

# MEJORA DE IMÁGENES

# Procesamiento en el Dominio Espacial

- $g(x,y) = T(f(x,y))$

f: imagen origen

g: imagen destino

T: operador que actúa sobre f

T está definido en (x,y) o en algún entorno o vecindad de (x,y)

Entorno de un punto (x,y): subimagen cuadrada o rectangular centrada en el punto (x,y). El nuevo valor depende de lo que sucede en la **vecindad** de un píxel.

Pueden existir entornos tales como diamantes o círculos, pero los más comunes son cuadrados y rectángulos.

## Filtrado 2D – Máscaras o Kernels

- Base para Algunas Operaciones de Filtrado

-Matriz

```
[[1  2  1  5]
 [2  5  3  1]
 [1  4  1  0]
 [1  3  0  2]]
```

-Máscara

```
[[0 2 3]
 [1 1 0]
 [1 2 1]]
```

- Pasos

Sobrepone el centro de la máscara sobre el elemento de interés.

```
[1  0  2  2  1  3  5]
[2  1  5  1  3  0  2]
[1  1  4  2  2  1  0]
[1   3   0   2]
```

Hace el producto punto

a-Multiplicar cada valor (peso) de la máscara por el píxel homólogo de la matriz

b.-Suma los productos individuales en el apartado anterior.

Se desplaza la máscara y se repite la operación.

El primer elemento es:  $1*0+2*2+1*3+2*1+5*1+3*0+1*1+4*2+2*1=$

El resultado entonces es:  $\begin{bmatrix} 25 & 31 \\ 31 & 21 \end{bmatrix}$

## Filtrado 2D – Máscaras o Kernels

Qué hacer con los píxeles de borde?

Con algún método se agregan filas y columnas en los extremos para tener una matriz resultado de las mismas dimensiones que la dada:

- Completar con ceros
- Repetir los valores de borde
- Completar con valores simétricos
- Dejarlos con el valor original

No es muy importante.

*Los productos de convolución y correlación están definidos de la misma forma, pero en la convolución hay inversión de la máscara.*

## Filtros Suavizantes: Filtro de la Media

Este filtro se obtiene con máscara, de la forma vista  
Tomamos máscara 3x3

**Media o Promediado en el entorno**  $1/9^*$

1	1	1
1	1	1
1	1	1

**Media ponderada- Filtro Gaussiano**

Le da más peso a ciertos píxeles  
(menor peso a los de mayor distancia)  $\frac{1}{16}^*$

Un tipo es el que usa una máscara gaussiana.  
Se obtiene un *blur* o imagen difuminada.

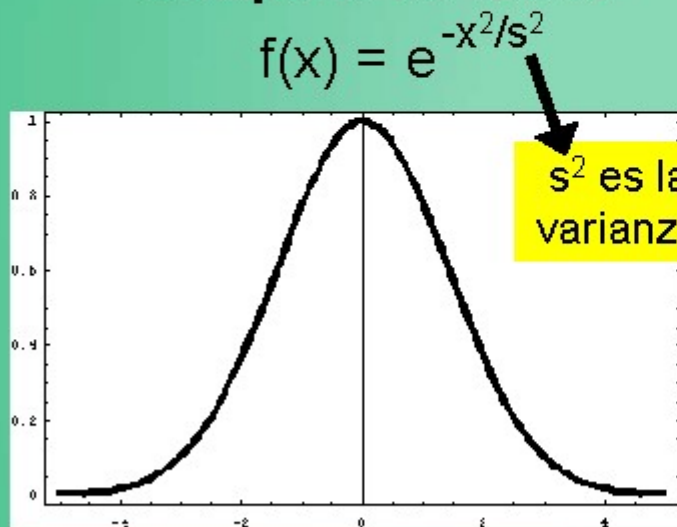
1	2	1
2	4	2
1	2	1

# Filtros Suavizantes: Filtro Gaussiano- Introducción

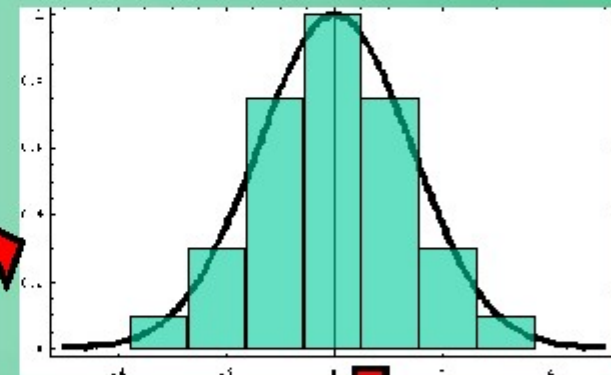
## Operadores de suavizado.

- **Suavizado gaussiano:** media ponderada, donde los pesos toman la forma de una campana de Gauss.
- **Ejemplo.** Suavizado gaussiano horizontal.

