

# **Filtros y convoluciones**

# Filtros y convoluciones.

- **Recordatorio:** en las transformaciones **globales**, cada píxel de salida depende sólo de un píxel de entrada.

90	67	75	78	<b>Transf. global</b>	62	68	78	81
92	87	78	82		102	89	76	85
45	83	80	130		83	109	80	111
39	69	115	154	<b>Transf. local</b>	69	92	115	120

**Entrada**  **Salida**

- No se tiene en cuenta la relación de **vecindad** entre píxeles. El resultado no varía si los píxeles son *permutados* aleatoriamente y después *reordenados*.
- **Transformación local:** el valor de un píxel depende de la vecindad local de ese píxel.

# Filtros y convoluciones.

- **Transformación global:**

$$R(x,y) := f(A(x,y)) \quad \text{ó} \quad R(x,y) := f(A(x,y), B(x,y))$$

- **Filtros y transformaciones locales:**

$$R(x,y) := f(A(x-k,y-k), \dots, A(x,y), \dots, A(x+k,y+k))$$

- **Ejemplo. Filtro de la media.**

$$R(x,y) := (A(x-1,y-1) + A(x,y-1) + A(x-1,y) + A(x,y))/4$$

92	78	82
45	80	130
39	115	154

**A**

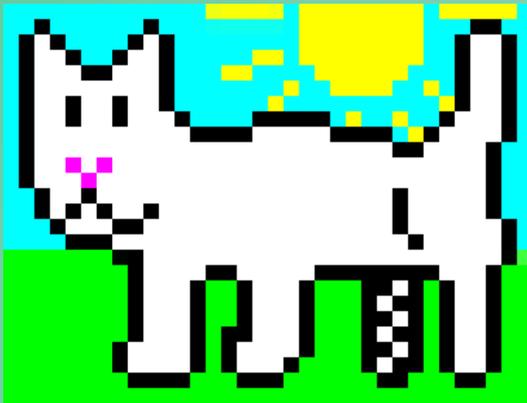
$\Sigma / 4$

-	-	-
-	74	93
-	70	120

**R**

# Filtros y convoluciones.

- **Ejemplo.** Entrada, A



Salida, R



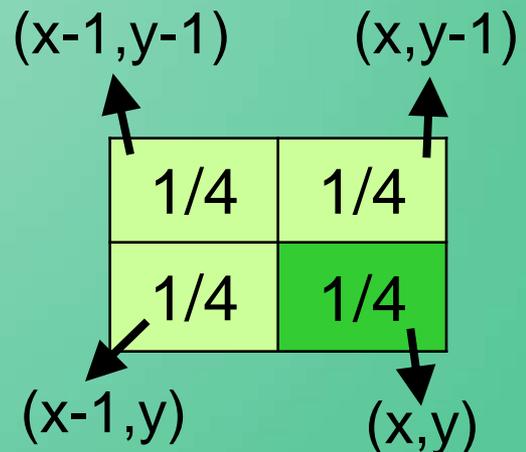
- **Resultado:** la imagen se *suaviza*, *difumina* o *emborrona*.
- Las transformaciones locales tienen sentido porque existe una relación de **vecindad** entre los píxeles.
- **Recordatorio:** un píxel representa una magnitud física en un punto de una escena → dos píxeles próximos corresponden a puntos cercanos de la escena → el mundo es “continuo” → los píxeles próximos tendrán valores parecidos.

# Filtros y convoluciones.

- Un tipo interesante de transformaciones locales son las convoluciones discretas.
- **Convolución discreta:** transformación en la que el valor del píxel resultante es una **combinación lineal** de los valores de los píxeles vecinos en la imagen.
- **Ejemplo.** El filtro de la media es una convolución.  
$$R(x,y) := 1/4 \cdot A(x-1,y-1) + 1/4 \cdot A(x,y-1) + 1/4 \cdot A(x-1,y) + 1/4 \cdot A(x,y)$$

- **Otra forma de ver la convolución:**

Matriz de coeficientes de la combinación lineal.



# Filtros y convoluciones.

- La matriz de coeficientes es conocida como la **máscara o núcleo (*kernel*) de convolución**.
- **Idea intuitiva:** se pasa la máscara para todo píxel de la imagen, aplicando los coeficientes según donde caigan.

Máscara de convolución

.1/4	.1/4
.1/4	.1/4

¿Cuánto valen estos píxeles?

Imagen de entrada, **A**

92	78	82
45	80	130
39	115	154

Imagen de salida, **R**

□	□	□
□	□	□
□	□	□

$\Sigma$

# Filtros y convoluciones.

- Sea **M** una máscara de convolución. Se puede definir como **array**  $[-k\dots k, -p\dots p]$  de real

En **X** la máscara va de  $-k$  a  $k$ , y en **Y** de  $-p$  a  $p$ . El punto central es  $(0,0)$

- **Algoritmo.** Cálculo de una convolución. Denotamos la convolución como:  $R := M \otimes A$

- **Entrada.**  $A$ : imagen de  $\max_x \times \max_y$   
 $M$ : array  $[-k\dots k, -p\dots p]$  de real

