

Procesamiento de Imágenes Biomédicas



Clase 2: Filtros Espaciales

Filtrado espacial

- Ya trabajamos procesando sólo al pixel individualmente.
- Ahora vamos a hacer un procesamiento en una vecindad de cada pixel.
- Este proceso consistirá en :
 - Definir un pixel central
 - Ejecutar la operación sobre los pixels de una zona predefinida (vecindad)
 - El resultado de la operación es la respuesta del proceso en el pixel
 - Repetir el proceso para los demás pixels

Recordemos.....

- Interpretación de convolución: “rotábamos una función, la desplazábamos y calculábamos áreas comunes. Integral ó sumatoria según continuo ó discreto.
- Correlación: “desplazábamos una función respecto de otra, integrábamos ó sumábamos”, sin rotación.

Correlación y Convolución en 1D

- Para ejecutar la convolución, rotamos 180° a w y colocamos el valor de "más a la derecha" de w coincidiendo con el origen de $f(j)$.
- Completamos con ceros (k)
- El primer valor de la convolución es la suma de productos en la posición mostrada (k).
- Movemos una posición a la derecha y repetimos el proceso (l).
- (n) es la última posición

Correlación y Convolución en 1D

- El resultado en (o) llamado "full", implica un cálculo en la manera vista con la imagen completada con ceros. Recordar la convolución daba la suma de los 2 anchos.
- El resultado en (p) llamado "same" tiene el mismo tamaño que f y se calcula agregando ceros a la imagen pero se comienza con el punto central de la máscara del filtro (valor 3).
- A la izquierda se muestra la correlación.

Correlation

Convolution

(a) \swarrow Origin f w
 0 0 0 1 0 0 0 0 1 2 3 2 0

\swarrow Origin f w rotated 180°
 0 0 0 1 0 0 0 0 0 2 3 2 1 (i)

(b) \downarrow
 0 0 0 1 0 0 0 0
 1 2 3 2 0
 \uparrow Starting position alignment

0 0 0 1 0 0 0 0 (j) $_$
 0 2 3 2 1

(c) Zero padding
 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 1 2 3 2 0

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (k)
 0 2 3 2 1

(d) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 1 2 3 2 0
 \uparrow Position after one shift

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (l)
 0 2 3 2 1

(e) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 1 2 3 2 0
 \uparrow Position after four shifts

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (m)
 0 2 3 2 1

(f) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
 1 2 3 2 0
 Final position \uparrow

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (n)
 0 2 3 2 1

(g) 'full' correlation result
 0 0 0 0 2 3 2 1 0 0 0 0

'full' convolution result (o)
 0 0 0 1 2 3 2 0 0 0 0 0

(h) 'same' correlation result
 0 0 2 3 2 1 0 0

'same' convolution result (p)
 0 1 2 3 2 0 0 0

Convolución por rotación y desplazamiento de la matriz de peso

- En nuestro caso discreto (imágenes) la matriz de peso es una matriz de números.
- Este filtro es el que rotamos y desplazamos.
- El pixel (r,c) en la imagen de salida es la suma pesada de los pixels de la imagen original vecinos a (r,c) .
- Cada pixel es multiplicado por el peso correspondiente.
- La suma de esos productos es el valor del pixel de la imagen final.

Correlación en 2 dimensiones

Initial position for w	Full correlation result	Cropped correlation result
$\begin{array}{ c c c } \hline 1 & 2 & 3 \\ \hline \end{array}$	0 0 0 0 0 0 0 0	0 0 0 0 0
$\begin{array}{ c c c } \hline 4 & 5 & 6 \\ \hline \end{array}$	0 0 0 0 0 0 0 0	0 9 8 7 0
$\begin{array}{ c c c } \hline 7 & 8 & 9 \\ \hline \end{array}$	0 0 0 0 0 0 0 0	0 6 5 4 0
0 0 0 0 0 0 0 0	0 0 0 9 8 7 0 0 0	0 3 2 1 0
0 0 0 0 1 0 0 0 0	0 0 0 6 5 4 0 0 0	0 0 0 0 0
0 0 0 0 0 0 0 0 0	0 0 0 3 2 1 0 0 0	
0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	