### Campana de Gauss



### Campana discreta



1/64 ·

1	6	15	20	15	6	1
---	---	----	----	----	---	---

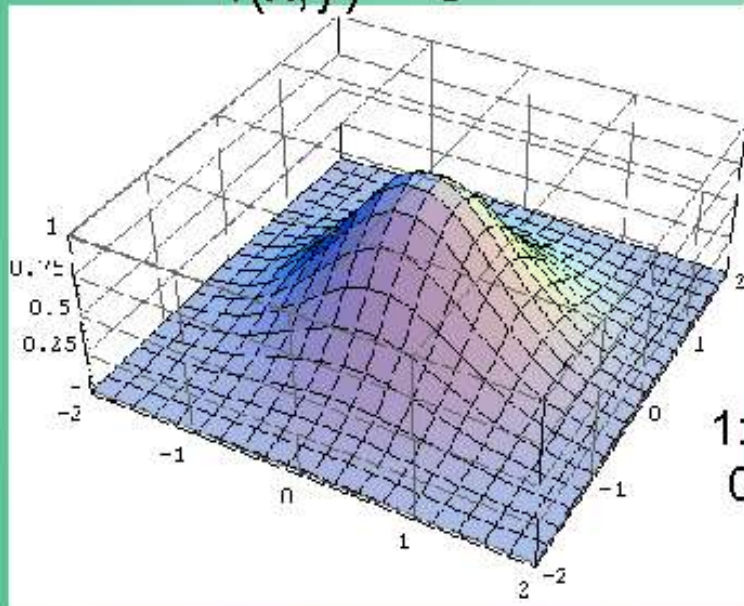
# Filtros Suavizantes: Filtro Gaussiano-2D (xy)

## Operadores de suavizado.

- Normalmente, el suavizado gaussiano se aplica en dos dimensiones. Los pesos de la máscara dependen de la distancia al píxel central.

### Campana de Gauss 2D

$$f(x,y) = e^{-(x^2+y^2)/s^2}$$

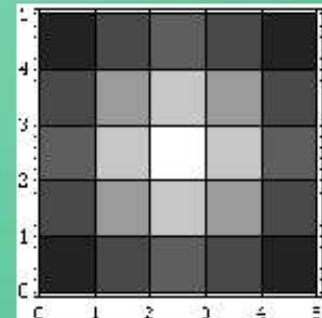
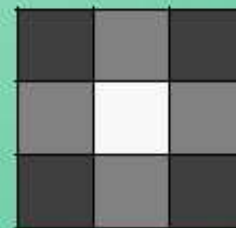


Máscara gaussiana de 3x3

1	2	1
2	4	2
1	2	1

1/16 ·

1: blanco  
0: negro



## Filtros Suavizantes: Filtro Gaussiano

El filtro gaussiano se usa para emborronar imágenes y eliminar ruido.

Es similar al filtro de media pero se usa una máscara diferente, modelizando la función gaussiana:

Ejemplo de máscara 5x5 para el filtro gaussiano con  $\sigma=1.0$ :

$1/256 \times$

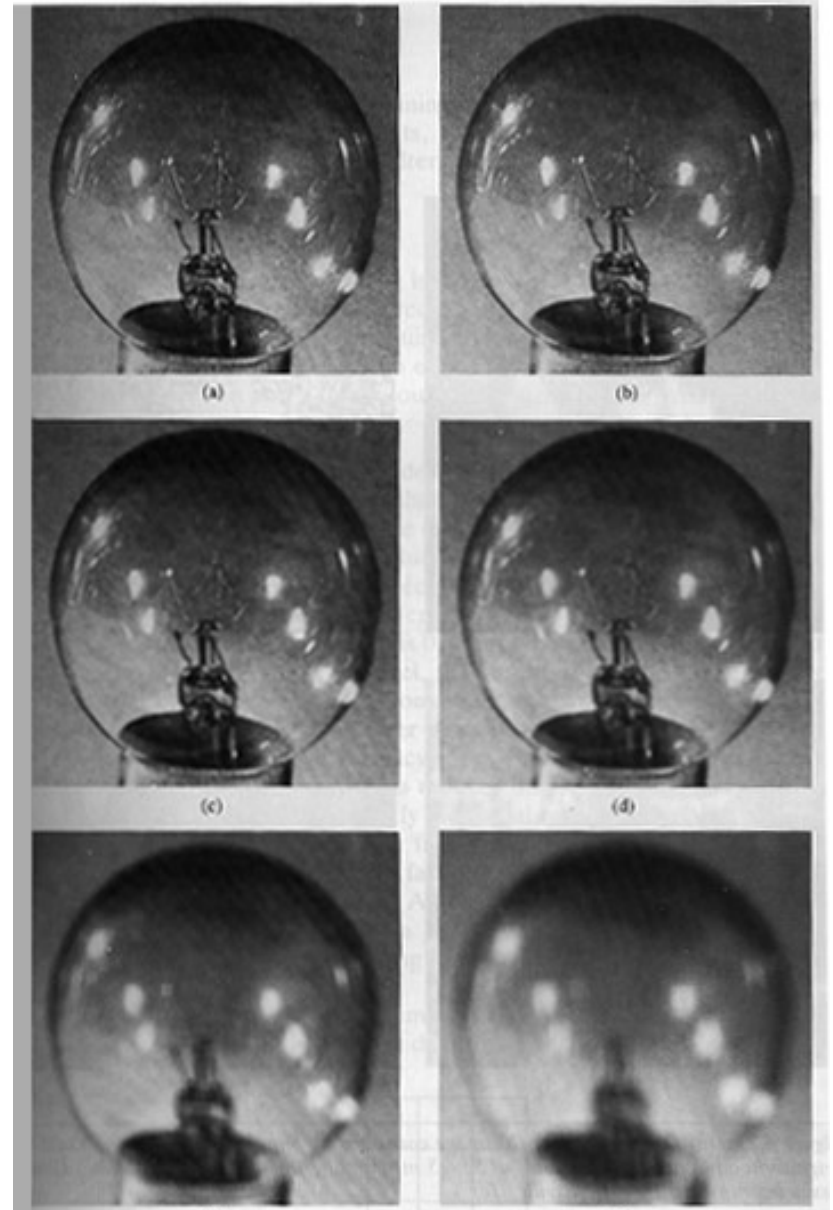
1	4	6	4	1
4	16	24	16	4
6	24	36	24	6
4	16	24	16	4
1	4	6	4	1



# Filtros Suavizantes

## Influencia del tamaño de la máscara

**Filtros de media aritmética** con tamaños de máscara de 3,5,7,15,25, se va perdiendo la silueta del filamento de la bombilla al hacer el promedio con los píxeles de alrededor, conforme más grande es la máscara.



