



DEPARTAMENTO DE  
**INGENIERÍA  
INFORMÁTICA**  
UNIVERSIDAD DE SANTIAGO DE CHILE

# Procesamiento y Análisis de Imágenes

**Violeta Chang**

[violeta.chang@usach.cl](mailto:violeta.chang@usach.cl)

Créditos por slides: José M. Saavedra

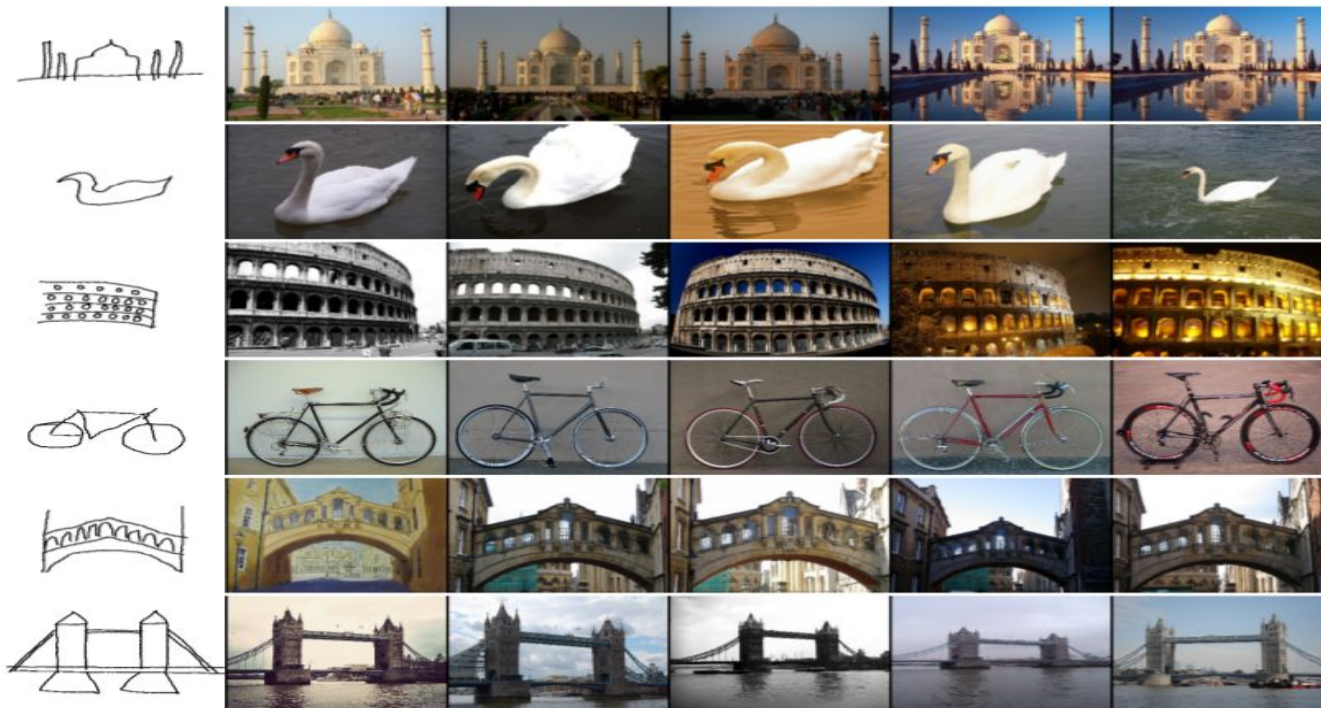
# Detección de Bordes

- **Objetivo:** Identificar cambios repentinos o discontinuidades en una imagen.
  - Los bordes llevan alto valor semántico
  - Permiten representar la forma de los objetos de una imagen
  - El cerebro responde con mayor fuerza a los bordes.
  - Más compacto que los píxeles.
- **Entrada:** Imagen en escala de grises de tamaño  $M \times N$
- **Salida:** Imagen en escala de grises de tamaño  $M \times N$



Fuente: Lowe

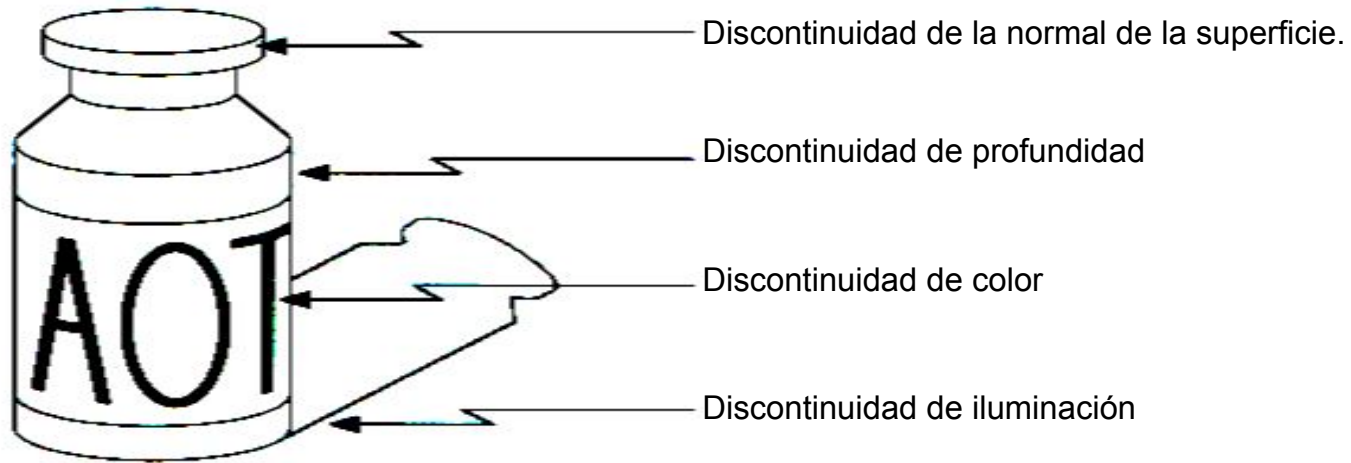
# Detección de Bordes



Recuperación de Imágenes basada en contornos

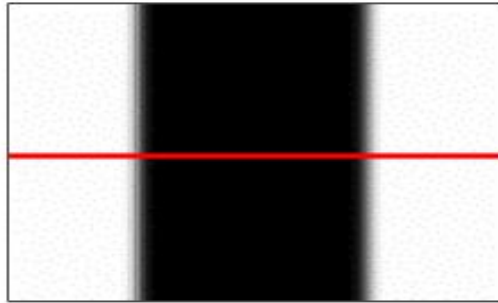
# Detección de Bordes

- Los bordes se generan por una variedad de factores:

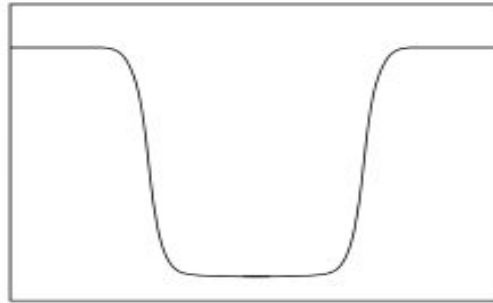


# Detección de Bordes

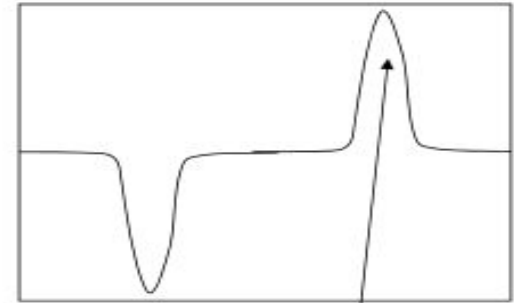
- Caracterización de un borde
  - Un borde se detecta en el lugar de rápido cambio de intensidad en la imagen → **derivada** o diferencial en



imagen



Función de intensidad  
a lo largo de la línea de barrido



Puntos del borde corresponden  
a extremos

# Detección de Bordes

- Gradiente de una imagen

$$\nabla I = [g_x, g_y]^T = \left[ \frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right]^T$$

- Magnitud y ángulo del gradiente

$$M(x, y) = \text{mag}(\nabla I) = \sqrt{g_x^2 + g_y^2}. \quad \alpha(x, y) = \tan^{-1} \left[ \frac{g_y}{g_x} \right]$$

