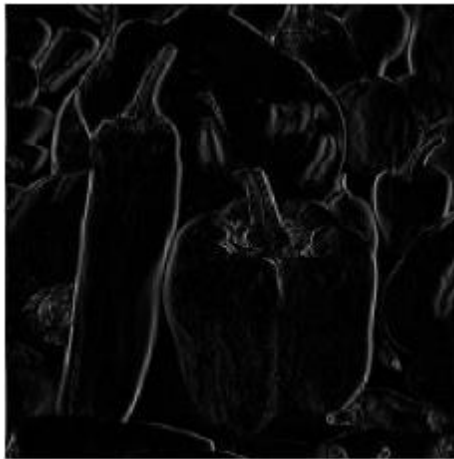
Ejemplo. Detección de bordes verticales y horizontales.

Imagen
original

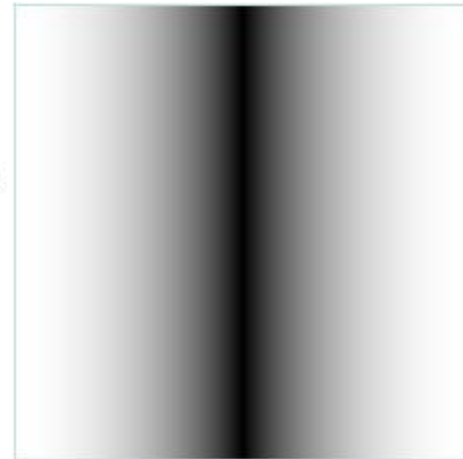


- Primera derivada horizontal (**bordes verticales**)

Imagen
Filtrada

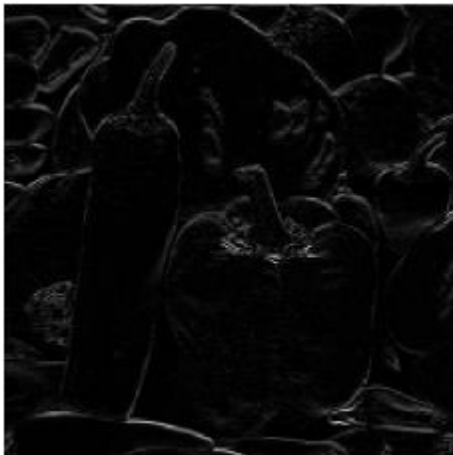


Respuesta en
Frecuencia

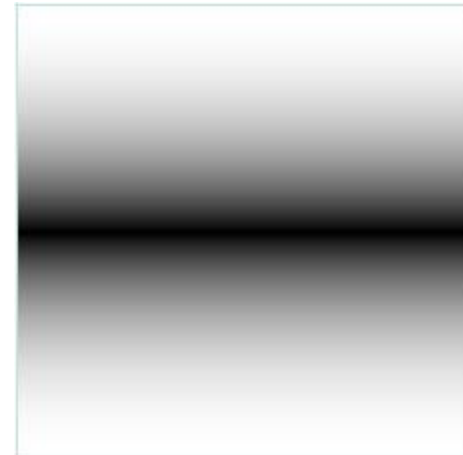


- Primera derivada vertical (**bordes horizontales**)

Imagen
Filtrada

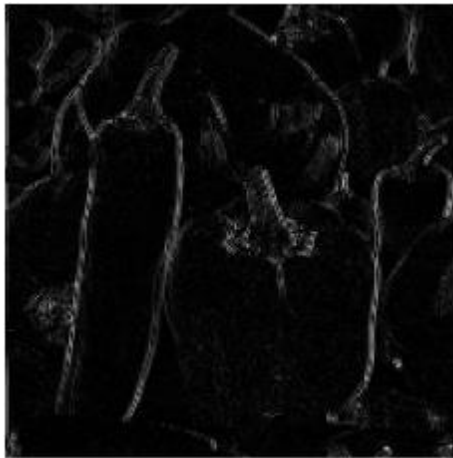


Respuesta en
Frecuencia

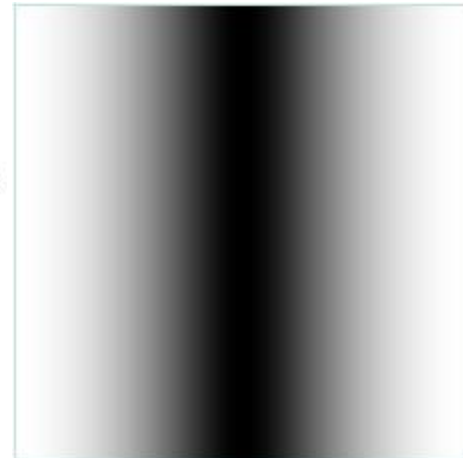


- Segunda derivada horizontal (**bordes verticales**)

Imagen
Filtrada

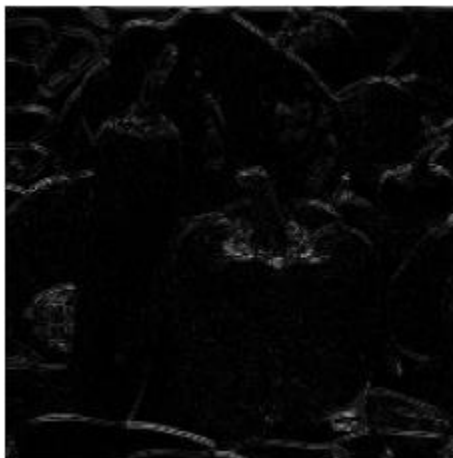


Respuesta en
Frecuencia

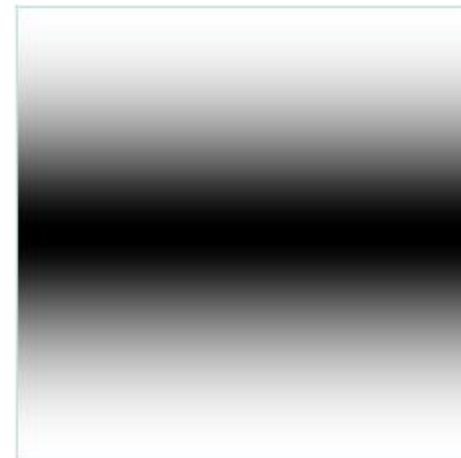


- Segunda derivada vertical (**bordes horizontales**)

Imagen
Filtrada

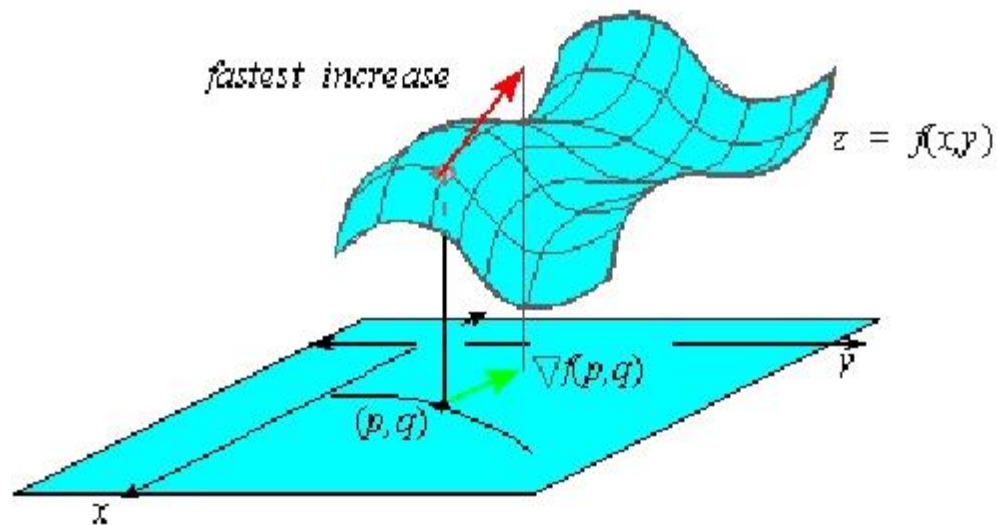


Respuesta en
Frecuencia



Gradiente de una imagen

$$\nabla f = \left[\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right]$$



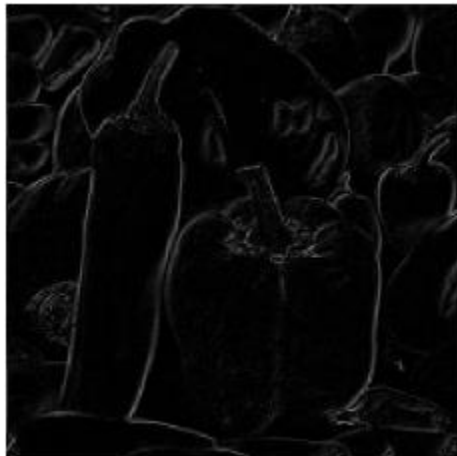
Gradiente de una imagen

$$\nabla f = \left[\frac{\partial f(x,y)}{\partial x}, \frac{\partial f(x,y)}{\partial y} \right]$$

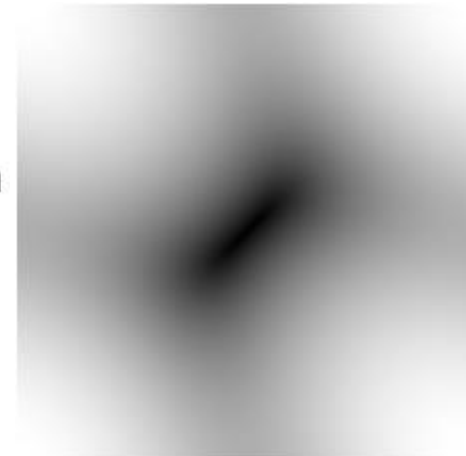
$$|\nabla f| = \sqrt{\frac{\partial f(x,y)}{\partial x}^2 + \frac{\partial f(x,y)}{\partial y}^2} \approx \left| \frac{\partial f(x,y)}{\partial x} \right| + \left| \frac{\partial f(x,y)}{\partial y} \right| \approx \left| \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y} \right|$$

$$\frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y} = f(x+1) + f(y+1) - 2f(x) \quad \text{Máscara} \quad \begin{bmatrix} -2 & 1 \\ 1 & 0 \end{bmatrix}$$

Imagen
Filtrada



Respuesta en
Frecuencia



Detecta bordes horizontales y verticales!