# Filtros Suavizantes

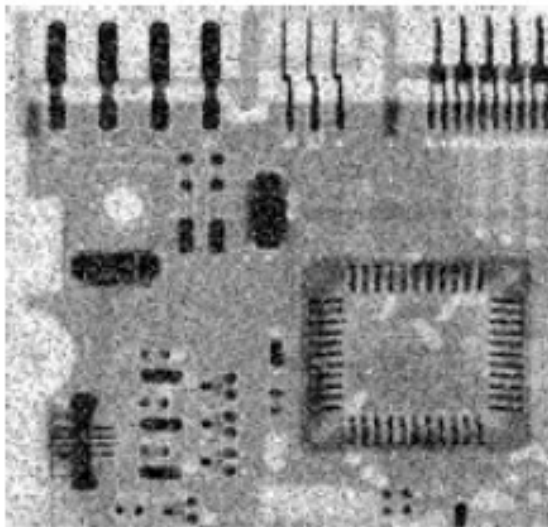
## Filtros de Orden: Filtro de Mediana

**Filtro de Mediana (es no lineal):** en lugar de reemplazar por el valor medio, reemplaza por la mediana. El filtro de media difumina los bordes y otros detalles. Este no.

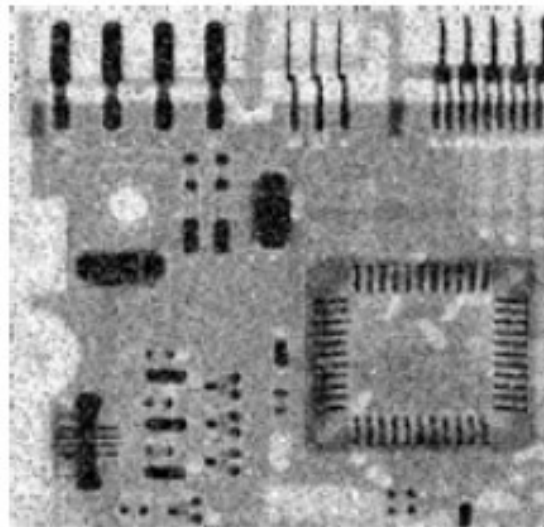
Mediana de un conjunto de valores es tal que la mitad de los valores del conjunto quedan por debajo de  $m$  y la otra mitad por encima.) No es un operador lineal.

Procedimiento: toma todos los píxeles de la máscara  $n \times n$ . Los ordena. Toma el elemento situado en  $(n \times n)/2$  del vector ordenado.

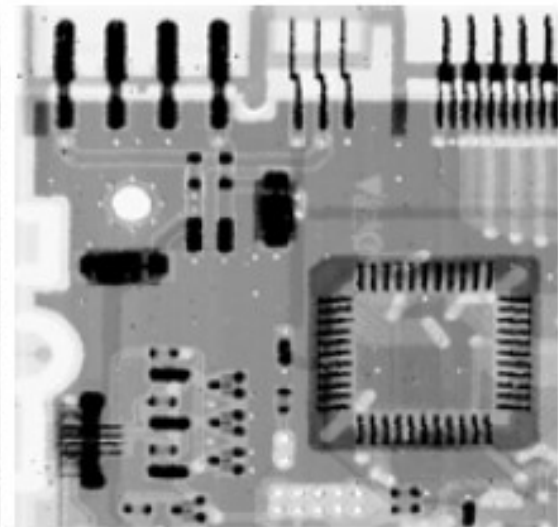
Elimina los valores extremos.



a) Imagen original



b) Filtrado de media



c) Filtrado de mediana

# Filtros Suavizantes- Filtros de Orden- Comparación

## Filtros de orden:

De mediana

De moda

De máximo

De mínimo

## Discusión General:

Ordenan los valores de una vecindad y toman, el central, el máximo o el mínimo, respectivamente.

Son de resultado impredecible, en especial el máximo y el mínimo. Estos últimos se pueden mejorar tomando no el mínimo sino el elemento que le sigue, lo mismo con el máximo.

Todos estos filtros, incluido el de media, pierden información.

***Filtros de media:*** se adaptan mejor al ruido gaussiano.

***Filtros de orden:*** se adaptan mejor al ruido sal y pimienta y de Raleigh. El de máximo es mejor para valores bajos de ruido (sal) y el de mínimo para eliminar valores altos (pimienta).

# Detección de Puntos aislados

Para la detección de puntos aislados se puede usar una máscara, que separa alta frecuencia. En este caso:

$$R = -z_1 - z_2 - z_3 - z_4 - z_6 - z_7 - z_8 + 8z$$

Si  $|R| > T$  donde **T es un umbral (threshold no negativo)** comparamos las diferencias entre un punto y sus vecinos.

Si la diferencia es grande se trata de un punto aislado.