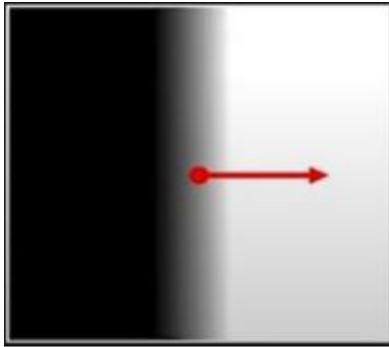
- Aproximación de la magnitud (menos cómputo)

$$|g_x| + |g_y|$$

$$\max(|g_x|, |g_y|)$$

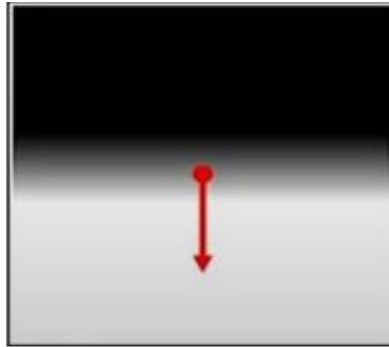
# Detección de Bordes

- Gradiente de una imagen



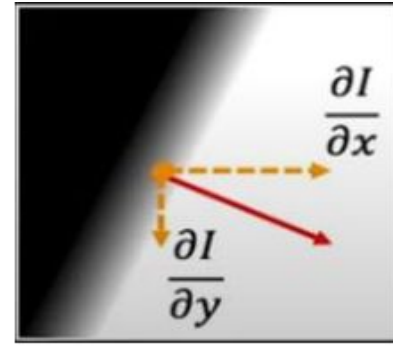
$$\nabla I = \frac{dI}{dx}, 0$$

a



$$\nabla I = 0, \frac{dI}{dy}$$

b



$$\nabla I = \frac{dI}{dx}, \frac{dI}{dy}$$

c

Dirección de cambio es perpendicular al borde

## Detección de Bordos

- Sea  $f(x,y)$  una función de dos variable, la derivada parcial es:

$$\frac{\partial f(x, y)}{\partial x} = \lim_{\varepsilon \rightarrow 0} \frac{f(x + \varepsilon, y) - f(x, y)}{\varepsilon}$$

- Podemos discretizarla como:

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

Entonces es posible aplicar filtros de correlación o convolución



# Detección de Bordes



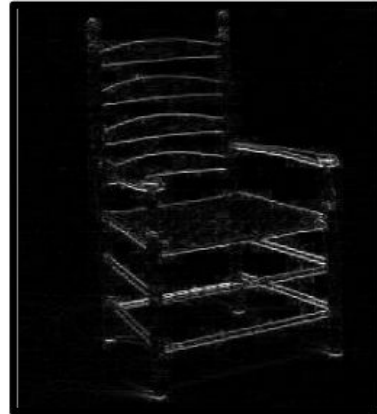
$$\frac{\partial f(x, y)}{\partial x}$$

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



$$\frac{\partial f(x, y)}{\partial y}$$

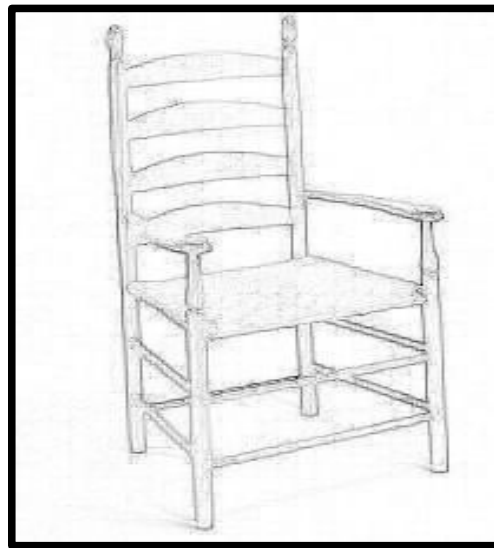
0
-1
1



# Detección de Bordes



*negativo del resultado*



$|gx| + |gy|$

# **Detección de Bordes**

## **Operadores de Primer Orden**

## Operador de Roberts

- Vecindad de 2x2

$$G_x = \begin{array}{|c|c|} \hline 1 & 0 \\ \hline 0 & -1 \\ \hline \end{array}$$

$$G_y = \begin{array}{|c|c|} \hline 0 & 1 \\ \hline -1 & 0 \\ \hline \end{array}$$

$$G(f(i, j)) = |f(i, j) - f(i+1, j+1)| + |f(i, j+1) - f(i+1, j)|$$

- $G_x$  es aproximación de  $\left(\frac{\partial f}{\partial x}\right)$
- $G_y$  es aproximación de  $\left(\frac{\partial f}{\partial y}\right)$

# Operador de Roberts

- Desventajas:
  - Pocos pixeles en la vecindad  $\rightarrow$  muy sensible al ruido
  - Máscara de  $2 \times 2 \rightarrow$  aproximación de  $(i,j)$  en  $(i+\frac{1}{2}, j+\frac{1}{2})$

## Operador de Roberts



Imagen Original

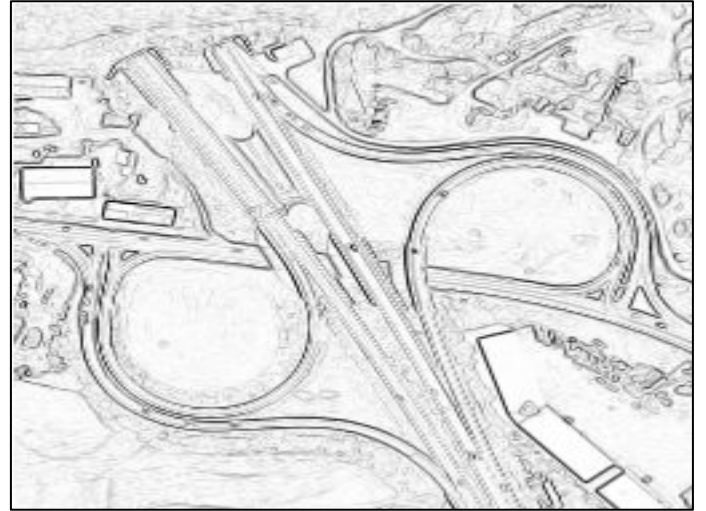
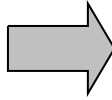


Imagen Procesada

# Operador de Prewitt

- Filtro para estimar derivadas

mask_y	-1	-1	-1
	0	0	0
	1	1	1

mask_x	-1	0	1
	-1	0	1
	-1	0	1

## Operador de Prewitt



Imagen Original

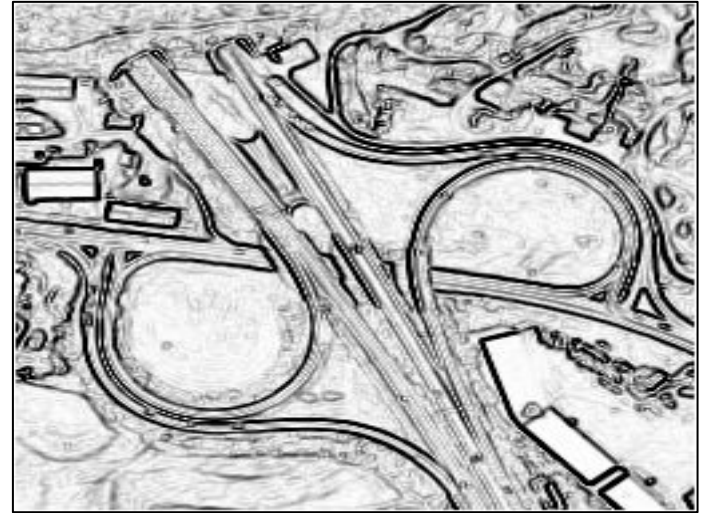
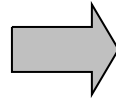