Imagen de entrada

		I_0	I_1	I_2					
		I_3	I_4	I_5					
		I_6	I_7	I_8					

Ventana de convolución

I_0	I_1	I_2
I_3	I_4	I_5
I_6	I_7	I_8

Máscara de convolución

M_0	M_1	M_2
M_3	M_4	M_5
M_6	M_7	M_8

×

Imagen de salida

Nuevo píxel = $I_0 \times M_0 + I_1 \times M_1 + I_2 \times M_2 +$
 $I_3 \times M_3 + I_4 \times M_4 + I_5 \times M_5 +$
 $I_6 \times M_6 + I_7 \times M_7 + I_8 \times M_8$

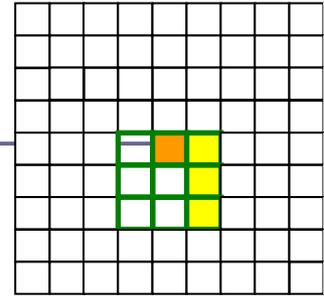
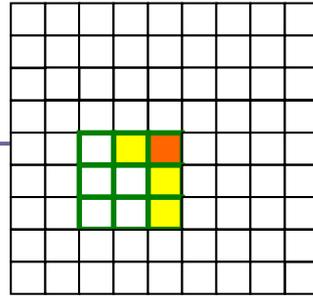
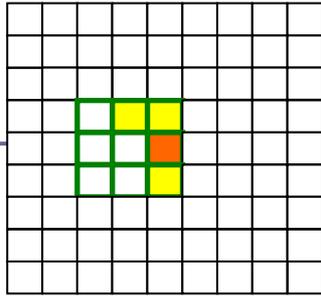
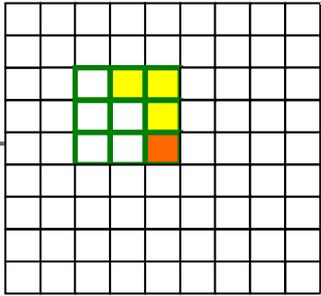
Correlación y Convolución en 2D

- ❑ La operación matemática en que consiste la convolución es simplemente una suma “pesada” de píxeles vecinos del píxel fuente.
- ❑ Los pesos son determinados por una matriz llamada máscara de convolución, que determina los coeficientes a aplicar sobre los puntos de una determinada área.
- ❑ Las dimensiones de la matriz son normalmente impares e iguales, de forma que se pueda determinar un centro de una matriz cuadrada.
- ❑ La posición del valor central se corresponde con la posición del píxel de salida.

Correlación y Convolución en 2D

- La suma de los pesos en la máscara de convolución afecta a la intensidad general de la imagen resultante.
- Máscaras de convolución tienen coeficientes que suman un total de 1 (como los filtros paso bajo usados en el suavizado de imagen)
- Máscaras de convolución (como las usadas en detección de bordes) tienen coeficientes positivos y negativos y suman un total de 0

Desplazamiento del filtro



Resultado suma de productos

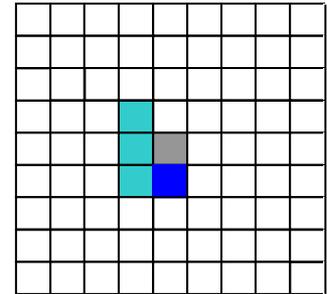
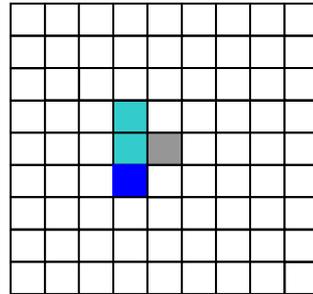
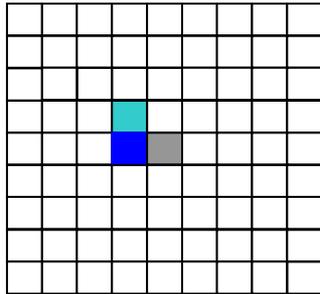
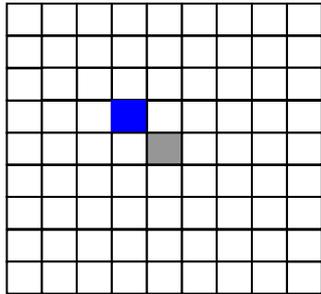
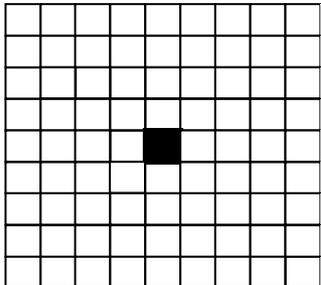