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

# Detección de Puntos aislados

## Ejemplo

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

Máscara

Matriz Imagen

1	1	1	1	1
1	10	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Matriz Resultado

-	-	-	-	-
-	72	-9	0	-
-	-9	-9	0	-
-	0	0	0	-
-	-	-	-	-

No se consideran los bordes.

***Según el umbral de valor  $T$  se obtienen, según el valor asignado al umbral:***

4 puntos con umbral ( $0 < T \leq 9$ ). Por ejemplo  $\text{abs}(T)=5$  .

1 punto si ( $9 < \text{abs}(T) \leq 72$ ) El que tiene valor 10

Ningún punto si  $\text{abs}(T) > 72$

# **DETECCIÓN BORDES**

# Bordes - Contornos

Contienen mucha información.

**Los bordes son cambios importantes en la intensidad de la imagen.**

No se deben al ruido, ni al desvanecimiento.

Es difícil hallar los bordes en imágenes ruidosas.

ya que el ruido agrega bordes.

El contorno de un objeto se forma a partir de bordes.



## Bordes - Contornos



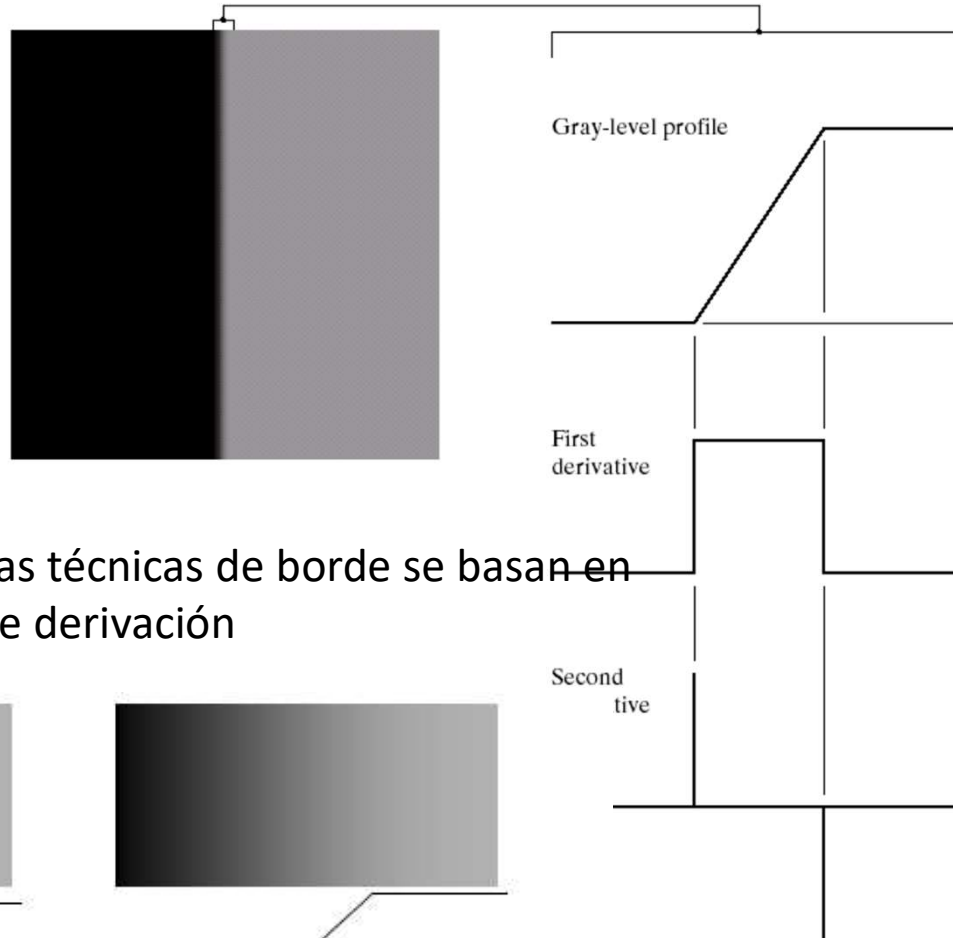


# Los bordes y las derivadas

a b

**FIGURE 10.6**

(a) Two regions separated by a vertical edge.  
(b) Detail near the edge, showing a gray-level profile, and the first and second derivatives of the profile.

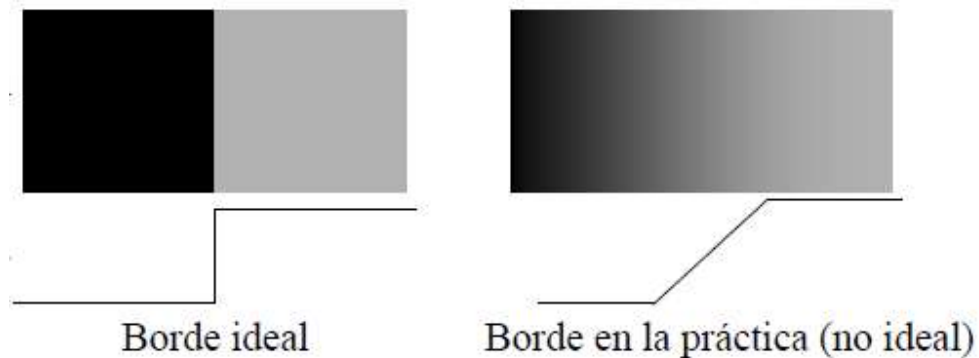


## Signos:

Derivada primera de la rampa: positiva y constante (pasa de oscuro a claro).