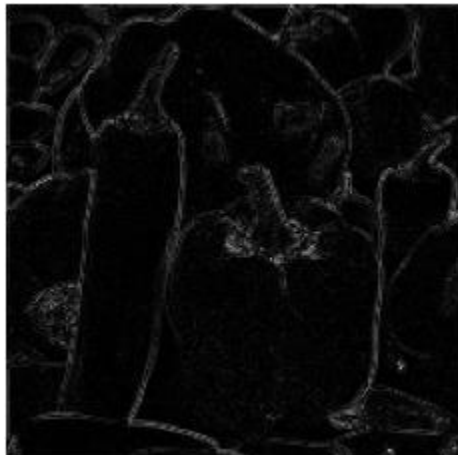
- Laplaciano

$$\begin{aligned}\nabla^2 f &= \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \\ &= [f(x+1, y) - 2f(x, y) + f(x-1, y)] + [f(x, y+1) - 2f(x, y) - f(x, y-1)] \\ &= f(x+1, y) + f(x, y+1) - 4f(x, y) + f(x-1, y) + f(x, y-1)\end{aligned}$$

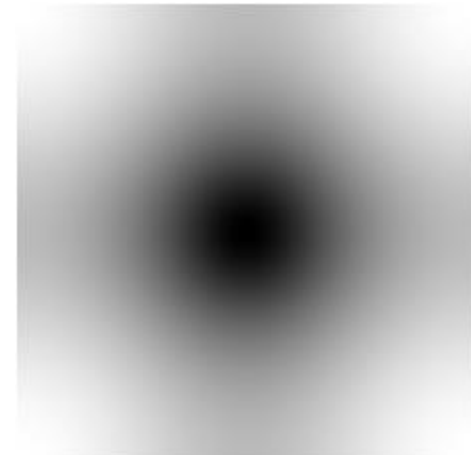
Máscara

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Imagen
Filtrada



Respuesta en
Frecuencia



Inconveniente de la **aproximación hacia atrás**

La derivada se computa en la mitad de los dos píxeles

$$\begin{array}{|c|c|} \hline -1 & 1 \\ \hline \end{array}$$

Solución. Utilizar la aproximación central

$$\frac{df(x)}{dx} \approx \underbrace{\frac{f(x+1) - f(x-1)}{2}}_{\text{Aprox. discreta}}$$

$$\begin{array}{|c|c|c|} \hline -1/2 & 0 & 1/2 \\ \hline \end{array}$$

Máscaras para detectar bordes

Derivada	Bordes	Máscara
1	Verticales	$\begin{bmatrix} -1/2 & 0 & 1/2 \end{bmatrix}$
1	Horizontales	$\begin{bmatrix} -1/2 & 0 & 1/2 \end{bmatrix}^T$

Problema. Una imagen ruidosa puede producir **falsos bordes**.

Idea. **Suavizar** la imagen antes de detectar los bordes.

OPERADOR DE SOBEL

$$\nabla f = [G_x, G_y]$$

$$G_x = \begin{bmatrix} -1/2 & 0 & 1/2 \end{bmatrix} * \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

bordes verticales

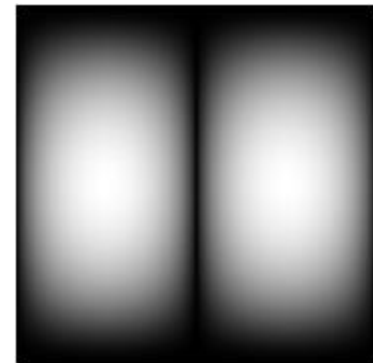
suavizado vertical

$$G_y = \begin{bmatrix} -1/2 \\ 0 \\ 1/2 \end{bmatrix} * \begin{bmatrix} 2 & 4 & 2 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

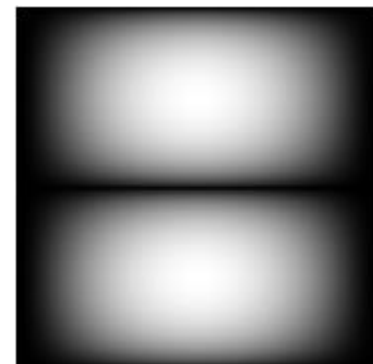
bordes horizontales

suavizado horizontal

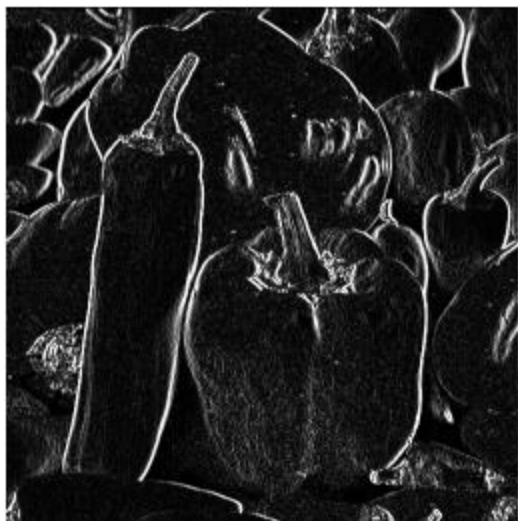
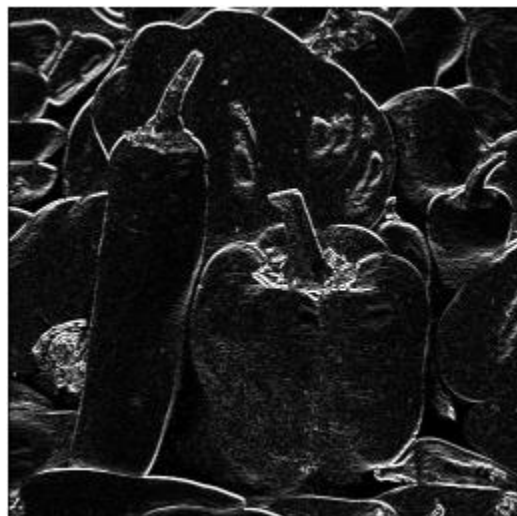
Respuesta en
Frecuencia