Imagen original

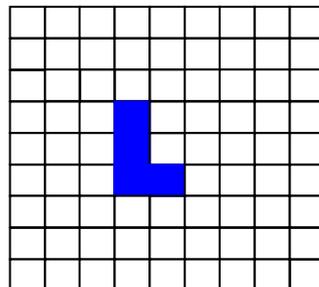
Filtro



*

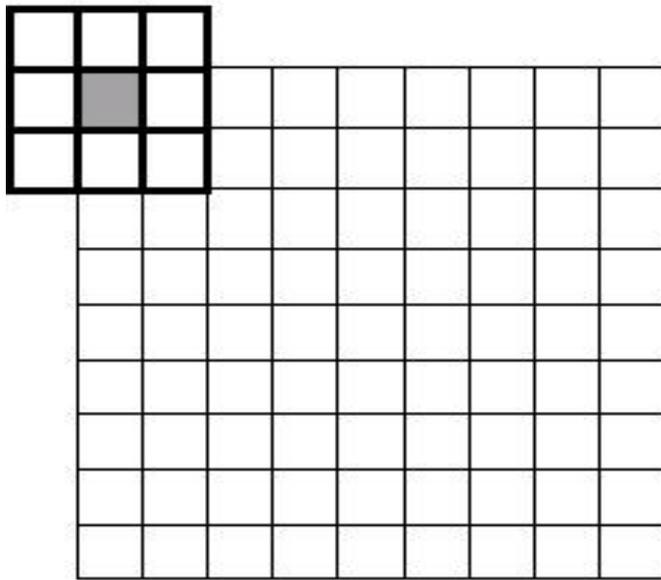


=



Resultado final. Se muestran sólo los productos distintos de cero

Otro problema



La primera cuestión al implementar una función de convolución es como tratar los bordes de la imagen. Cuando la ventana de convolución se centra en el primer píxel de una imagen en $(0,0)$, la ventana deslizante sobresale de la imagen en el borde superior y en el izquierdo. ¿Qué valores de la imagen deben ser multiplicados por esos coeficientes de convolución que sobresalen de los bordes?

Varias soluciones.....primera

- La primera solución es tratar las celdas vacías en la ventana de convolución como ceros. Esto se conoce como zero-padding. Es fácil de realizar pero no es una buena idea si los bordes de la imagen resultante son tan importantes como el resto de la imagen.

Segunda.....

- La siguiente solución es empezar la convolución en la primera posición donde la ventana no sobresalga de la imagen. Si la máscara de convolución es de tamaño 3×3 , se empezaría convolucionando con el píxel en $(1,1)$ en vez del píxel en $(0,0)$. Este método es sencillo de implementar. En la imagen de salida, los bordes convolucionados son copiados para crear una imagen con el mismo tamaño que la imagen de entrada.

Otra....

- El otro método es “envolver” la imagen, es decir, considerar como píxel contiguo al del borde izquierdo, el píxel del borde derecho y viceversa, así como con los del borde superior e inferior. Si se quiere convolucionar una imagen 256x256 con una máscara de convolución de 3x3, la primera ventana de convolución opera con los píxeles en las posiciones que se ven en la siguiente figura. Este último método ha sido el empleado en la implementación de la aplicación.