Derivada segunda: positiva en el lado oscuro de la transición y negativa en el lado claro. Cero en la rampa y en las zonas de Intensidad constantes

La mayor parte de las técnicas de borde se basan en un operador local de derivación



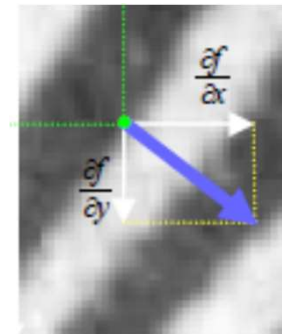
# Operadores

## Operador Derivada de Primer Orden:

**Gradiente:** vector que indica el máximo cambio de intensidad.

El gradiente es un vector perpendicular al borde.

$$f(x,y):$$
$$\vec{\nabla} f = \frac{\partial f}{\partial x} \vec{i} + \frac{\partial f}{\partial y} \vec{j}$$



Su **dirección** y **magnitud** están dadas por:

$$\theta = \tan^{-1} \left\{ \frac{\frac{\partial f}{\partial x}}{\frac{\partial f}{\partial y}} \right\} \quad \text{y} \quad \sqrt{\left( \frac{\partial f}{\partial x} \right)^2 + \left( \frac{\partial f}{\partial y} \right)^2}$$

Esta expresión es aproximada de diferentes formas.

# Operadores

- **Operador Gradiente**

**Aproximación digital del gradiente:** la más sencilla es

$$G_x \approx \Delta_x f(i, j) = f(i, j+1) - f(i, j)$$

$$G_y \approx \Delta_y f(i, j) = f(i, j) - f(i+1, j)$$

Equivalente a hacer la convolución con:

$$G_x = \begin{bmatrix} -1 & 1 \end{bmatrix}$$

$$G_y = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

Dada una fila  $x$  resta dos píxeles adyacentes. Dada una columna  $y$  resta dos píxeles

Se llama también gradiente de fila.

adyacentes.

Se llama también gradiente de columna.

# Operadores - Aproximaciones

$$|G| = |G_x| + |G_y|$$

Se puede aproximar el módulo con la suma de valores absolutos porque:

No interesa el valor en sí, sino en forma relativa.

Tiene menor tiempo de cómputo que calcular el módulo.

Se tiene en cuenta un umbral.

$$g(x, y) = \begin{cases} 1 & \text{si } G(f(x, y)) > T \\ 0 & \text{si } G(f(x, y)) \leq T \end{cases}$$

# Máscaras de Convolución para Cálculo de Bordes

Para computar en un píxel central  $i, j$  la máscara debe ser impar x impar, 3x3, por ejemplo.

$$G_{col} = \frac{1}{4} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad G_{fila} = \frac{1}{4} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Convención seguida por algunos autores:  
la primera es  $G_x$  y la segunda  $G_y$ .

Una es la transpuesta de la otra.

La primera permite obtener variaciones por columna.

La segunda por fila.

## Tamaño de la máscara:

A mayor tamaño, menos sensibilidad al ruido,

menor precisión en la ubicación

A mayor tamaño, más complejidad computacional.

# Máscaras Clásicas para Bordes

## Basadas en el Gradiente

-1	0	0	-1
0	1	1	0

Robertis

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

Prewitt

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

Sobel

Son operadores diferenciales.

La suma de los elementos siempre es 0, de este modo la respuesta es 0 en caso de áreas constantes.

# Máscara de Sobel

Es un operador que se usa mucho.

El resultado del cálculo se aplica al punto central de la máscara.

*Produce, como efecto colateral, un suavizado.*

Pone énfasis en los puntos más cercanos a  $i, j$  (no tanto en las diagonales).

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

$G_f$

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$G_c$

# Máscara de Prewitt

Similar a Sobel.

**Diferencia respecto a Sobel: *No pone énfasis en ningún punto en especial.***

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

$G_f$

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

$G_c$



## Ejemplo de cálculo

$$\begin{bmatrix} 0 & 8 & 8 & 8 & 8 & 8 \\ 1 & 8 & 9 & 7 & 6 & 8 \\ 2 & 3 & 4 & 8 & 8 & 8 \\ 2 & 2 & 2 & 8 & 8 & 8 \\ 2 & 2 & 2 & 2 & 8 & 8 \end{bmatrix}$$

$$G_c = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$G_f = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Queremos calcular el gradiente en el punto marcado.

Si aplicamos los operadores de Sobel obtenemos:

$$G_c = 26$$

$$G_f = -12$$

Si el umbral es  $T=30$

$$|G| = |G_f| + |G_c| = 38 > 30 \quad \text{hay borde}$$

Si el umbral  $T$  fuera mayor que 38 no sería considerado borde.

# Ejemplo Sobel



Sobel total



Horizontal



Vertical



# Ejemplo Sobel

Sobel total



Horizontal



Vertical



# Operadores Basados en la Derivada

## Segunda

Las **máscaras basadas en el gradiente** son adecuadas para **transiciones de niveles de gris bruscas** (función tipo escalón o paso).

Las máscaras basadas en el gradiente son **poco sensibles a cambios graduales en los niveles de gris** (funciones tipo rampa).

Para estas funciones (cambios graduales en los niveles de gris) conviene usar operadores basados en las derivadas de segundo orden, como el operador laplaciano.

En el caso de imágenes digitales (discretas) el operador se obtiene aplicando consecutivamente los operadores gradiente en dirección  $x$  e  $y$ .