Imagen Procesada



# Operador de Sobel

- Filtro para estimar derivadas
  - SOBEL: Prewitt + filtro gaussiano

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

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

## Operador de Sobel



Imagen Original

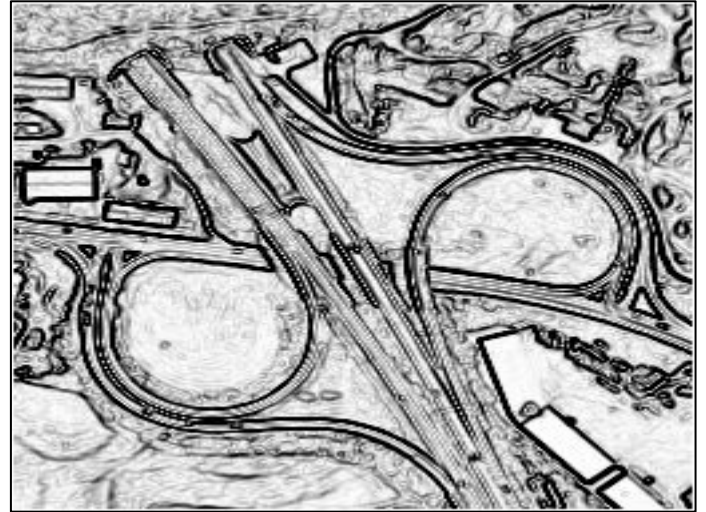
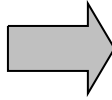
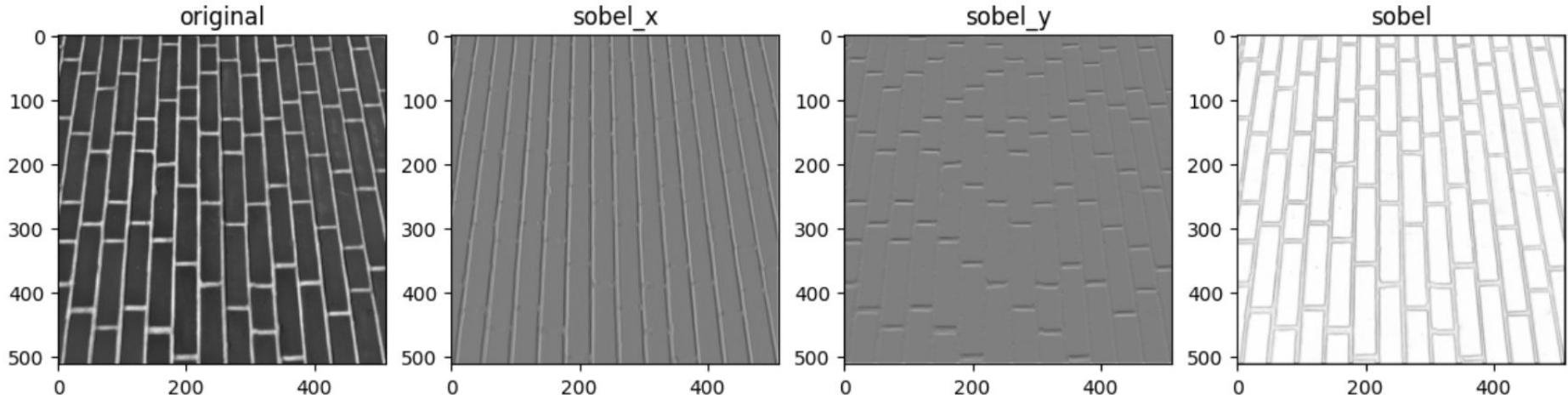


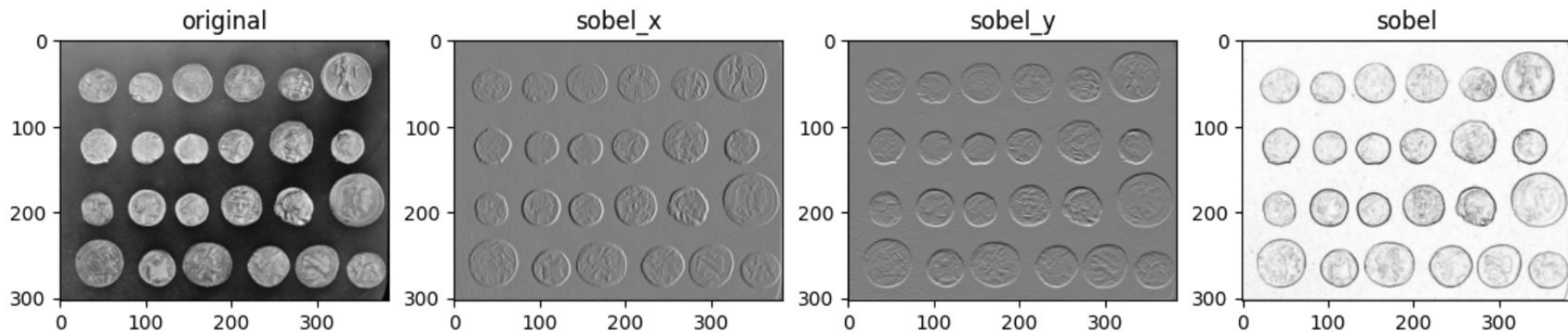
Imagen Procesada

# Operador de Sobel

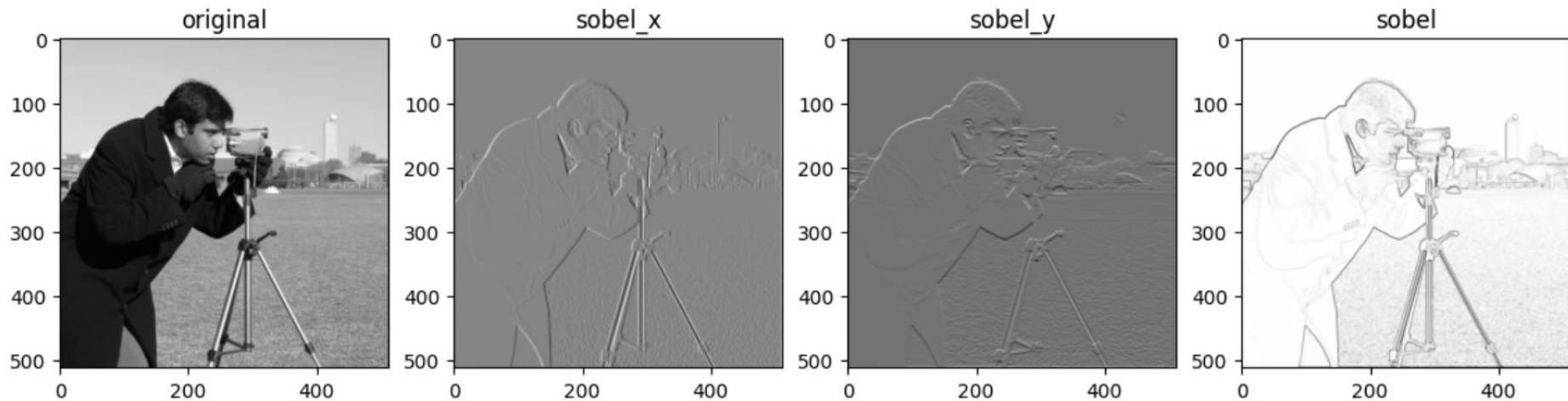


(La última columna muestra el negativo del resultado)