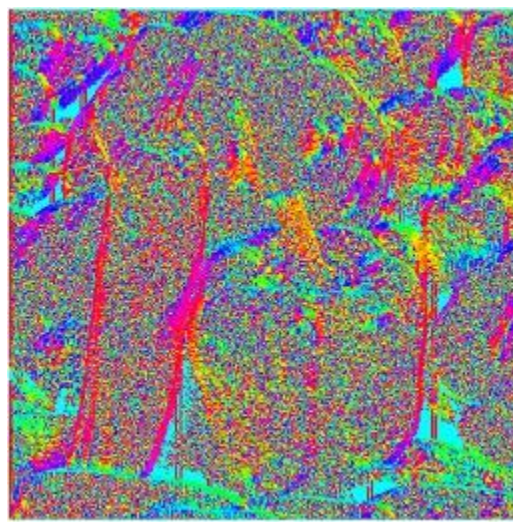
Respuesta en
Frecuencia



`h=fspecial('sobel');`

G_x  G_y 

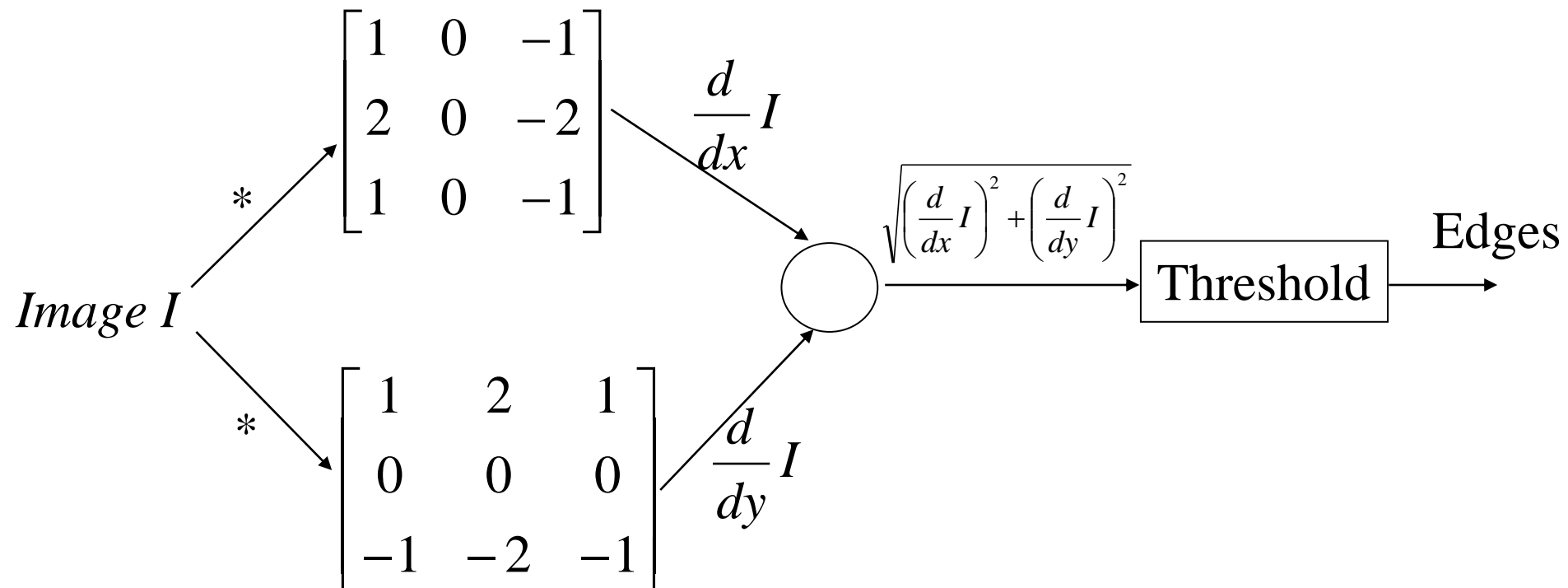
$$|\nabla f| = \sqrt{G_x^2 + G_y^2}$$



$$\angle \nabla f = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

Detecting Edges in Image

- Sobel Edge Detector



Sobel Edge Detector

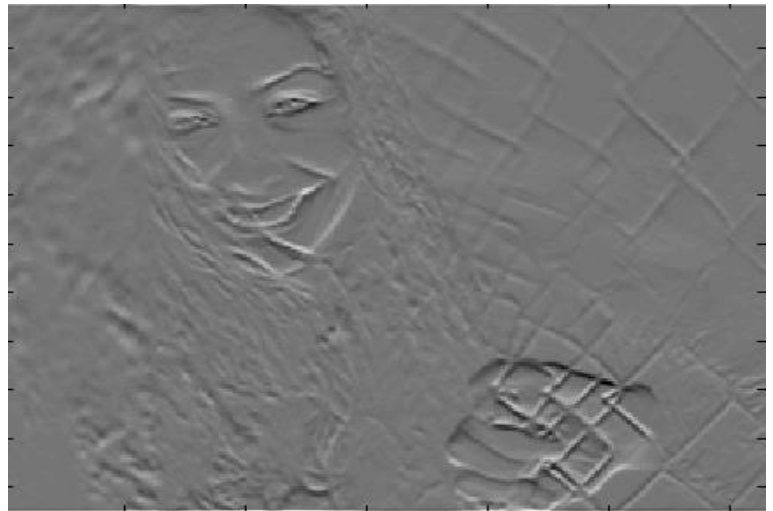
I



$\frac{d}{dx} I$



$\frac{d}{dy} I$



Sobel Edge Detector

I

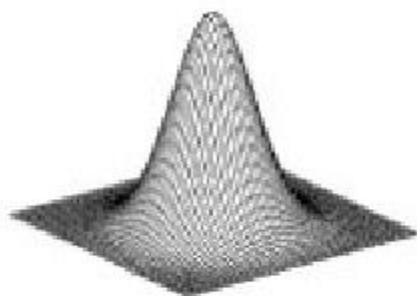


$$\Delta = \sqrt{\left(\frac{d}{dx} I\right)^2 + \left(\frac{d}{dy} I\right)^2}$$

$$\Delta \geq \textit{Threshold} = 100$$

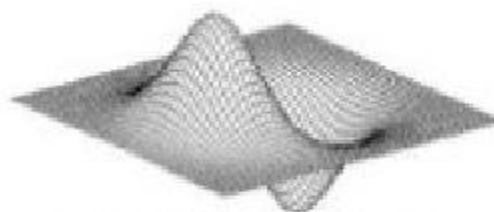


2D edge detection filters



Gaussian

$$h_{\sigma}(u, v) = \frac{1}{2\pi\sigma^2} e^{-\frac{u^2+v^2}{2\sigma^2}}$$



derivative of Gaussian

$$\frac{\partial}{\partial x} h_{\sigma}(u, v)$$

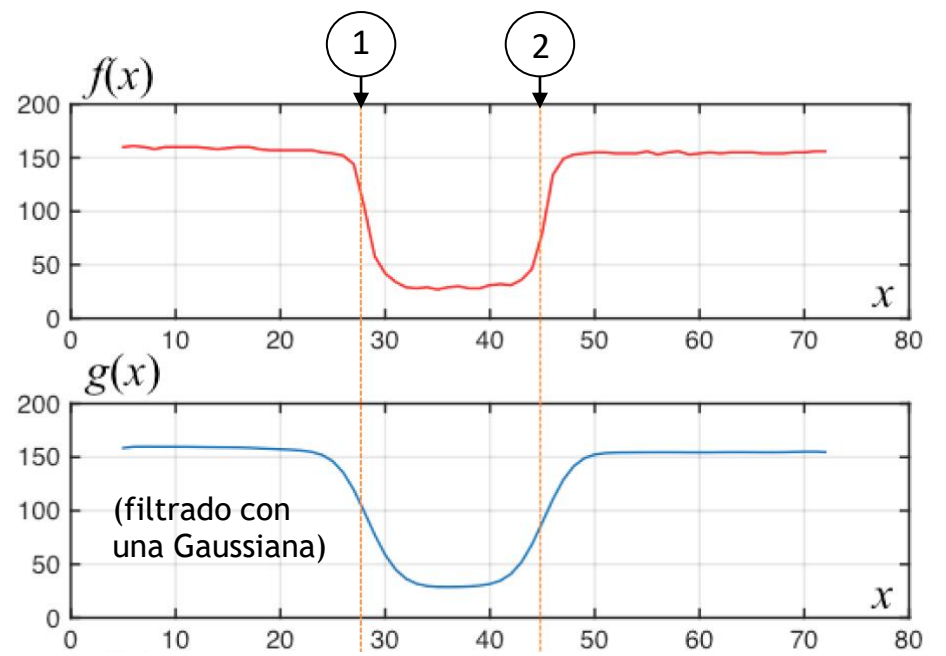
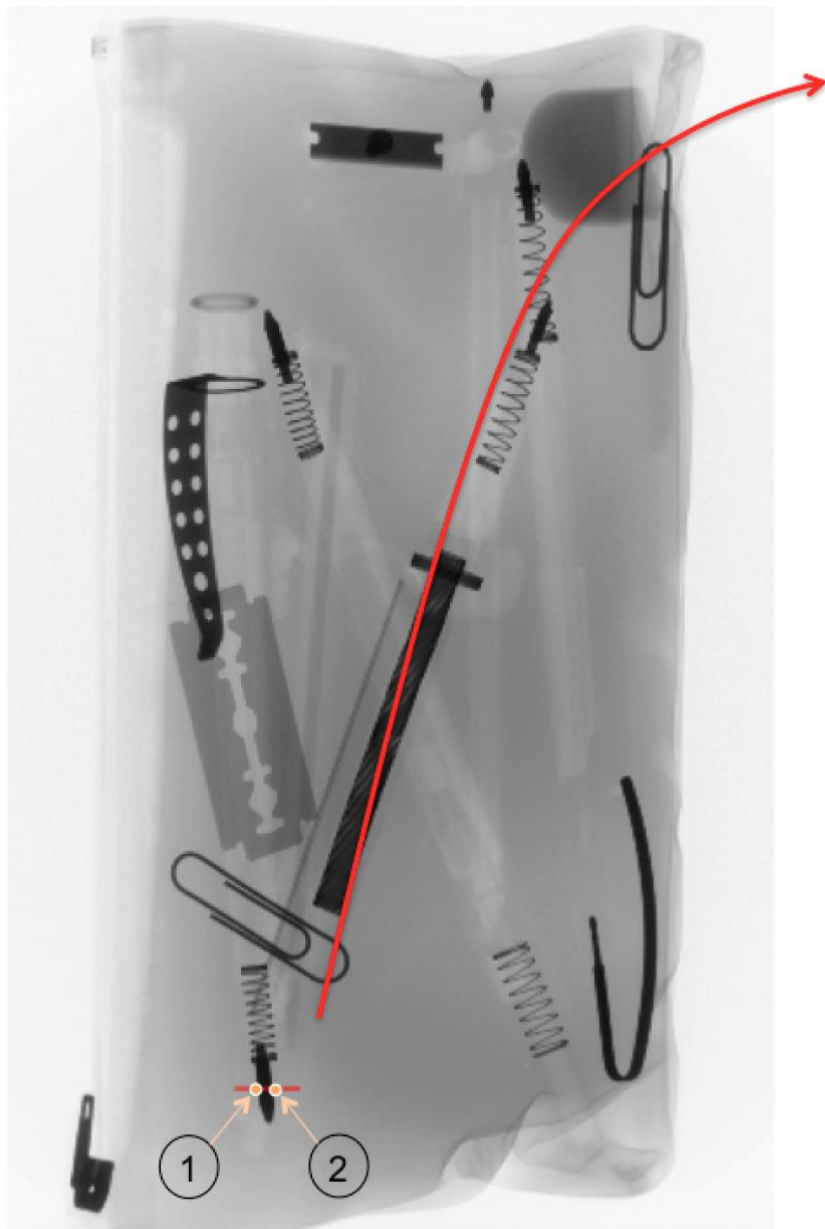
Laplacian of Gaussian