# Operador Laplaciano

Trabaja considerando la derivada segunda.

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

Laplaciano de una función bidimensional

La aproximación para una función discreta es:

$$\nabla^2 f(m, n) = D_x(D_x(f(m, n))) + D_y(D_y(f(m, n)))$$

$$\nabla^2 f(m, n) = D_x(f(m+1, n) - f(m, n)) + D_y(f(m, n+1) - f(m, n))$$

$$\nabla^2 f(m, n) = [(f(m+1, n) - f(m, n)) - (f(m, n) - f(m-1, n))] + [(f(m, n+1) - f(m, n)) - (f(m, n) - f(m, n-1))]$$

$$\nabla^2 f(m, n) = f(m+1, n) + f(m, n+1) + f(m-1, n) + f(m, n-1) - 4f(m, n)$$

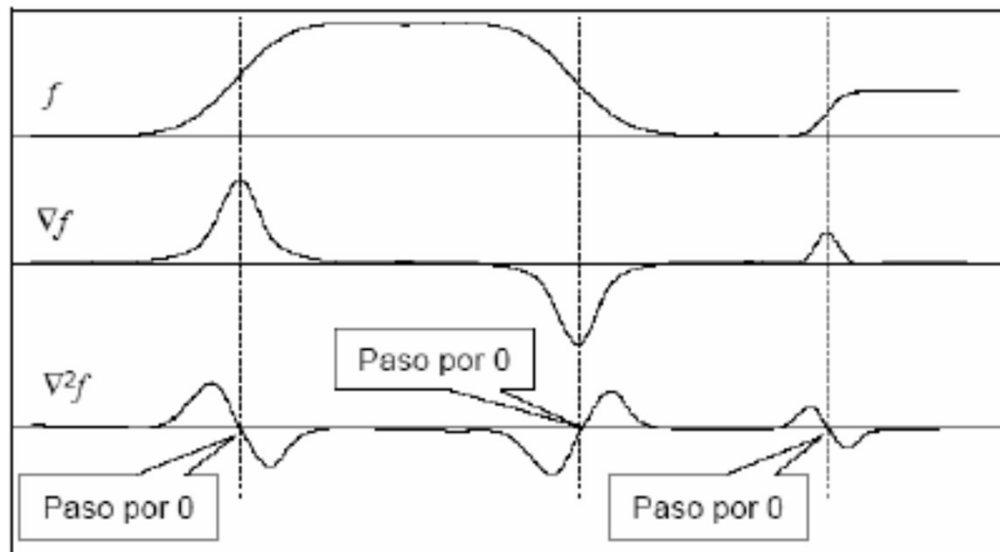
Resulta la siguiente matriz de convolución:

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

$$\nabla^2 f = f * M$$

O sea que se puede interpretar como un filtro de primer orden.

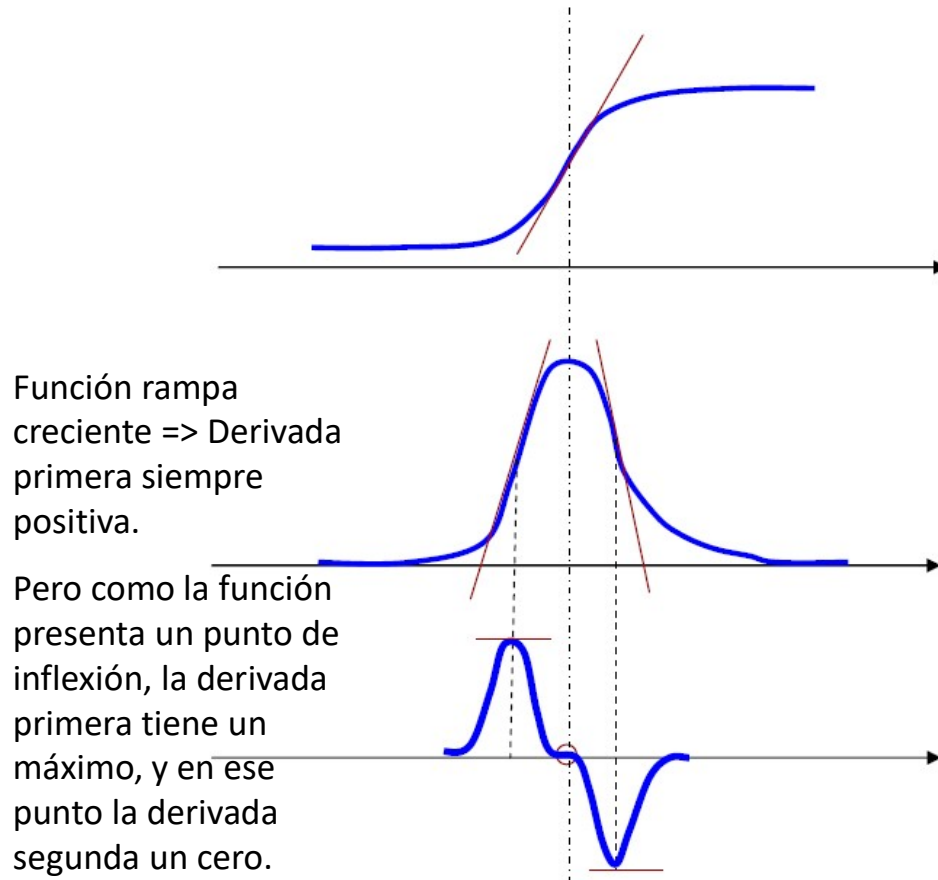
## Diferencia entre el accionar del Gradiente y del Laplaciano



Como el laplaciano trabaja con las derivadas segundas **es más sensible al ruido** que los anteriores.