En Matlab

- El filtrado lineal espacial está implementado por la función :

```
g=imfilter (f, w, filtering_mode,  
boundary_options, size_options)
```

f= imagen de entrada

w= máscara del filtro

filtering_mode = corr ó conv

boundary_options= p:padding (completar con ceros), r:replicate (repetir el valor de los bordes)

size_options= full, same.

Filtros espaciales

- Los filtros espaciales tienen como objetivo modificar la contribución de determinados rangos de frecuencias a la formación de la imagen. El término espacial se refiere al hecho de que el filtro se aplica directamente a la imagen y no a una transformada de la misma, es decir, el nivel de gris de un píxel se obtiene directamente en función del valor de sus vecinos.

Clasificación

- Los filtros espaciales pueden clasificarse basándose en su linealidad: *filtros lineales* y *filtros no lineales*. A su vez los *filtros lineales* pueden clasificarse según las frecuencias que dejen pasar: los *filtros paso bajo* ; los *filtros paso alto*; los *filtros pasa banda*.

-
- La forma de operar de los filtros lineales es por medio de la utilización de máscaras que recorren toda la imagen centrando las operaciones sobre los píxeles que se encuadran en la región de la imagen original que coincide con la máscara y el resultado se obtiene mediante una computación (suma de convolución) entre los píxeles originales y los diferentes coeficientes de las máscaras.

-
- Los *filtros espaciales no lineales* también operan sobre entornos. Ejemplos de filtros no lineales habituales son los filtros mínimo, máximo y de mediana que son conocidos como *filtros de rango*. El filtro de mediana tiene un efecto de difuminado de la imagen, y permite realizar una eliminación de ruido, el filtro de máximo se emplea para buscar los puntos más brillantes de una imagen produciendo un efecto de *erosión*, y el filtro de mínimo se emplea con el objetivo contrario, buscar los puntos más oscuros de una imagen produciendo un efecto de *dilatación*.

Otra clasificación

- Filtros espaciales: basándose en su finalidad, y así tenemos:
- *Filtros de suavizado* (Smoothing) para difuminar la imagen.
- *Filtros de realce* (Sharpening) para eliminar zonas borrosas
- Filtros diferenciales que se componen de varios tipos de máscaras (Laplaciano, Prewitt, Sobel, etc.), y se utilizan para la *detección de bordes*. El proceso de detección de bordes se basa en realizar un incremento del contraste en las zonas donde hay una mayor diferencia entre las intensidades, y en una reducción de éste donde no tenemos variación de intensidad.

Filtro espacial: suavizado

- El suavizado de imágenes se utiliza normalmente bajo dos supuestos: dar a una imagen un difuminado o efecto especial y para la eliminación de ruido
- El *suavizado o filtrado espacial paso bajo* borra los detalles más finos de una imagen, atenúa las altas frecuencias

Filtros de suavizado

- ❑ Los filtros de suavizado se utilizan principalmente para el procesamiento de una imagen con el objeto de remover pequeños detalles.
- ❑ En general funcionan promediando los pixels vecinos.
- ❑ Como el ruido en general es una función impulsiva sumada a la imagen, estos filtros tienden a eliminar el ruido. Sin embargo los bordes que también son abruptos tienden a ponerse borrosos con estos filtros.

Filtros pasa bajo (suavizado)

- Las máscaras de los filtros paso bajo deben tener todos sus coeficientes positivos y la suma de ellos debe ser igual a uno. Las máscaras de tamaño 3x3 más utilizadas son las siguientes:
- La llamada propiamente paso bajo
- La llamada "Smooth"

$$\frac{1}{9} \times$$

1	1	1
1	1	1
1	1	1

$$\frac{1}{16} \times$$

1	2	1
2	4	2
1	2	1

Filtros de realce (pasa altos)

- El objetivo principal del *realce* es el de destacar los detalles finos de una imagen
- El problema puede haber sido el método de adquisición de la imagen
- El realce de una imagen aumentará su contraste
- Utilización: imágenes médicas, detección de blancos de armas inteligentes

Filtros pasa altos

- Se utilizarán filtros pasa altos
- Atenúa las componentes bajas de frecuencia
- Dejará pasar componentes de alta frecuencia
- Mostrará los detalles
- Para implementar un *filtro paso alto*, es necesario que el filtro posea coeficientes negativos en la periferia y positivos en el centro.
- Así, cuando la máscara se encuentra sobre una zona uniforme, la salida proporcionada por la máscara será 0 o próxima a dicho valor.