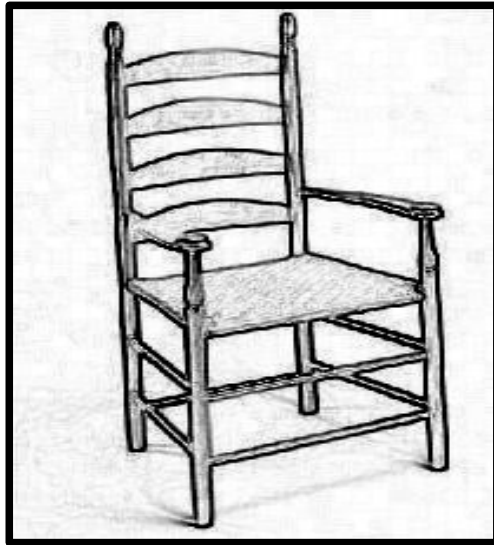
# Operador de Sobel



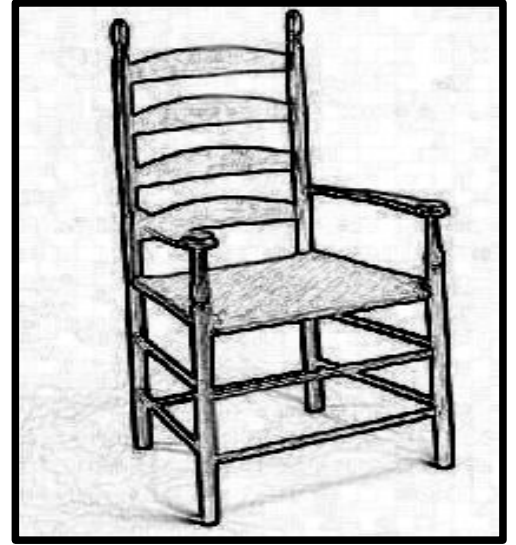
# Operador de Sobel



## Prewitt VS Sobel



PREWITT



SOBEL

# Optimizando la Detección de Bordes

# Método de Canny

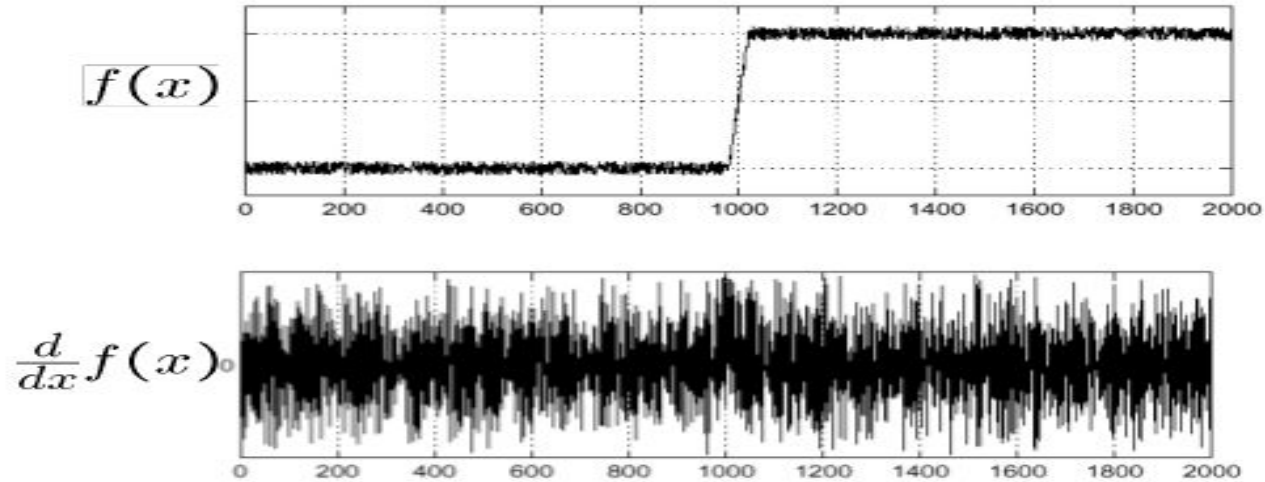
Canny, J., A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986

- Objetivos
  - **Detección óptima**: No respuestas espurias.
  - **Buena localización**: Mínima distancia al borde real.
  - **Respuesta única**: Eliminar múltiples respuestas en un solo punto de borde.



# Método de Canny

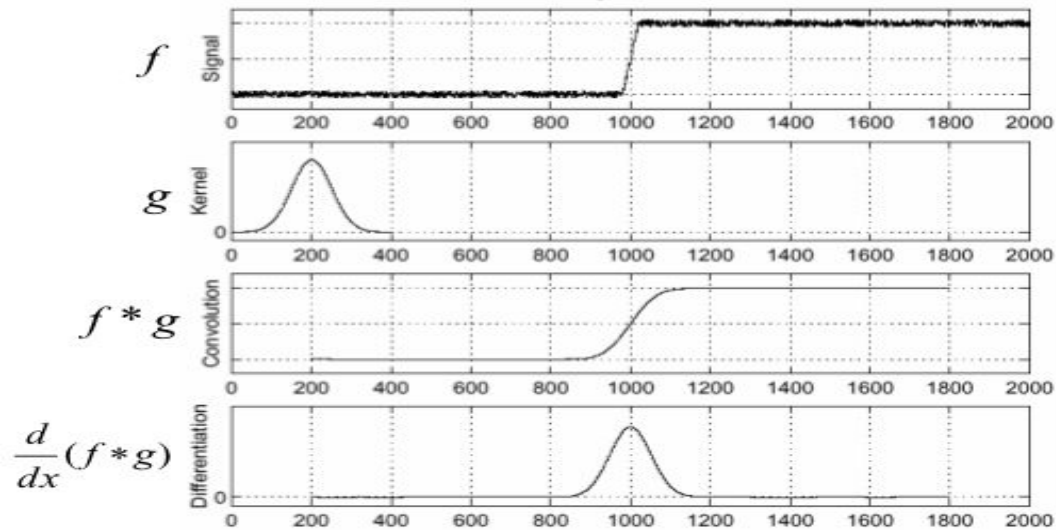
Detección óptima: Reducir respuesta al ruido.



¿Dónde está el borde?

# Método de Canny

- **Solución:** Suavizar primero (filtro gaussiano)



## Método de Canny

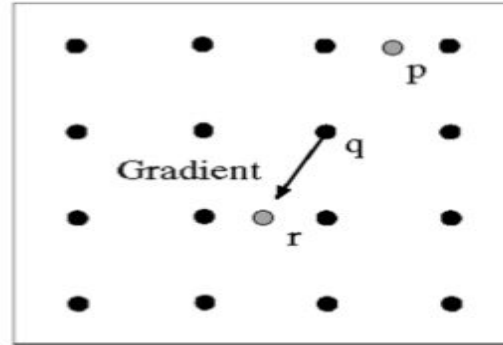
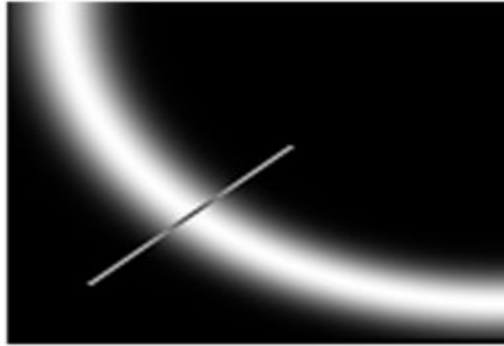
- Determinar potenciales bordes usando Sobel.
- Buena localización



Borde conectado a un  
borde fuerte

## Método de Canny

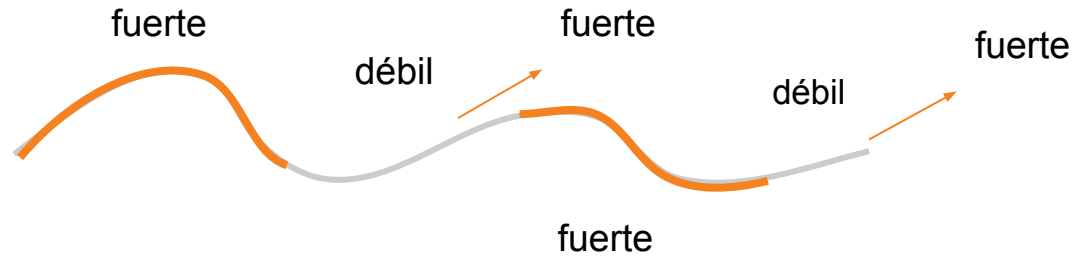
- **Respuesta única:** Aplicar supresión de no-máximos en la dirección del gradiente. Un punto se descarta como borde si tiene una respuesta Sobel menor a alguno de sus vecinos en la dirección del gradiente.



## Método de Canny

- **Solución:** Mediante **supresión de no-máximos** e **histéresis** → Tres tipos de bordes a través de 2 umbrales:
  - *Fuerte*: si  $\text{magnitud} > \text{TH\_HIGH}$
  - *Débil*: si  $\text{magnitud}$  entre  $\text{TH\_LOW}$  y  $\text{TH\_HIGH}$
  - *No\_borde*: Si  $\text{magnitud} < \text{TH\_LOW}$
- Un borde débil se convierte en fuerte si es conexo a un borde fuerte.
- Los débiles alcanzados por los fuertes son fuertes.
- Finalmente, el borde está definido por puntos fuertes.