Una vez aplicada la máscara se tiene en cuenta un umbral.

# Diferencia entre el accionar del Gradiente y del Laplaciano



La **derivada segunda** de una función rampa suavizada que puede representar el borde de un objeto, es una función que presentará un máximo y un mínimo, en lugar de ser una función con un máximo único (caso de la derivada primera).

Lo que se busca para determinar los bordes son los **puntos de cruce por cero** en la derivada segunda (imagen filtrada), es decir los cambios de signo, de positivo a negativo y viceversa.

## Ventajas del Operador Laplaciano

Detecta bordes, **sin importar la dirección de los mismos (las máscaras son simétricas rotacionalmente)**.

Se basan en que cuando hay un borde en una dirección, la derivada primera pasa por un máximo (mínimo) y la segunda cruza por cero.

Localizan bien, si la relación señal/ruido es alta.

No precisa dos pasadas. (Contrariamente los filtros que trabajan con la derivada primera requieren dos pasadas: una para el borde horizontal y otra para el vertical y luego un cálculo).

Da bordes más gruesos.

Los otros métodos pueden dejar puntos de lado.

## Desventajas del Operador Laplaciano

Es muy **sensible al ruido**, por lo que puede mostrar bordes falsos.

Produce bordes dobles.

**No detecta direcciones.**

Una máscara laplaciana que tiene en cuenta las diagonales:

Conclusión: no se usa solo. Se usa para detectar cruces por cero, o con el propósito complementario de detectar si un píxel está en el lado claro u oscuro.

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



# Laplaciano de Gaussiano o LOG (Laplacian of Gaussian)

La segunda derivada es muy afectada por el ruido.

***Por eso se hace un filtrado previo mediante un filtro Gaussiano.***

Con sigma se controla el grado de suavización (borrosidad) deseado (mayor sigma, más borroso) del filtro gaussiano.

A mayor sigma mayor suavizado.

pero altera los bordes y produce difuminado

Sigmas muy bajos dejan ruido

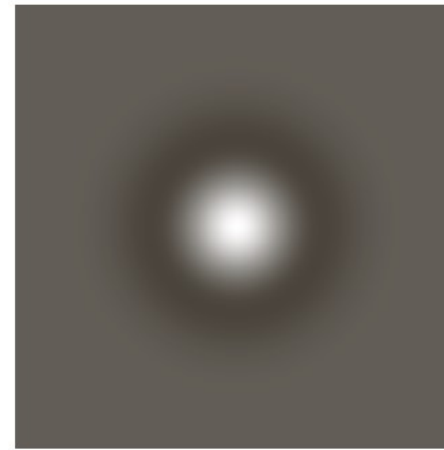
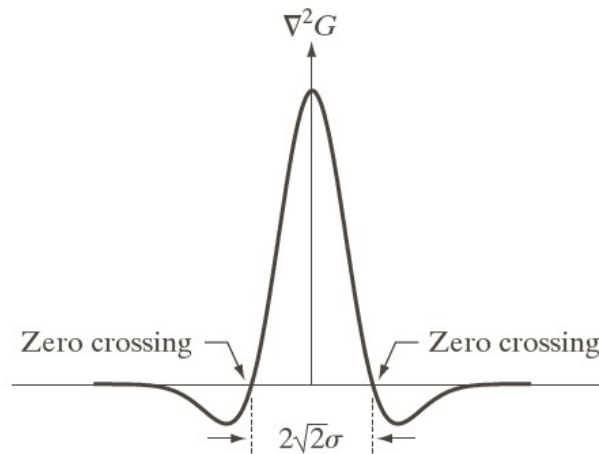
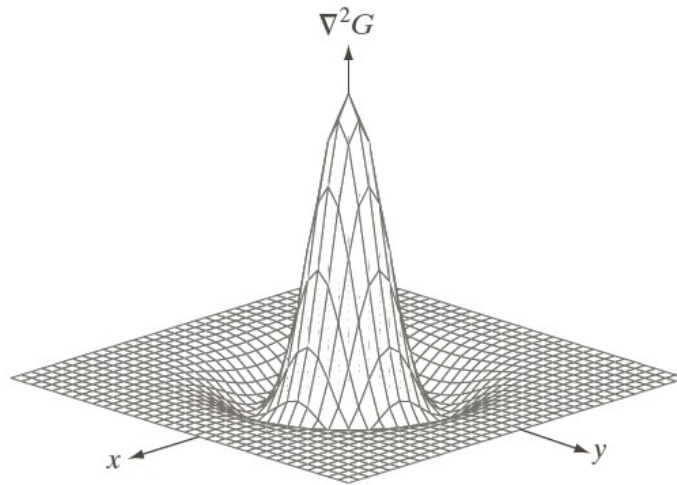
y producen falsos bordes.

El tamaño del filtro adecuado no se conoce por anticipado.