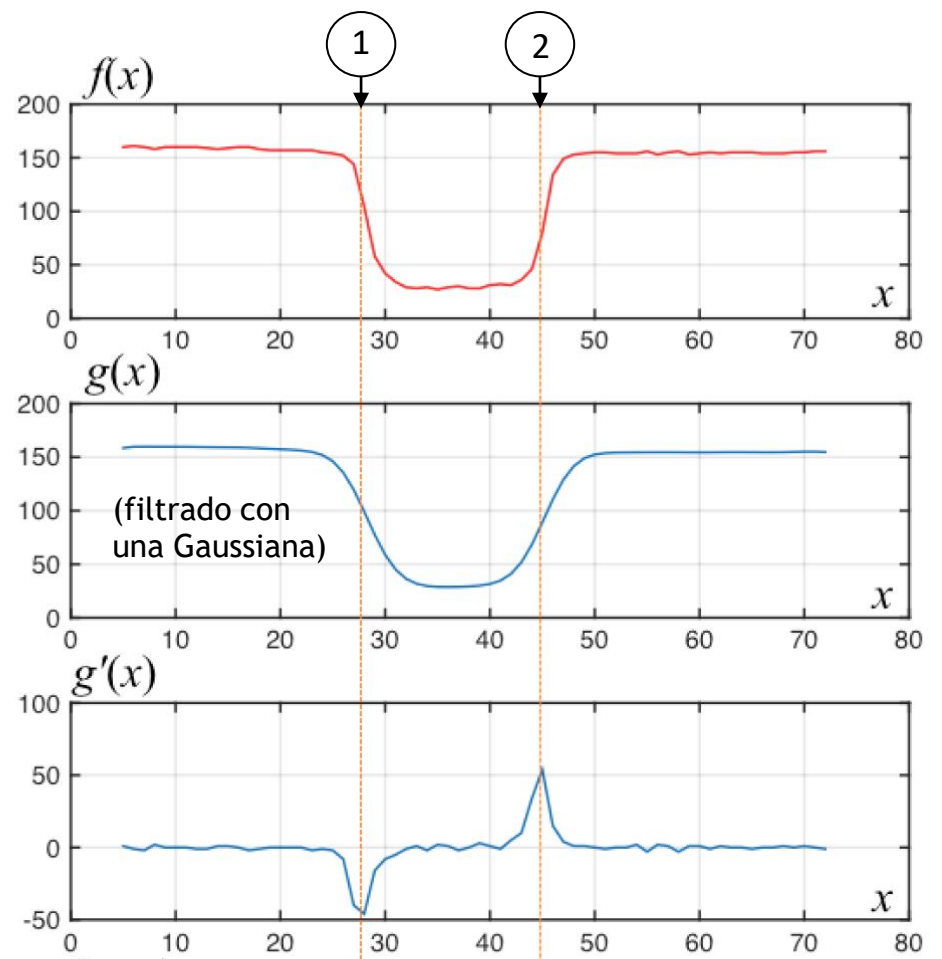
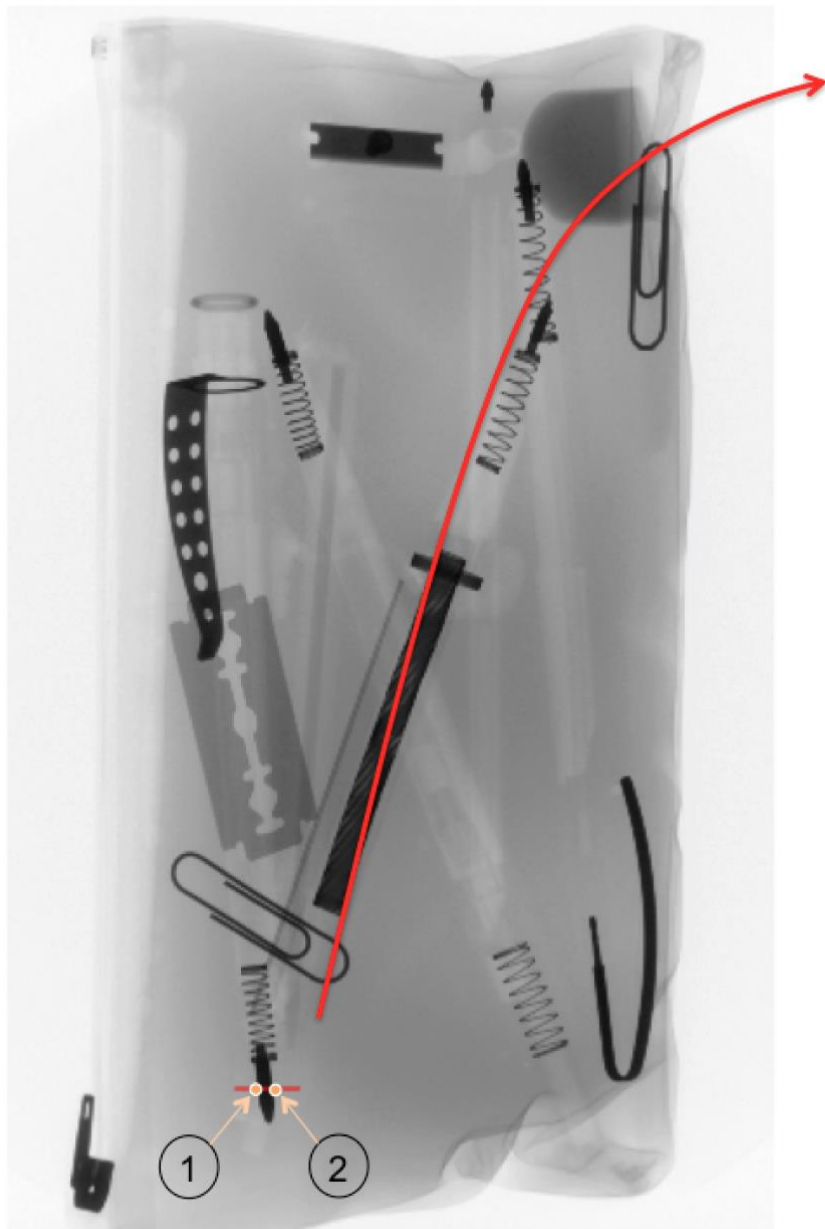


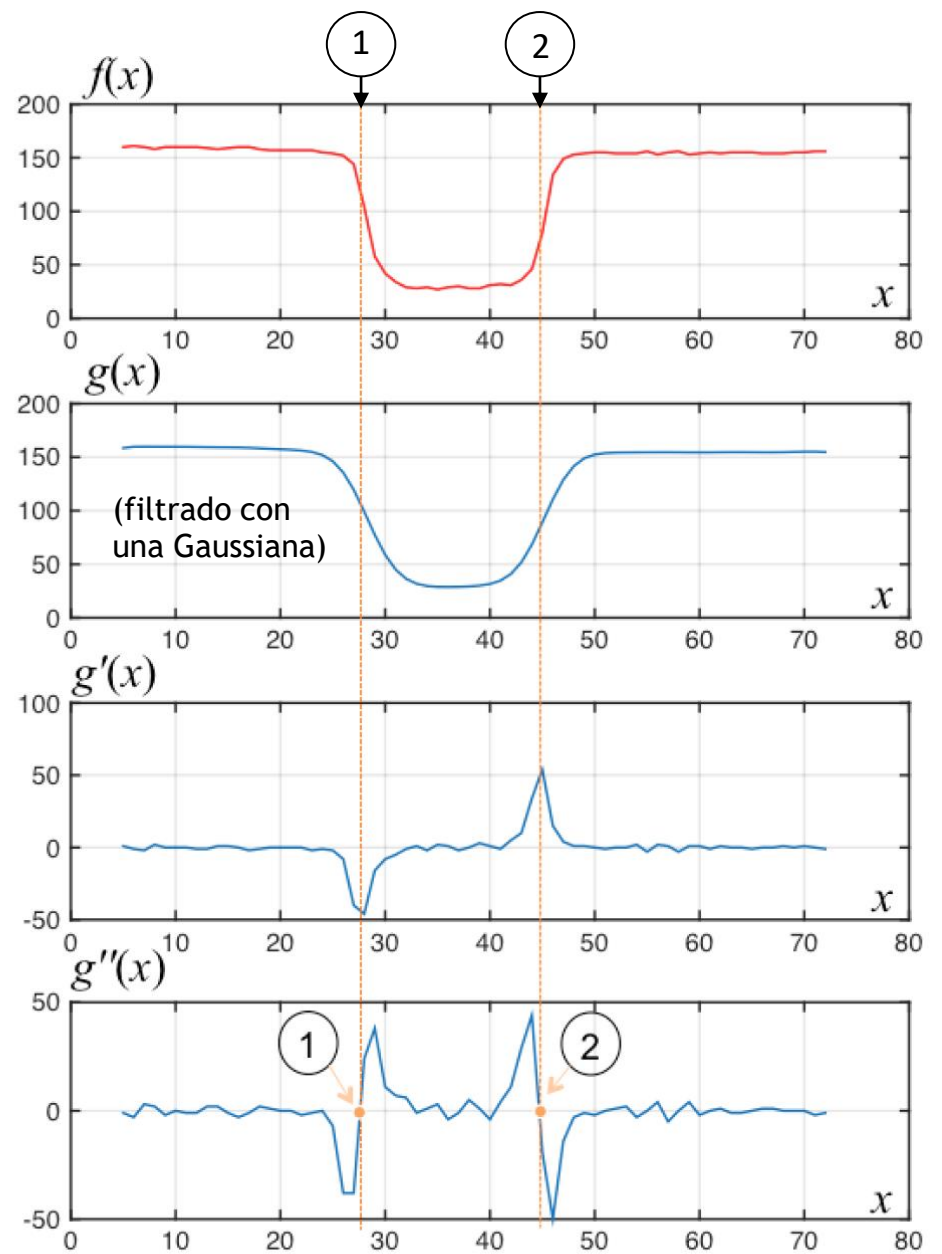
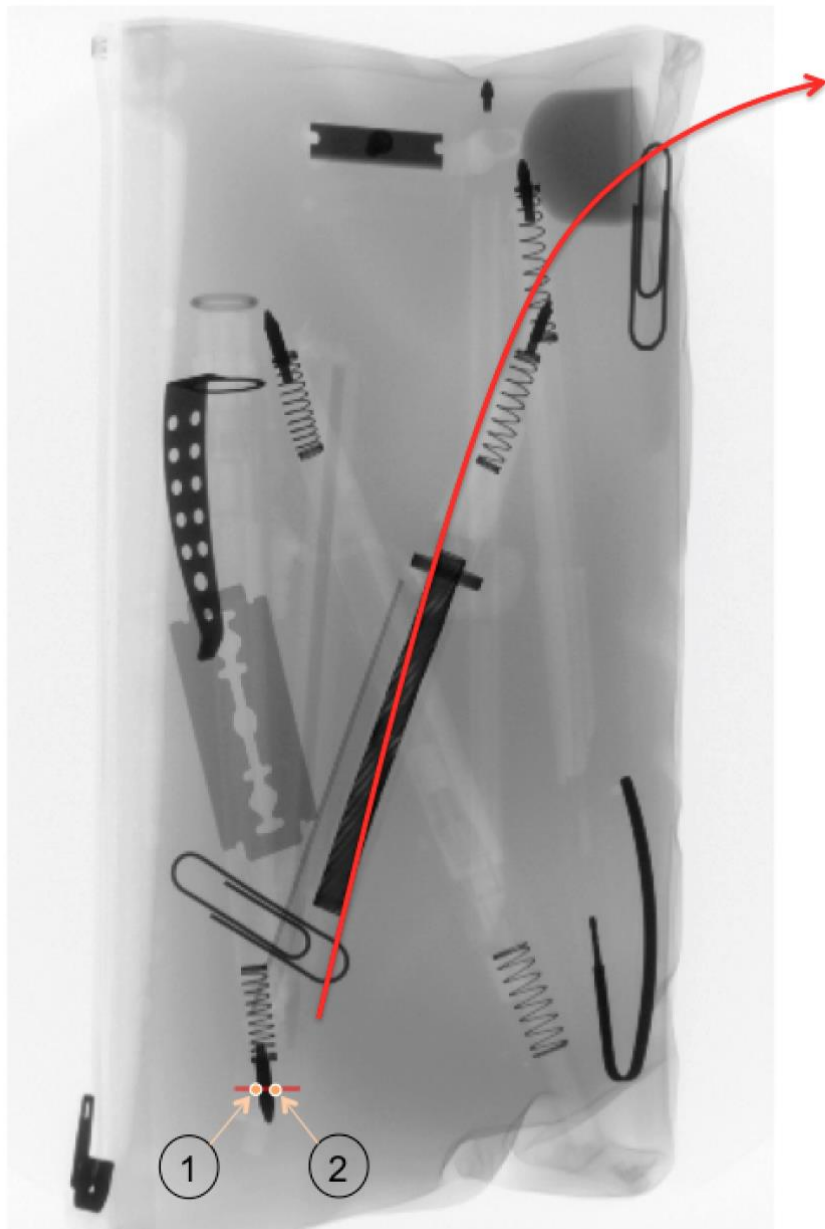
$$\nabla^2 h_{\sigma}(u, v)$$