# Método de Canny

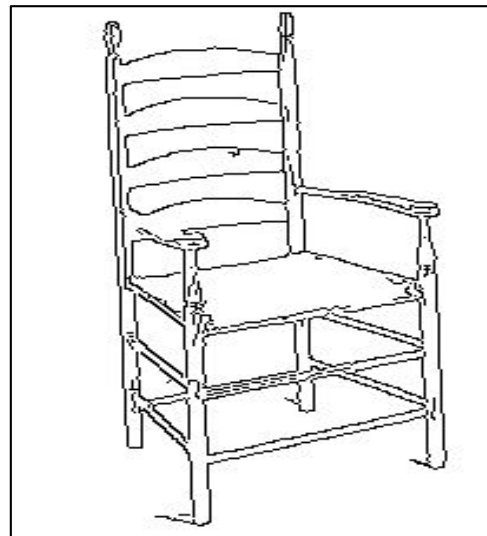


# Método de Canny

- Resumen:
  1. Filtrar la imagen usando filtro Gaussiano.
  2. Encontrar magnitud y dirección del gradiente (Sobel).
  3. Aplicar supresión de no-máximos.
  4. Aplicar histéresis.

# Método de Canny

- 

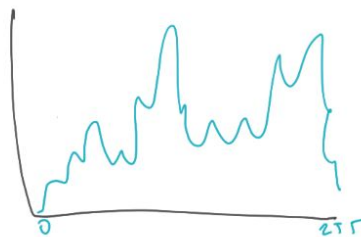




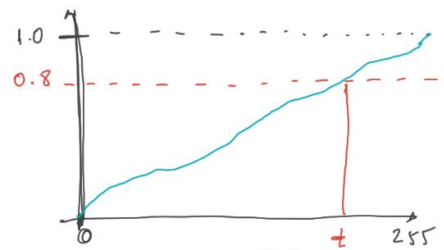
# Método de Canny

- TH\_HIGH=?
- TH\_LOW=?
- **Heurística**

- Suponer que el 20% de pixeles en una imagen corresponden a puntos de borde fuerte.
- TH\_HIGH se elige de modo que logre descartar 80%.
- $TH\_LOW = 0.4 * TH\_HIGH$



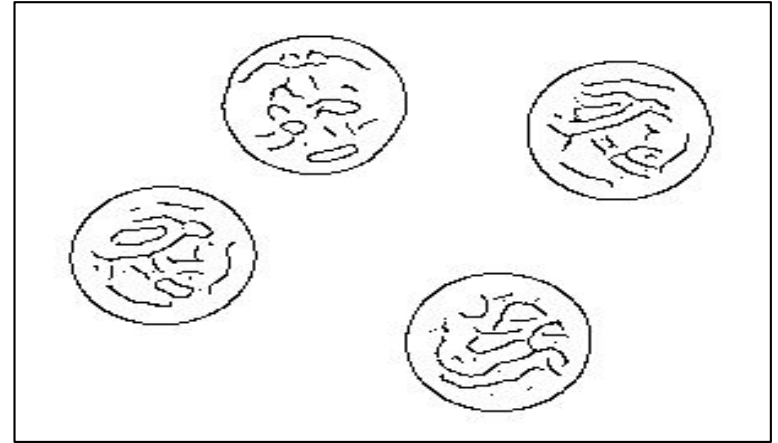
histograma de  
magnitudes



magnitudes  
accum

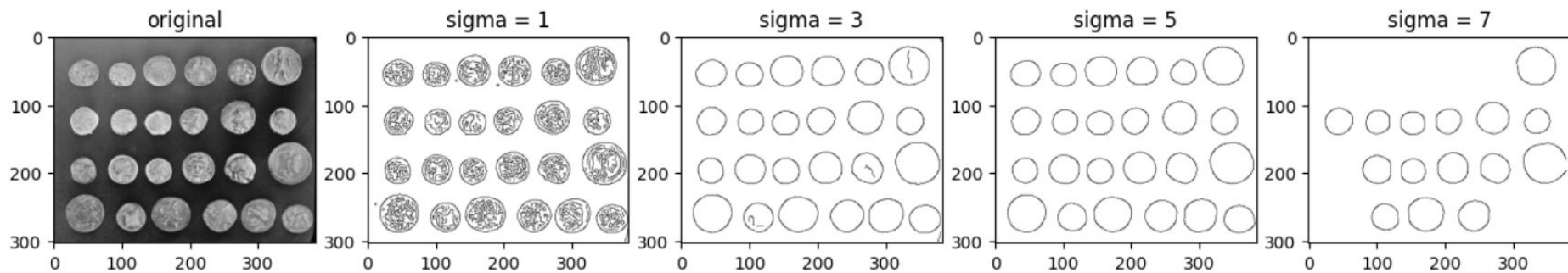
*Usar histograma de magnitudes, por ejemplo cuantizar en 256 valores*

# Método de Canny



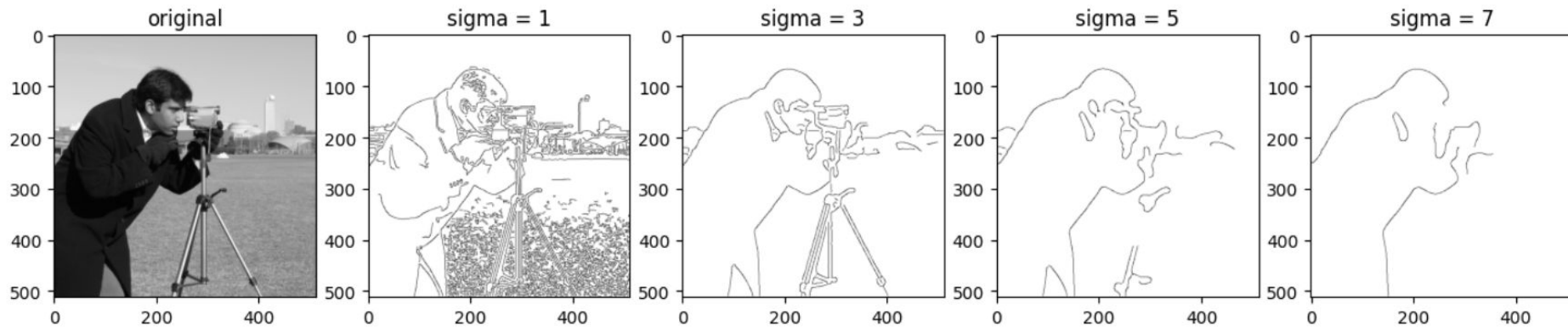
Detección de bordes con Canny

# Método de Canny

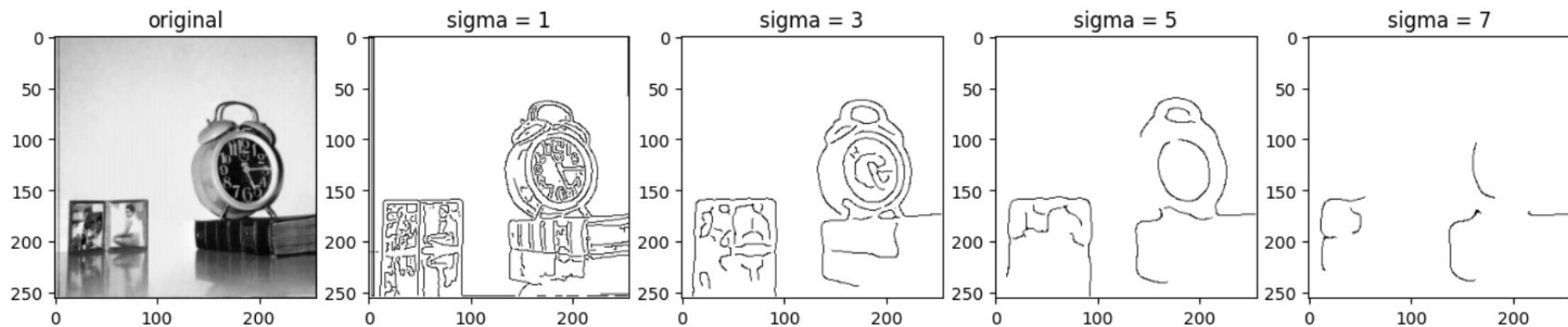


Resultado del método de Canny para varios valores de sigma  
(desviación estándar del filtro gaussiano)