Filtros pasa altos

- Algunos filtros

- $$\begin{array}{ccc} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{array} \quad \begin{array}{ccc} -2 & 0 & -2 \\ 0 & 8 & 0 \\ -2 & 0 & -2 \end{array} \quad \begin{array}{ccc} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{array}$$

- Cuando la máscara está sobre una zona de la imagen de poco contraste o pequeña variación del nivel de gris, la salida de la máscara es cero o muy pequeña.

Filtros pasa altos

- El filtrado paso alto disminuye considerablemente el número de grises presentes en la imagen original, con lo que se reduce el contraste global de la imagen

Filtros sharpen

- Los filtros pasa altos recién vistos disminuyen el contraste global de la imagen.
- Para realce se usan los filtros *sharpen*, que son una variedad de las máscaras ya vistas.

1	-2	1	-1	-1	-1	0	-1	0
-2	5	-2	-1	9	-1	-1	5	-1
1	-2	1	-1	-1	-1	0	-1	0

Sharpen 1

Sharpen 2

Sharpen 3

Resumiendo pasa altos

- ❑ Se aumenta en 1 el valor del centro al pasa altos para obtener la máscara sharpen correspondiente.
- ❑ Resumiendo:
- ❑ El filtro pasa altos disminuye el número de grises presentes en la imagen original, reduciendo el contraste global de la imagen.
- ❑ Se mejora la nitidez y los detalles al procesar la imagen con filtros sharpen.
- ❑ Con la máscara pasa alto se detectan los detalles de la imagen, detalles que se verán realzados con un filtro sharpen.

Filtros derivativos

- El hacer un promedio de unos píxeles sobre una región de la imagen, tiende a emborronar los detalles de una imagen. Como el promedio es similar a la integración, la diferenciación tendrá el efecto opuesto y ello realzará los detalles de una imagen.
- El método de diferenciación más común en el proceso de aplicaciones es el gradiente. Para una función $f(x,y)$, el gradiente de f en las coordenadas (x,y) se define como el vector:

Filtros derivativos (2)

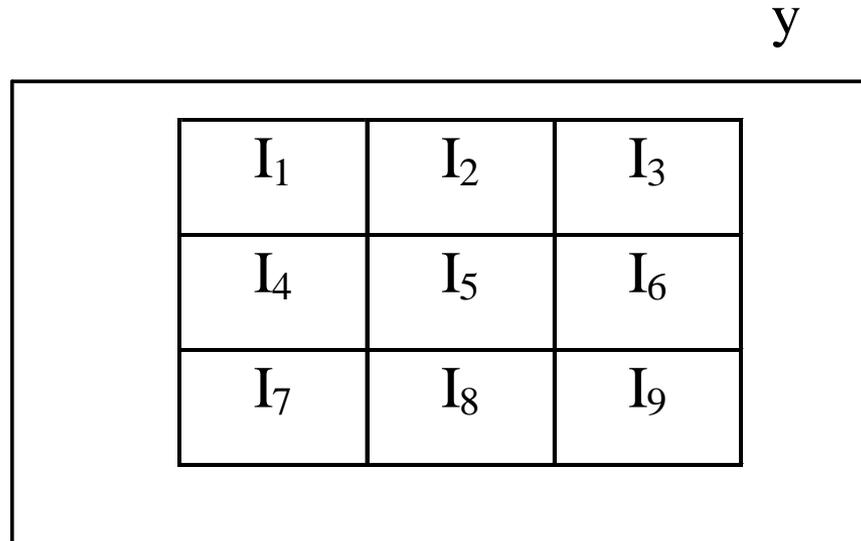
$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

$$\nabla f = \text{mag}(\nabla f) = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{1/2}$$