∇^2 is the **Laplacian** operator:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

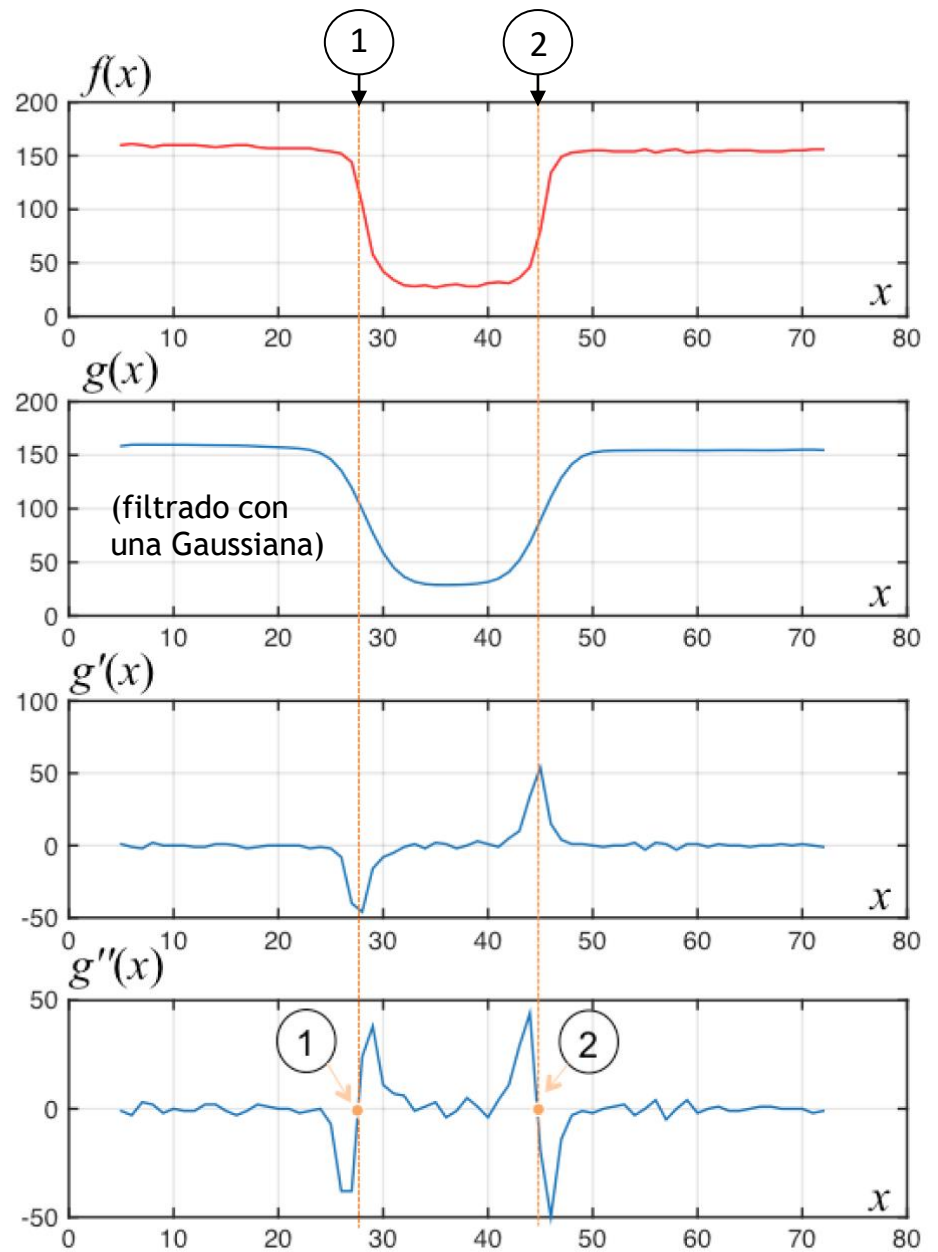






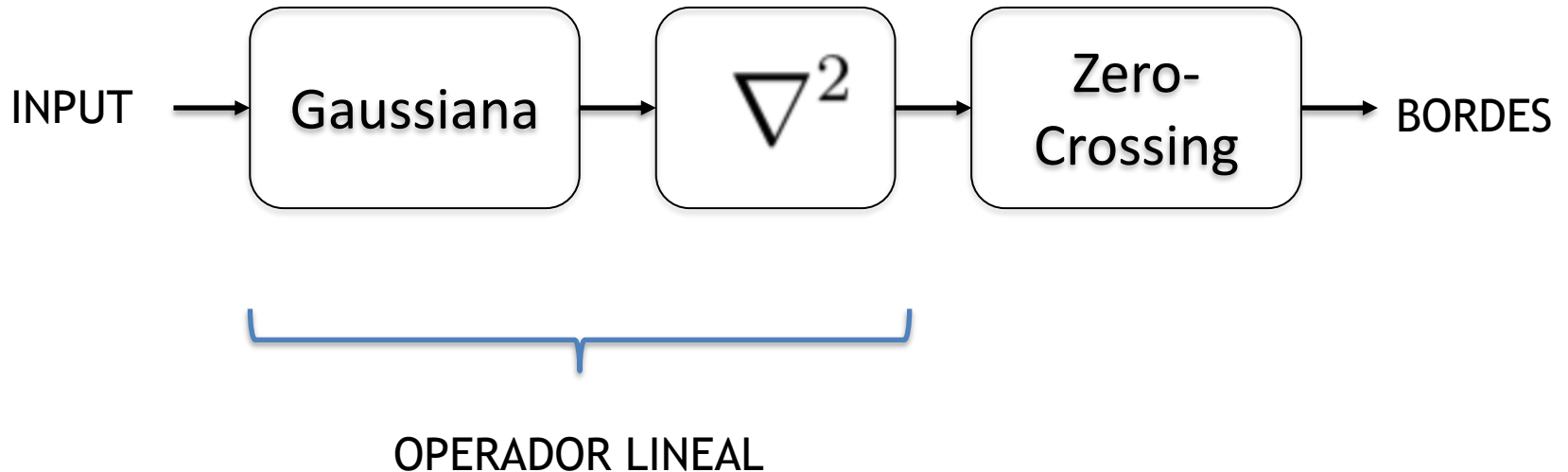


Los bordes, que equivalen a los máximos del gradiente, corresponden a los cruces por cero de la segunda derivada



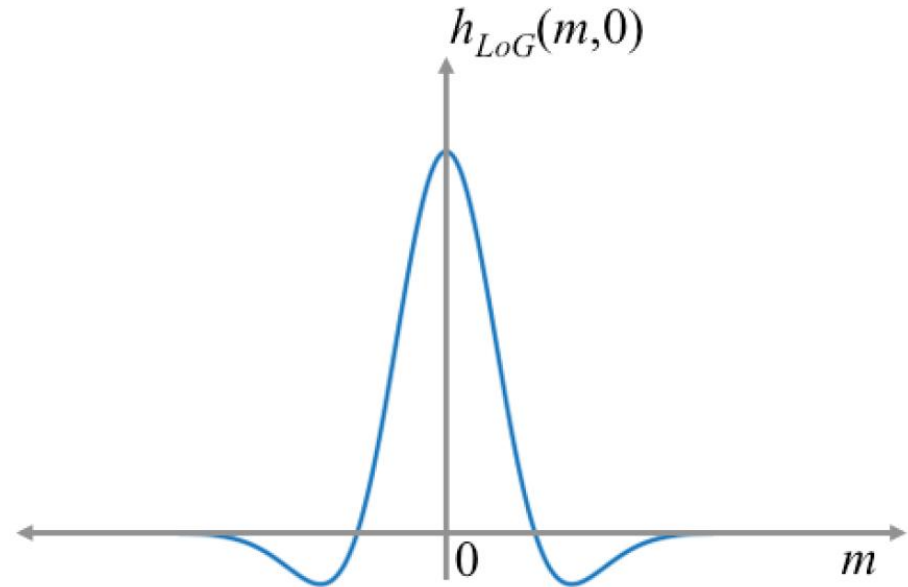
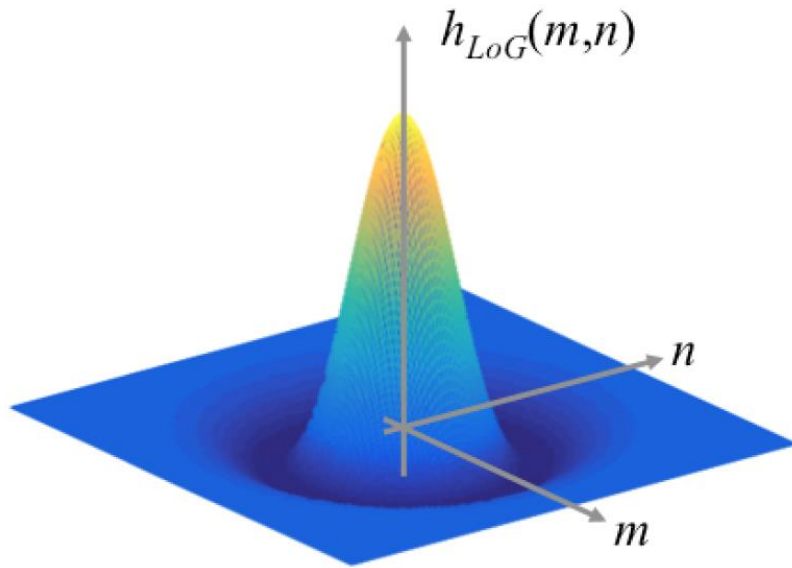
Detector de Bordos usando Laplacian of Gaussian (LoG)