# Método de Canny

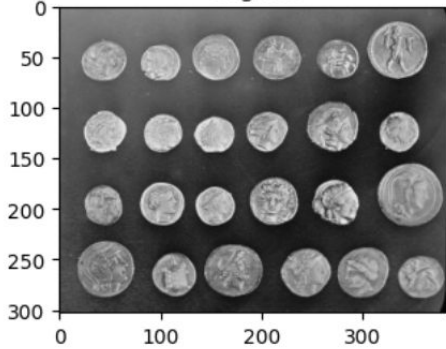


# Método de Canny

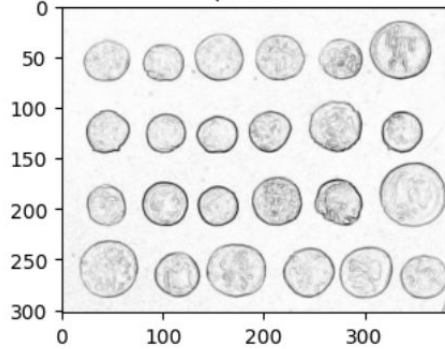


# Comparación entre métodos

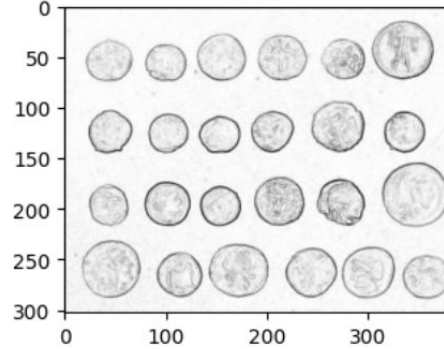
original



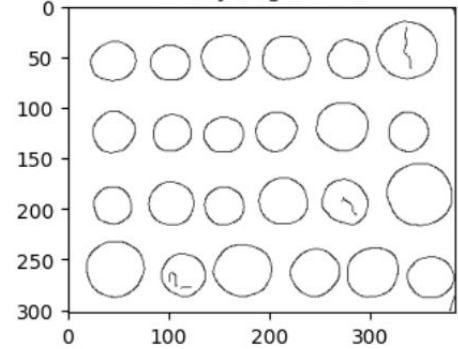
prewitt



sobel

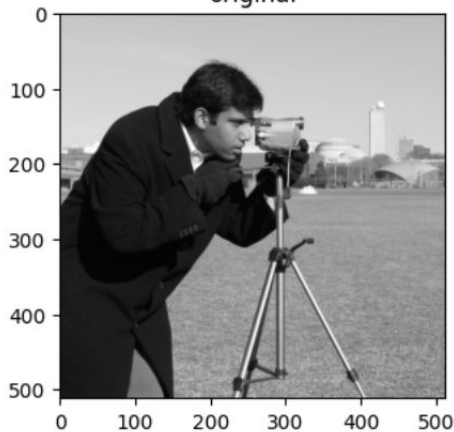


canny (sigma=3)

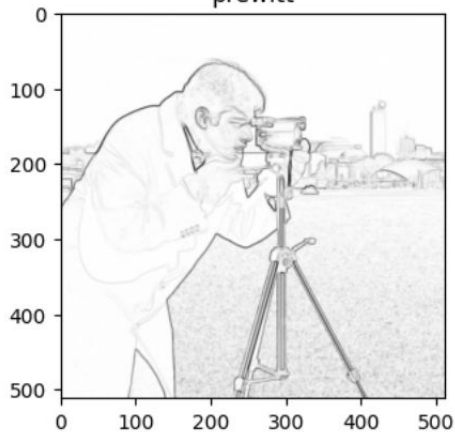


# Comparación entre métodos

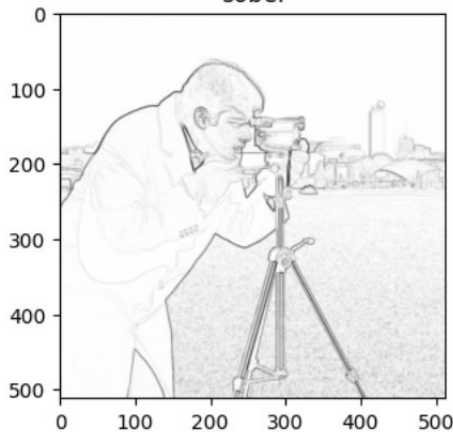
original



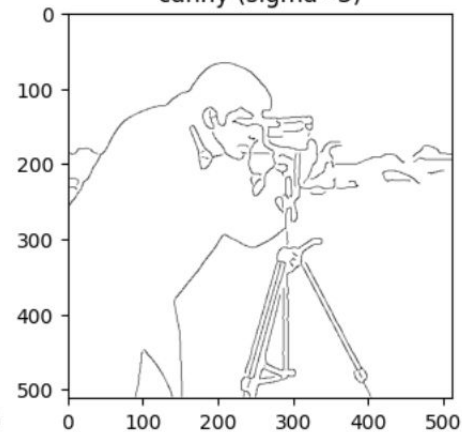
prewitt



sobel

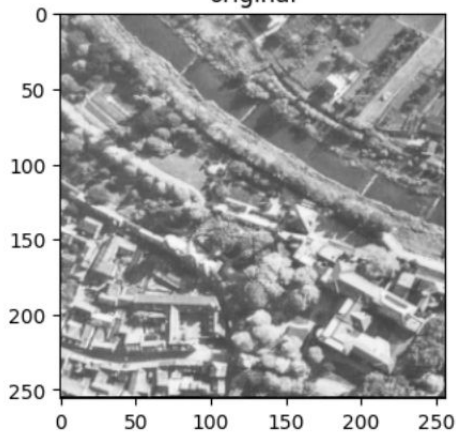


canny (sigma=3)

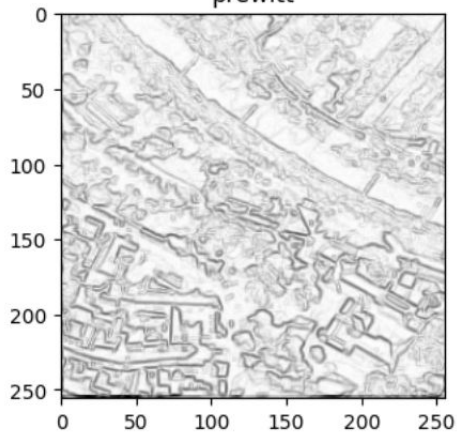


# Comparación entre métodos

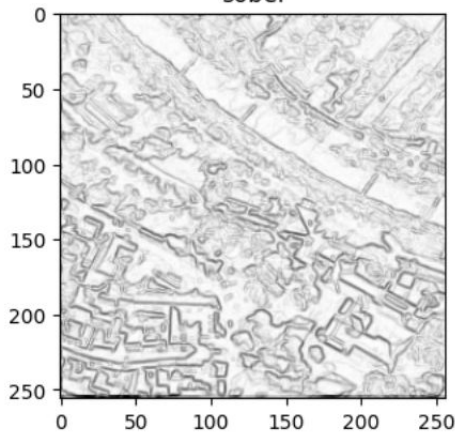
original



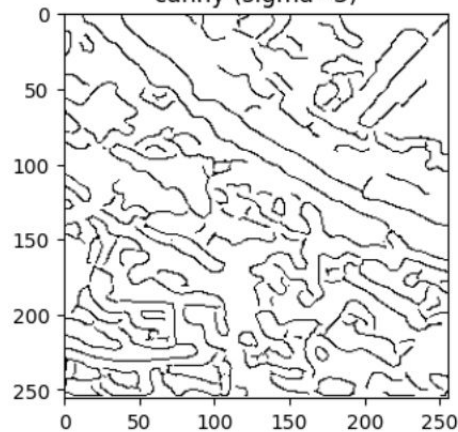
prewitt



sobel



canny (sigma=3)