## **Criterio de Detección:**

1. Cruce por cero de la derivada segunda.
2. Pico muy alto (mayor que un umbral en la derivada primera)

# Laplaciano de Gaussiano o LOG



0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

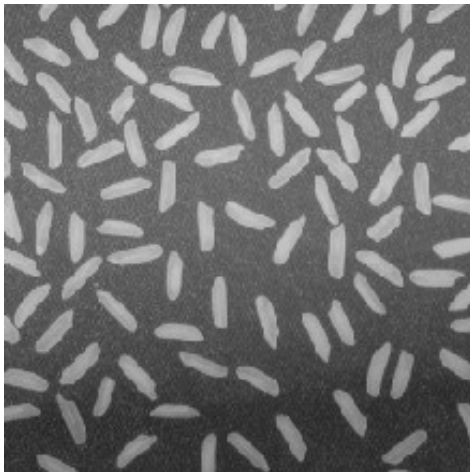
a b  
c d

**FIGURE 10.21**

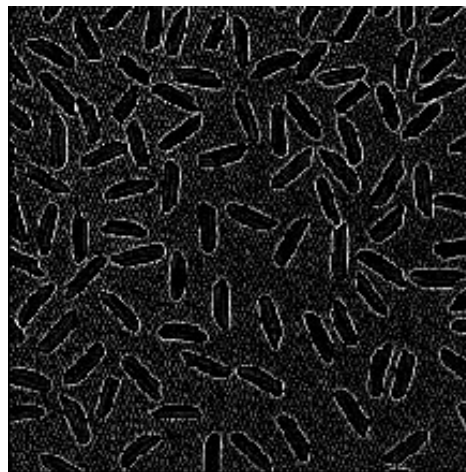
(a) Three-dimensional plot of the *negative* of the LoG. (b) Negative of the LoG displayed as an image. (c) Cross section of (a) showing zero crossings. (d)  $5 \times 5$  mask approximation to the shape in (a). The negative of this mask would be used in practice.

# Laplaciano y LOG

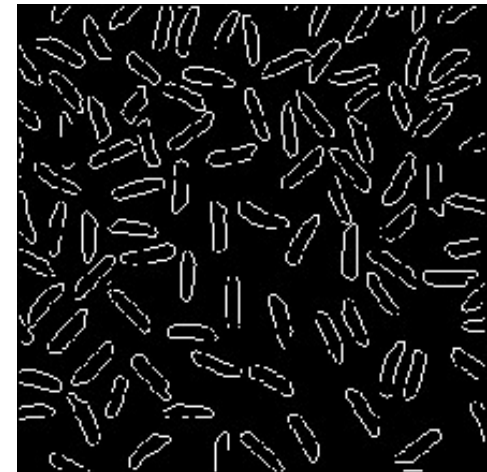
Rice.tif



Laplaciano



log

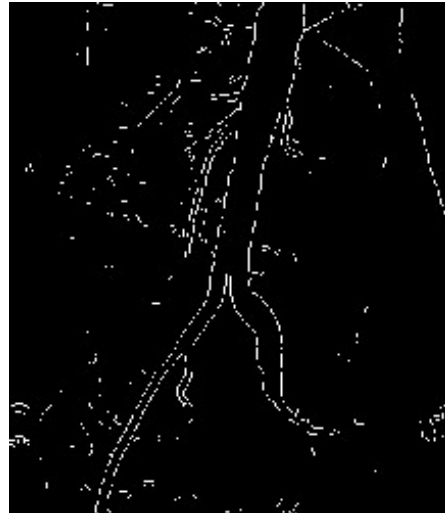


# Laplaciano de Gaussiano o LOG

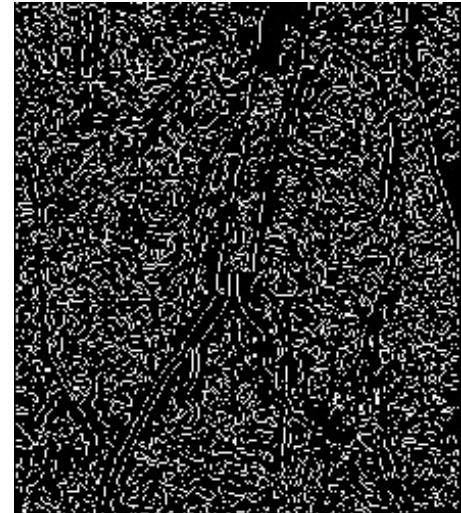
original



LOG



LOG con umbral 0



Efecto Spaghetti

# Filtro de Canny

Es un **buen filtro** cuando hay geometrías regulares.

Es **más complejo**.

Debe hallar los puntos de máximo gradiente.

Debe evitar el ruido que produce falsos bordes.

No debe identificar varios píxeles donde solo exista uno.

## Etapas

1) Se suaviza la imagen mediante un **filtro gaussiano**, con un valor dado de sigma.

2) Estima **el gradiente** (magnitud y dirección) para cada píxel de la imagen.

3) En esta matriz, suprime los puntos que no son máximos. Para esto compara cada punto con sus vecinos y umbraliza.

4) **Supresión no maximal** Asigna un umbral para eliminar los bordes espurios. Es lo más delicado porque si se asigna mal aparece ruido.

Trabaja con **dos umbrales**: uno inferior y otro superior. Por debajo del inferior rechaza y por encima del superior acepta (bordes fuertes). El resto corresponde a los llamados bordes débiles (entre  $T_1$  y  $T_2$ ).

Puede tomar  $T_2 \approx 2T_1$ .

5) **Enlace de bordes**: a los bordes débiles los acomoda: si están **conectados** a los píxeles cuyo valor es superior al umbral superior se aceptan. Se consideran así, si están dentro de un entorno 3x3 del borde fuerte. Así se eliminan los falsos bordes.

# Filtro de Canny

El resultado es una imagen binaria.