Si un filtro contiene como parte de sus coeficientes números negativos, su operación puede interpretarse como la diferencia de dos diferentes sumas: La suma de todas las combinaciones lineales de los coeficientes positivos del filtro menos la suma de todas las combinaciones lineales debidas a los coeficientes negativos.



Segunda derivada de la Gaussiana

Curiosamente esta función es conocida como «Mexican hat» (o sombrero Mexicano) dada la forma tan parecida de su grafica resultante con el sombrero típico Mexicano:



$$h_{LoG}(m,n) = \frac{1}{2\pi\sigma^4} \cdot \left(2 - \frac{m^2 + n^2}{\sigma^2}\right) \cdot e^{-\frac{m^2 + n^2}{2\sigma^2}}$$

LoG

θ



INPUT



σ



log



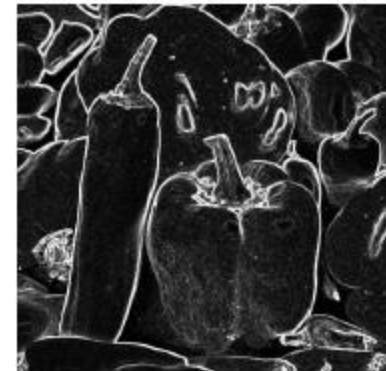
Detector de Bordos usando Canny

Detección de bordes

Algoritmo de Canny

1. Disminución de Ruido (Filtro gaussiano)

$$I' = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix} * I$$



2. Estimación del gradiente (Operador de Sobel)

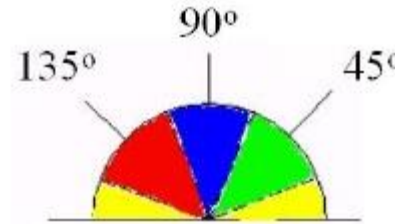
$$G_x = \begin{bmatrix} -1/2 & 0 & 1/2 \end{bmatrix} * \begin{bmatrix} 2 \\ 4 \\ 2 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad G_y = \begin{bmatrix} -1/2 \\ 0 \\ 1/2 \end{bmatrix} * \begin{bmatrix} 2 & 4 & 2 \end{bmatrix} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$$|G| \approx |G_x| + |G_y|$$

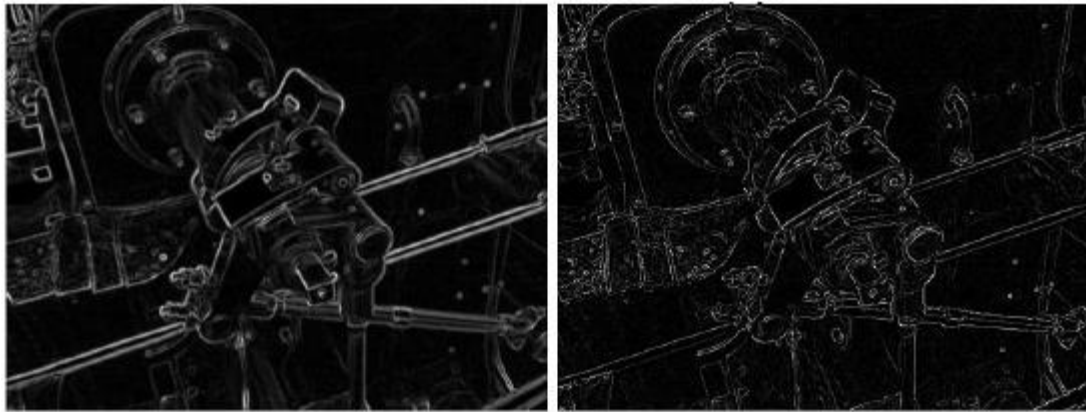
Detección de bordes (Cont.)

3. Estimación de la dirección del gradiente $\angle G = \arctan\left(\frac{G_y}{G_x}\right)$

4. Cuantificación de la dirección



5. Non-maximum suppression (Supresión de los No máximos)

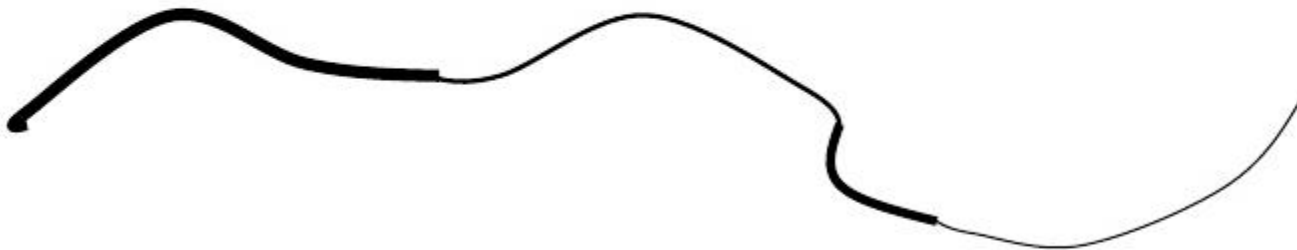


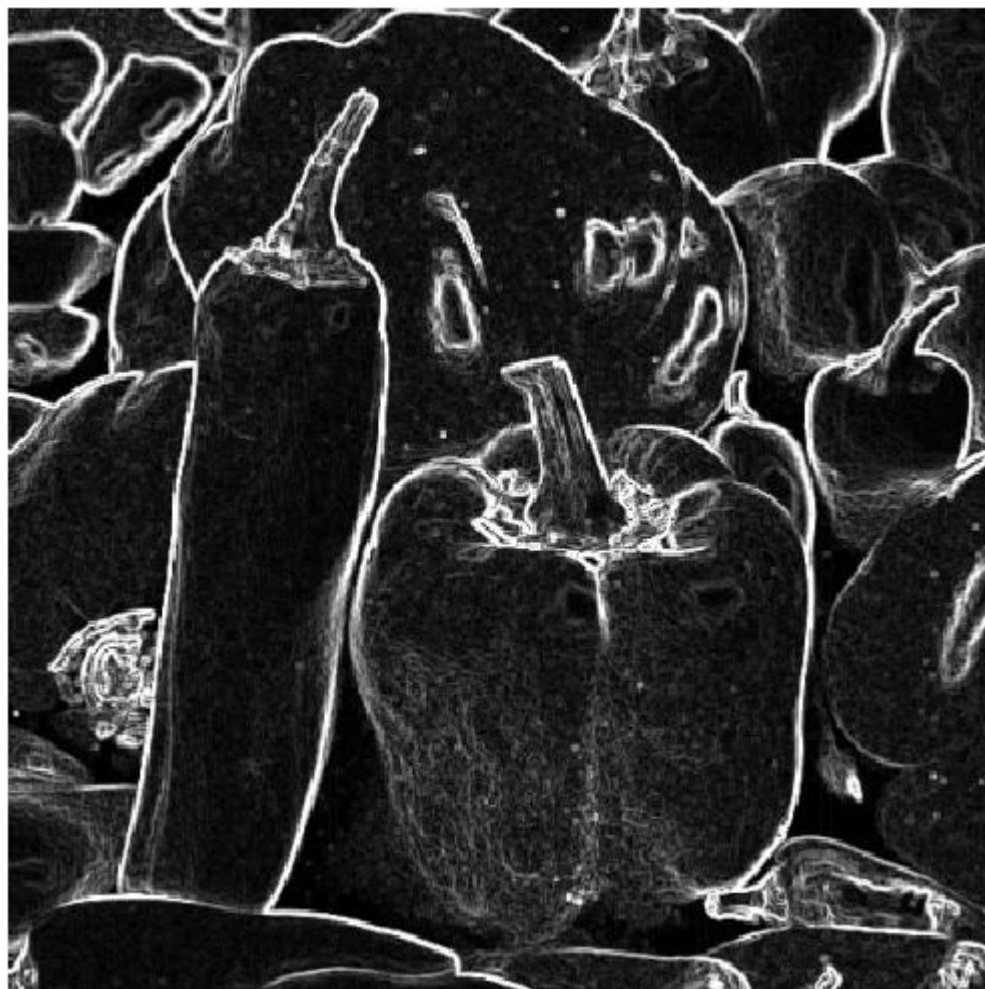
6. Umbralización por histéresis (High and Low)

Detección de bordes (Cont.)

Hysteresis thresholding

- Check that maximum value of gradient value is sufficiently large
 - drop-outs? use **hysteresis**
 - use a high threshold to start edge curves and a low threshold to continue them.





Detector de Canny

Canny propone el uso de una máscara 2D para detectar bordes basándose en un método de optimización que minimiza una función objetivo que incluye los siguientes criterios:

- A) Buena detección: La detección debe corresponder al borde y no al ruido.
- B) Buena Localización: El borde detectado debe ser cercano al real.
- C) Respuesta única: Se debe detectar sólo un borde y no múltiples.