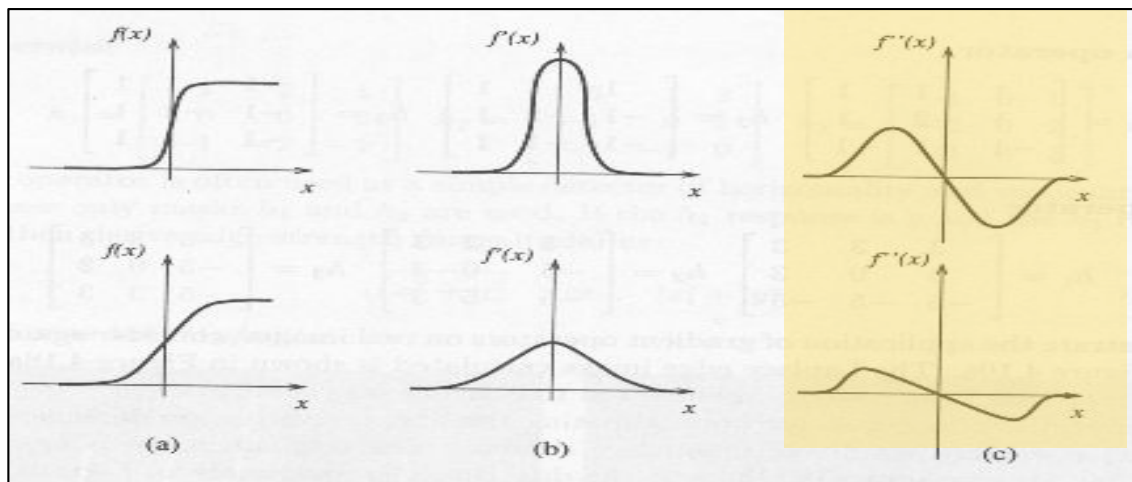


# Detección de Bordes

## Operadores de Segundo Orden

# Detección de Bordes

- Operadores de Segundo Orden
  - Requiere detectar zero-crossing.



## Operador Laplaciano

- Laplaciano de una función  $f(x,y)$  es:

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

- Segunda derivada se aproxima como:

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

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

# Operador Laplaciano

- Operador laplaciano:

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)$$

- Aproximaciones:

0	1	0	1	1	1
1	-4	1	1	-8	1
0	1	0	1	1	1

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

# Operador Laplaciano



Imagen Original

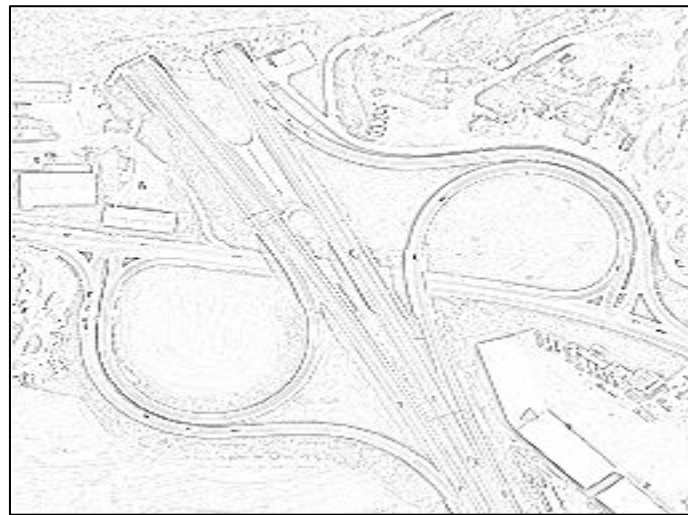
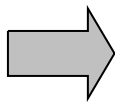


Imagen Procesada

## Laplaciano de Gaussiano

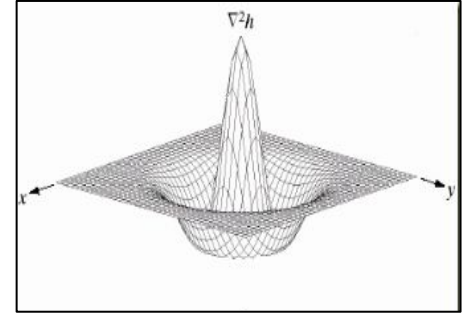
- El operador laplaciano es susceptible al ruido, por lo que es usado junto con un filtro gaussiano: suavizar según filtro gaussiano y luego detectar bordes usando un operador laplaciano
- LOG : Laplaciano del Gaussiano

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

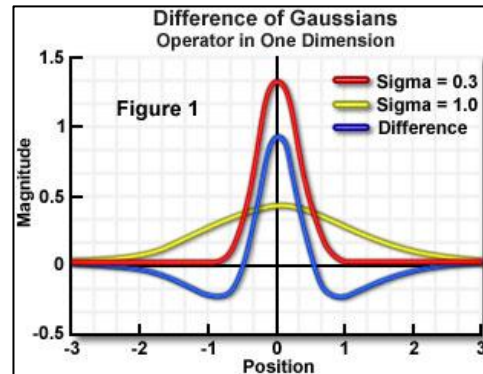
OBS. : G es una distribución gaussiana en 2D

# Laplaciano de Gaussiano

$$\nabla^2 g(x, y) = \frac{1}{2} \left( \frac{x^2 + y^2}{\sigma^2} - 2 \right) e^{-\frac{x^2 + y^2}{2\sigma^2}}$$



Aproximación por  
diferencia de  
Gaussianas (DoG)



# Laplaciano de Gaussiano



Imagen Original

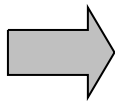
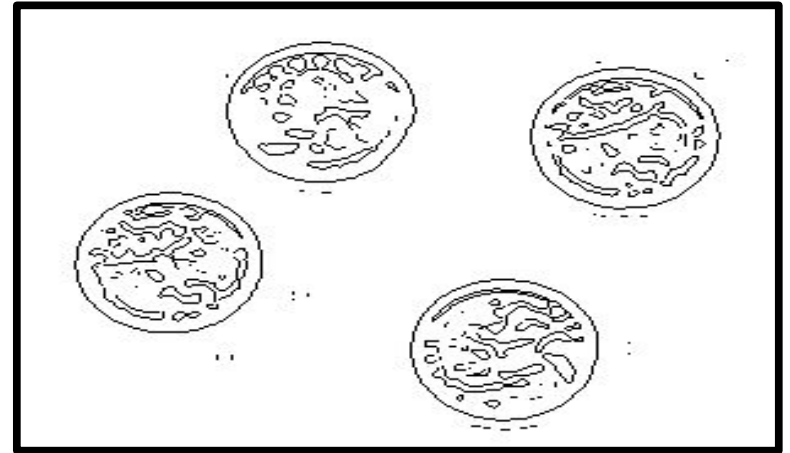


Imagen Procesada



# Laplaciano de Gaussiano

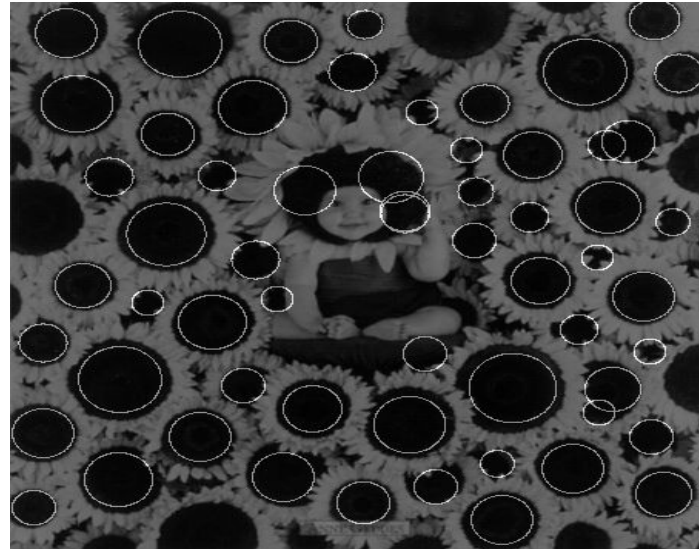
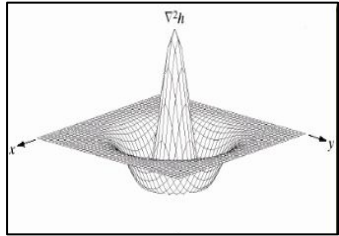