Filtros derivativos (3)

- Considerando la región de la imagen mostrada en la siguiente figura, donde I denota los valores de nivel de gris.
- La aproximación más simple es emplear la diferencia $(I_5 - I_8)$ en la dirección de x y $(I_5 - I_6)$ en la dirección de y combinados ambos: $\nabla f \approx [(I_5 - I_8)^2 + (I_5 - I_6)^2]^{1/2}$

Filtros derivativos (4)



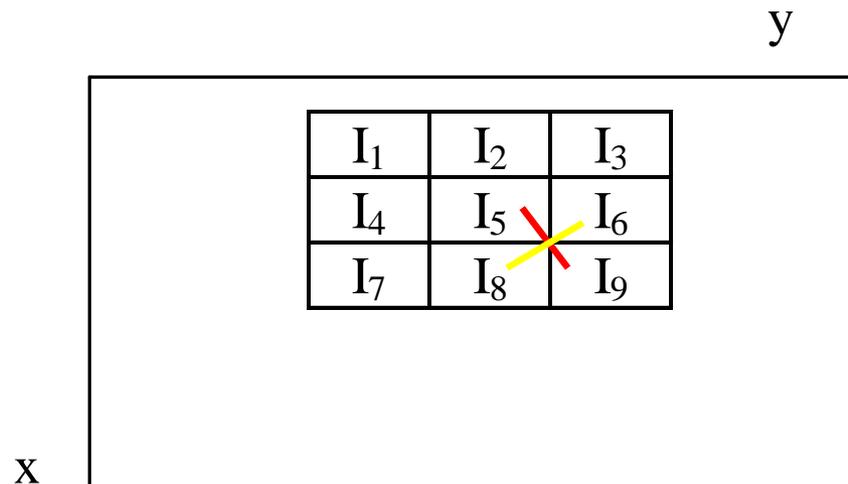
x

Filtros derivativos (5)

- En vez de utilizar los cuadrados o la raíz, podemos obtener resultados similares empleando los valores absolutos

$$\nabla f \approx |I_5 - I_8| + |I_5 - I_6|$$

- Otra aproximación es utilizar diferencias cruzadas



Filtros derivativos (6)

$$\nabla f \approx [(z_5 - z_9)^2 + (z_6 - z_8)^2]^{1/2}$$

- o sus valores absolutos

$$\nabla f \approx |z_5 - z_9| + |z_6 - z_8|$$

- las ecuaciones anteriores pueden implementarse empleando máscaras de tamaño 2x2, estas máscaras se llaman operadores de **Roberts**

Filtros derivativos (7)

1	0
0	-1

0	1
-1	0

Filtros derivativos (8)

- Otra aproximación es emplear máscaras de tamaño 3x3

$$\nabla f \approx |(I_7+I_8+I_9)-(I_1+I_2+I_3)| + |(I_3+I_6+I_9)-(I_1+I_4+I_7)|$$

- la diferencia entre la tercera y primera filas de la región 3x3 aproxima la derivada en la dirección x y la diferencia entre la tercera y la primera columna aproxima la derivada en la dirección y.

Filtros derivativos (9)

- Las máscaras que implementan esta nueva aproximación se denominan filtros de Prewitt:

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

- Otro tipo de filtro sería el de Sobel:

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

Filtro Laplaciano (1)

Este tipo de filtro se basa en un operador derivativo, por lo que acentúa las zonas que tienen gran discontinuidad en la imagen

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

Filtro Laplaciano (2)

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

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

Si la función depende de dos variables

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

Filtro Laplaciano (3)

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

La derivada de segundo orden con respecto a al variable x:

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

De forma similar, con respecto a y:

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

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

Filtro Laplaciano (4)

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

La anterior expresión es equivalente a aplicar una máscara definida por:

$$w = \begin{matrix} & 0 & 1 & 0 \\ & 1 & -4 & 1 \\ & 0 & 1 & 0 \end{matrix}$$

Filtro Laplaciano (5)