BORDE
VERDADERO



BORDE
VERDADERO



MALA
DETECTION



Criterio A) Buena detección: La detección debe corresponder al borde y no al ruido.

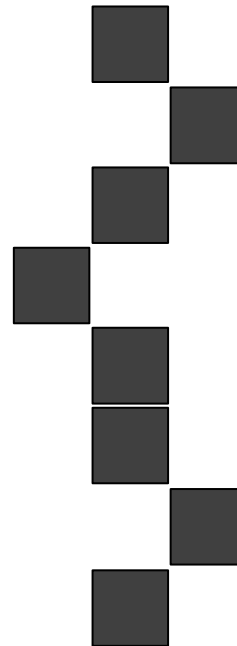
BORDE
VERDADERO



MALA
DETECTION



MALA
LOCALIZACIÓN



Criterio B) Buena Localización: El borde detectado debe ser cercano al real.

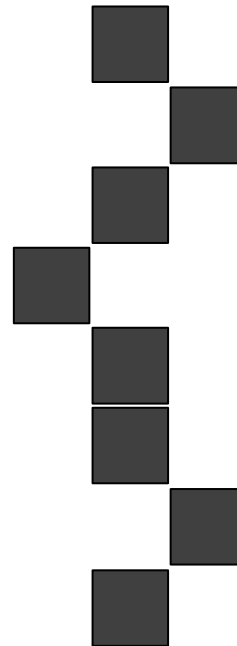
BORDE
VERDADERO



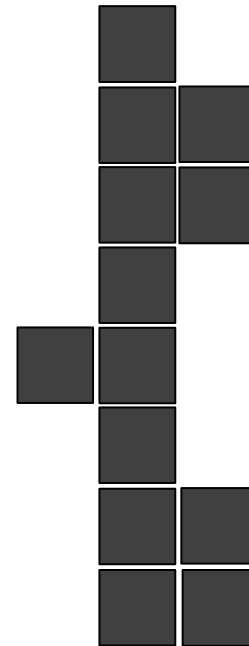
MALA
DETECTION



MALA
LOCALIZACIÓN



RESPUESTA
MULTIPLE



Criterio C) Respuesta única: Se debe detectar sólo un borde y no múltiples.

NO CUMPLE LOS CRITERIOS

BORDE
VERDADERO

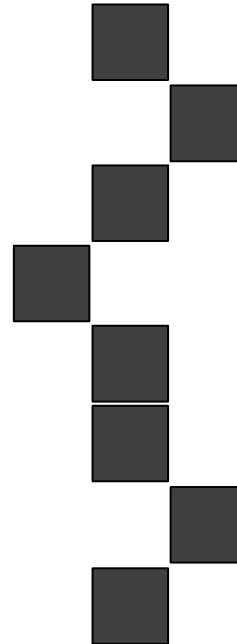


MALA
DETECTION



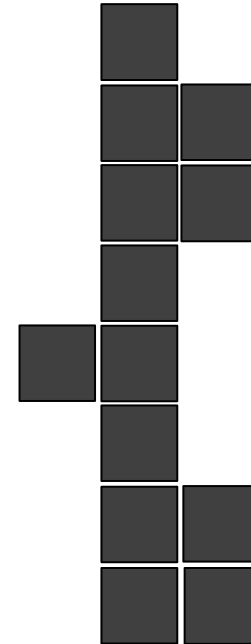
A)

MALA
LOCALIZACIÓN



B)

RESPUESTA
MÚLTIPLE



C)

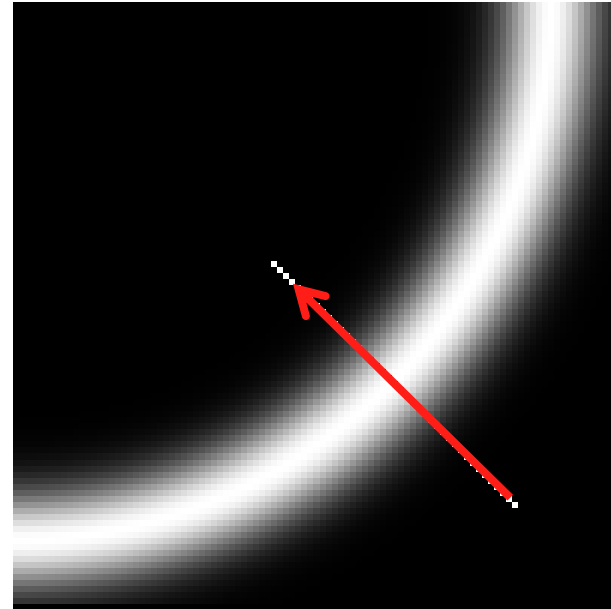
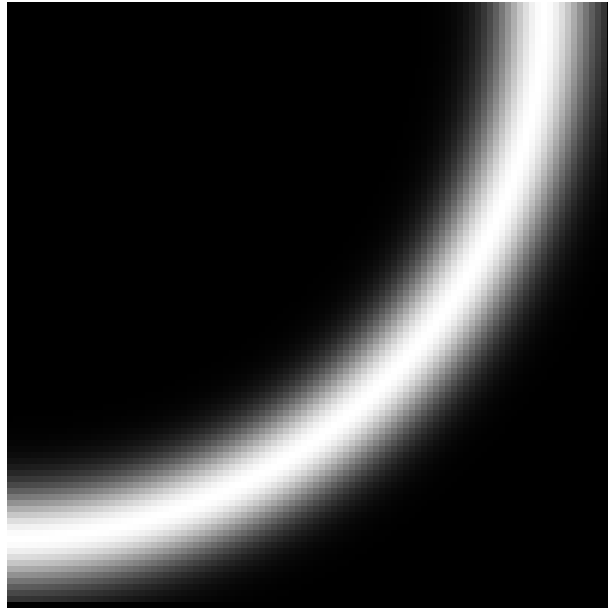
Detector de Canny

La máscara óptima es muy similar a la derivada de una Gaussiana.

La idea es usar esta máscara para encontrar los máximos locales del gradiente de la imagen (NMS).

La implementación práctica utiliza el umbral adaptativo del gradiente (para detectar los bordes fuertes y débiles) con histéresis (los bordes débiles se detectan sólo si están conectados a los bordes fuertes)

Canny Edge Detector - NMS



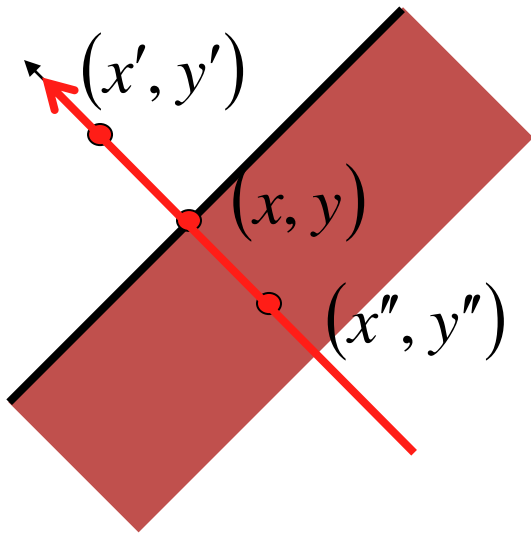
En cada píxel, suprime la intensidad del borde del píxel central (estableciendo su valor en 0) si su magnitud no es mayor que la magnitud de los dos vecinos en la dirección del gradiente.

Non-Maximum Suppression (NMS)

Detector Canny - NMS

(non-maximum suppression)

- Suprime los pixeles del Gradiente que no son máximos locales



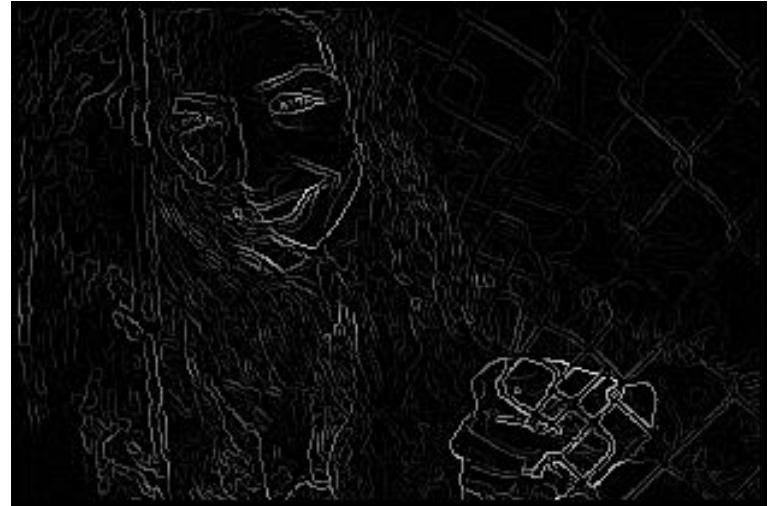
$$M(x, y) = \begin{cases} |\nabla S|(x, y) & \text{if } |\nabla S|(x, y) > |\nabla S|(x', y') \\ & \& |\nabla S|(x, y) > |\nabla S|(x'', y'') \\ 0 & \text{otherwise} \end{cases}$$