*Detección de Bordes con LoG*

# Laplaciano de Gaussiano

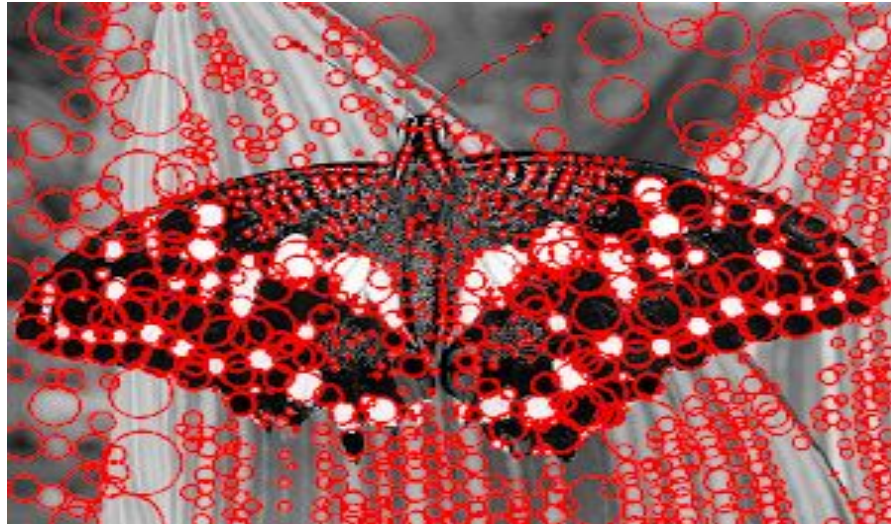
- LoG/DoG también se utiliza como detector de regiones de interés a partir de las cuales se extrae información para un análisis posterior (Blobs)

Detección de zonas en las que hay alto contraste entre la zona interna y externa



## Laplaciano de Gaussiano

- LoG como detector de Blobs (regiones de interés) → métodos como SIFT (Scale Invariant Feature Transform) utilizan este tipo de información

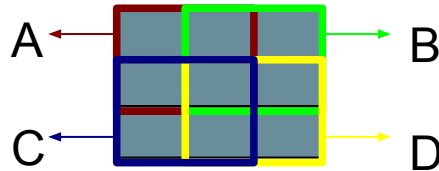


# Detección de Bordes

- **Operadores de Segundo Orden**

- Aplicación de Zero-Crossing

- Buscar ocurrencias de cruces por cero luego de filtrar la imagen con un operador de segundo orden.
    - **Estrategia:** Procesar la imagen filtrada en bloques de 3x3. Cada bloque se divide en cuadrantes, cada uno incluyendo al centro. *Calcular el promedio de los cuadrantes (A,B,C,D) y encontrar el mínimo y máximo de esos promedios. Si el mínimo es negativo y el máximo positivo, entonces se tiene un cruce por cero.*



## Operadores de Kirsch

- Objetivo: Obtener no sólo la magnitud de las orillas sino también su dirección
- Para detectar siluetas (contornos)
- Operadores de Prewitt:  $0^\circ$  y  $90^\circ$
- Operadores de Kirsch:  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  y  $135^\circ$

# Operadores de Kirsch

0 grados

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

45 grados

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

90 grados

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

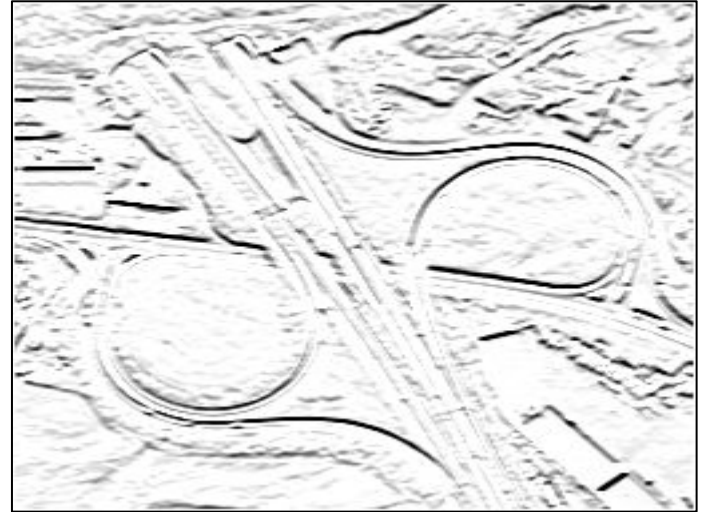
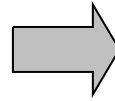
135 grados

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

# Operadores de Kirsch



Imagen Original

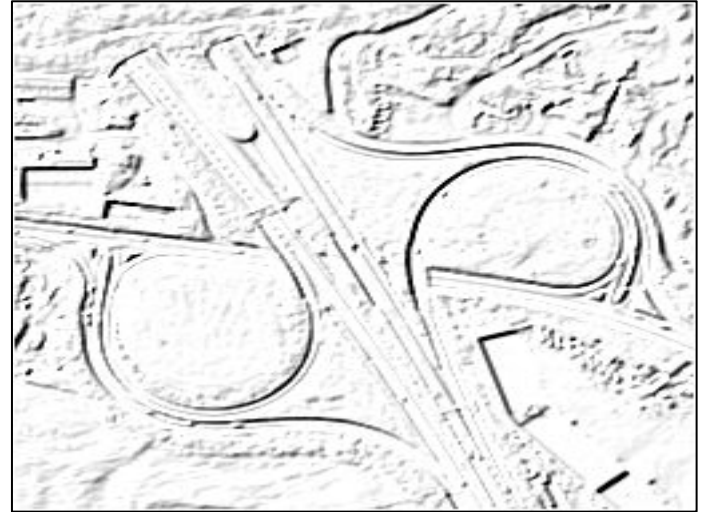
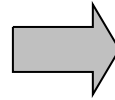


Operador de Kirsch de  $0^\circ$

# Operadores de Kirsch



Imagen Original



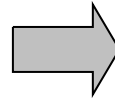
Operador de Kirsch de  $45^\circ$



# Operadores de Kirsch



.Imagen Original

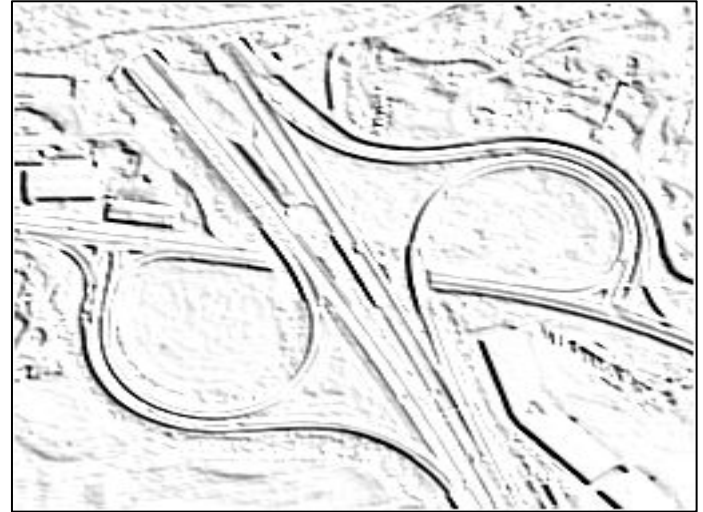
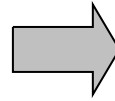


.Operador de Kirsch de  $90^\circ$

# Operadores de Kirsch



Imagen Original

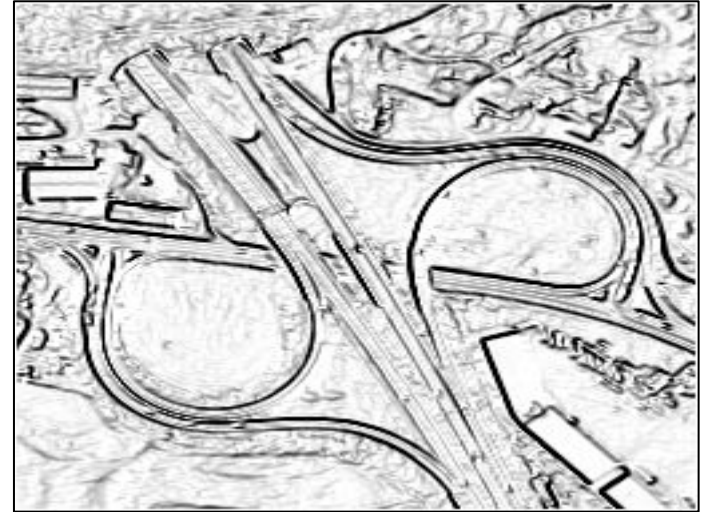
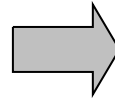


Operador de Kirsch de  $135^\circ$

# Operadores de Kirsch



Imagen Original



Operador de Kirsch  
con mayor magnitud



## Actividad de cierre

- ¿Qué aprendiste hoy?