$f(x-1, y-1)$	$f(x-1, y)$	$f(x-1, y+1)$
$f(x, y-1)$	$f(x, y)$	$f(x, y+1)$
$f(x+1, y-1)$	$f(x+1, y)$	$f(x+1, y+1)$

Filtro Laplaciano (6)

```
% Imagen original
>> I=imread('moon.tif' );
% Se define una matriz de pesos
>> w=[0, 1, 0; 1, -4, 1; 0, 1, 0];
% Se aplica el filtro
>> I2=imfilter(I,w);
```

Alternativa:

```
>> I=imread('moon.tif');
>> w=fspecial('laplacian',0);
>> I2=imfilter(I,w,'replicate');
```



Imagen Original

1.0	-2.0	1.0
-2.0	5.0	-2.0
1.0	-2.0	1.0



Imagen Filtrada Paso Alto

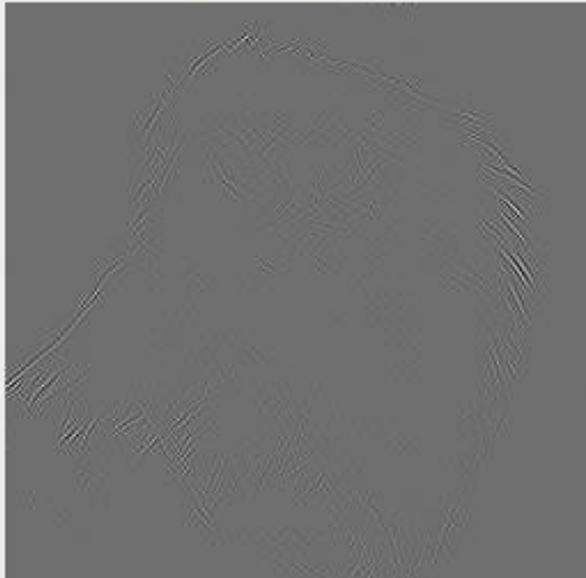


Imagen Realzada



Imagen Original

-1.0	-1.0	-1.0
-1.0	9.0	-1.0
-1.0	-1.0	-1.0



Imagen Filtrada Paso Alto



Imagen Realzada



Mascara: PrewittH ▼

Imagen: aguila ▼

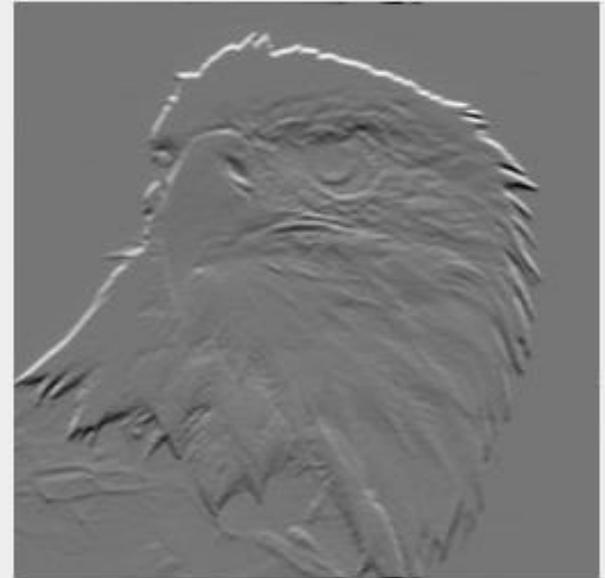
Aplicar

-1.0	-1.0	-1.0
0.0	0.0	0.0
1.0	1.0	1.0

Imagen Original



Imagen Filtrada



Mascara: **Laplaciano2** ▼

Imagen: **aguila** ▼

Aplicar

-1.0	-1.0	-1.0
-1.0	8.0	-1.0
-1.0	-1.0	-1.0

Imagen Original



Imagen Filtrada



Mascara: **Laplaciano3** ▼

Imagen: **aguila** ▼

Aplicar

0.0	-1.0	0.0
-1.0	4.0	-1.0
0.0	-1.0	0.0

Imagen Original



Imagen Filtrada