Detector Canny - NMS

$$|\nabla S| = \sqrt{S_x^2 + S_y^2}$$



M



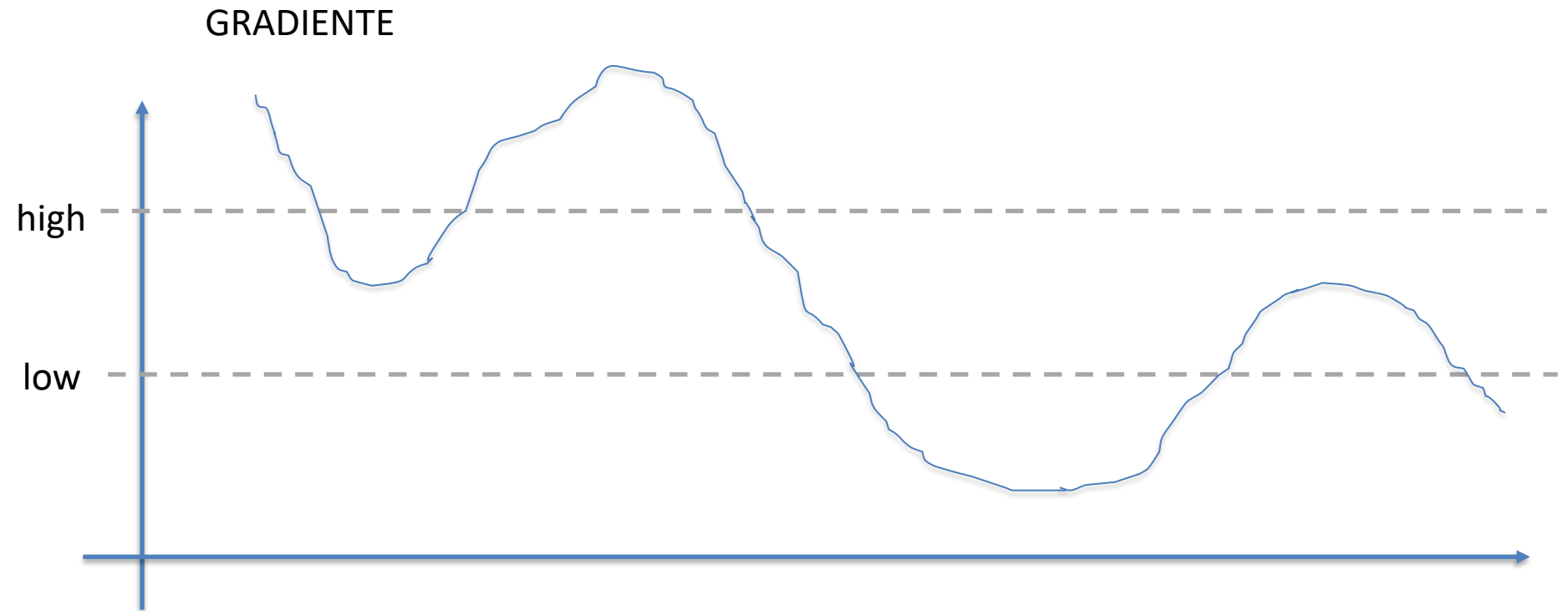
$$|\nabla S| \geq \textit{Threshold} = 25$$



$$M \geq \textit{Threshold} = 25$$



Detector Canny - Histéresis



Hysteresis Thresholding

Detector Canny - Histéresis

El umbral por histéresis se centra en establecer dos umbrales, uno máximo y otro mínimo.

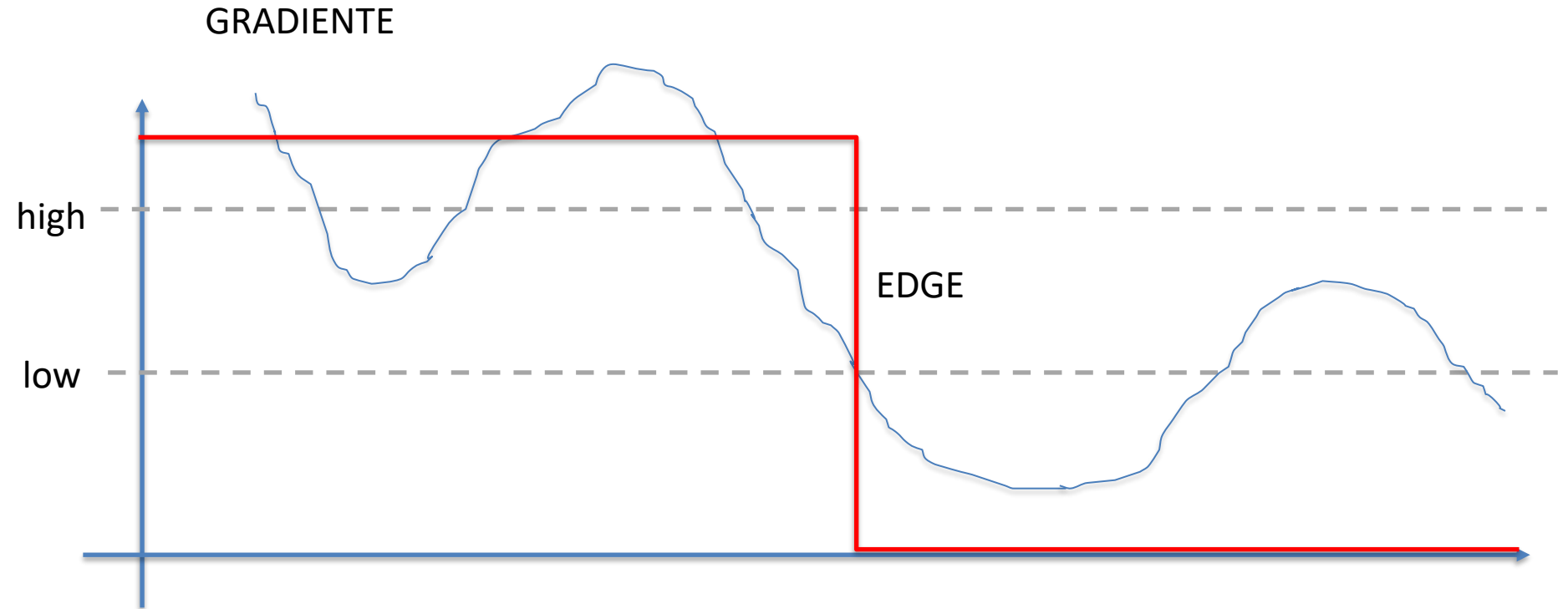
Esto te ayudará a determinar si un píxel forma parte de un borde o no. Pueden darse 3 casos:

Si el gradiente de un píxel está por encima de “Máximo”, declararlo un "píxel de borde”

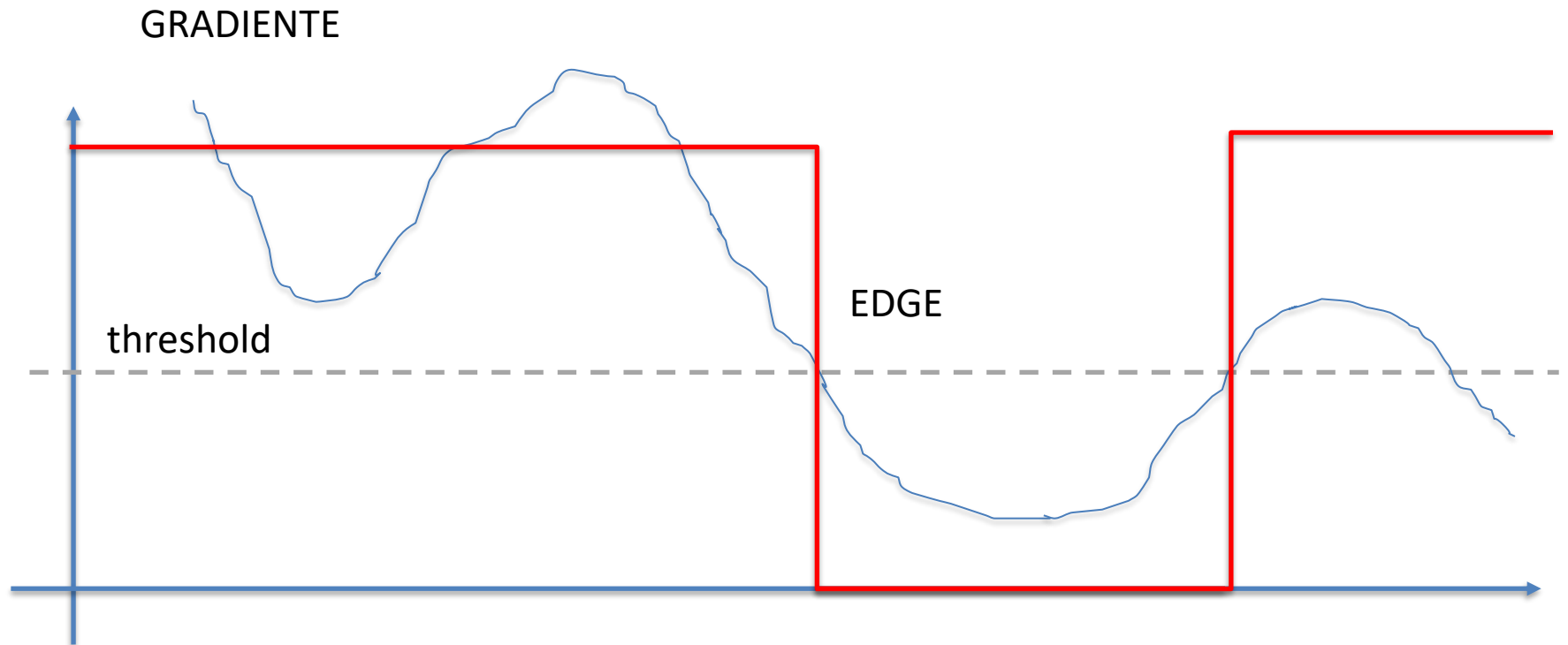
Si el gradiente de un píxel está por debajo de “Mínimo”, declararlo como un "no-píxel de borde".

Si el gradiente de un píxel está entre “Mínimo” y “Máximo”, entonces declararlo como un "píxel de borde" si y sólo si está conectado a un "píxel de borde" directamente o a través de píxeles entre “Mínimo” y “Máximo”.

Detector Canny - Histéresis



Detector Canny - Histéresis



SIN HISTÉRESIS

Canny

INPUT



θ



σ



canny



LoG

θ



INPUT



σ



log



Lecturas Sugeridas

- Chapter 8, David A. Forsyth and Jean Ponce, "Computer Vision: A Modern Approach"
- Chapter 4, Emanuele Trucco, Alessandro Verri, "Introductory Techniques for 3-D Computer Vision"
- Canny, John (1986): "A computational approach to edge detection." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6 (1986): 679-698.