- **Salida.**  $R$ : imagen de  $\max_x \times \max_y$

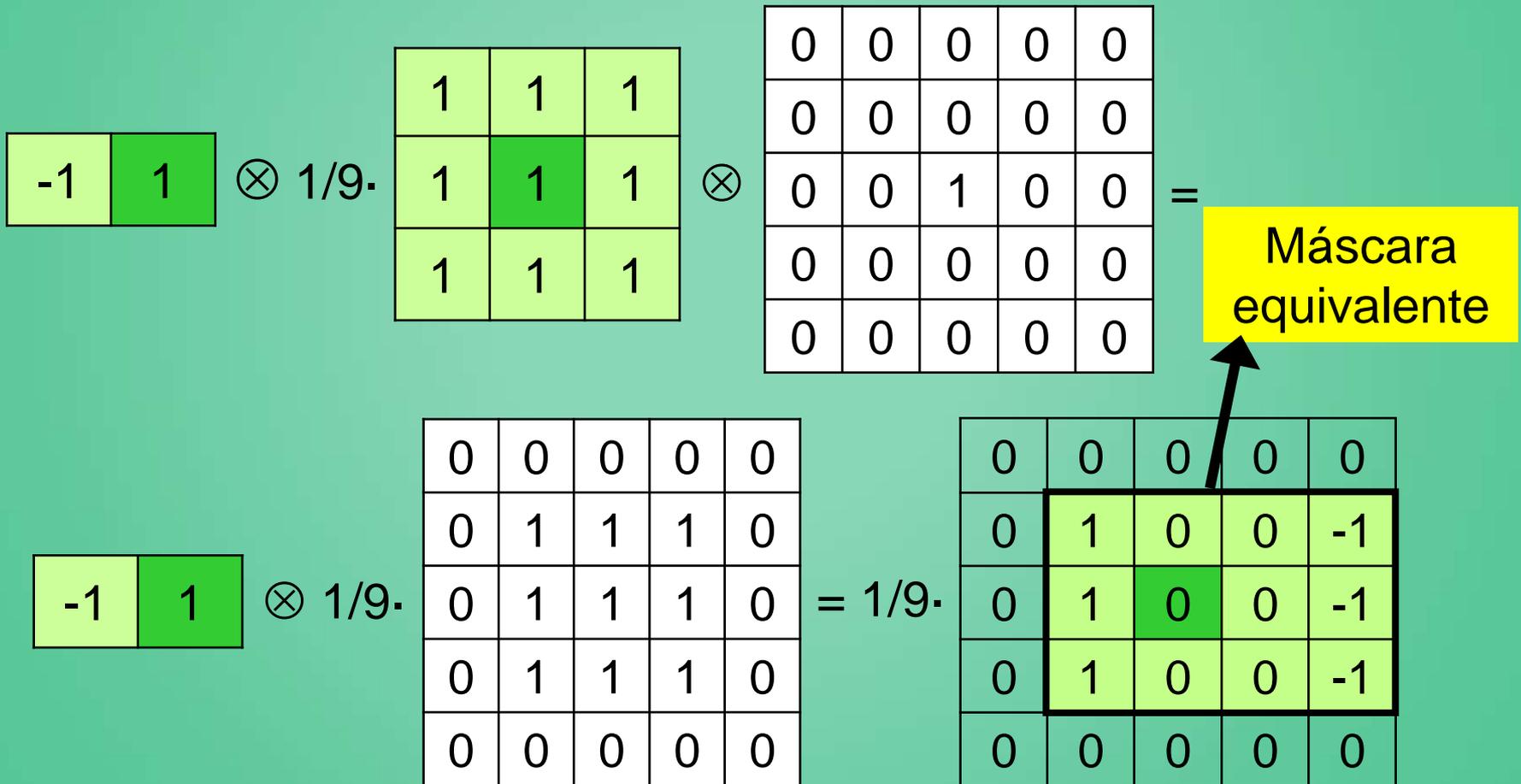
- **Algoritmo:**

**para cada** píxel  $(x, y)$  de la imagen  $A$  **hacer**

$$R(x, y) := \sum_{i=-k..k} \sum_{j=-p..p} M(i, j) \cdot A(x+i, y+j)$$

# Filtros y convoluciones.

- ¿Cómo calcular el resultado de la combinación?
- **Respuesta:** comprobar el efecto sobre una imagen sólo con el píxel central a UNO (“señal impulso”).



# Filtros y convoluciones.

- Análogamente, algunas convoluciones se pueden obtener combinando otras más simples: **núcleos separables**.
- **Ejemplo.**

$$\frac{1}{3} \cdot \begin{array}{|c|} \hline 1 \\ \hline 1 \\ \hline 1 \\ \hline \end{array} \otimes \frac{1}{3} \cdot \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline \end{array} \otimes A = \frac{1}{9} \cdot \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \otimes A$$

- **Resultado:** el filtro de la media es separable.
  - En lugar de aplicar una máscara de 3x3 se pueden aplicar dos máscaras de 1x3 y 3x1 (**máscaras unidimensionales**).
  - Puede ser útil para hacer los cálculos más **eficientes**.

# Filtros y convoluciones.

- ¿Qué hacer con los píxeles de los bordes?

.1/4	.1/4
.1/4	.1/4

⊗

9	4	8
7	8	4
3	2	2

- Posibilidades:**

- Asignar un 0 en el resultado a los píxeles donde no cabe la máscara.
- Suponer que los píxeles que se salen tienen valor 0 (u otra constante).
- Modificar la operación en los píxeles que no caben (variar el multiplicador).
- Suponer que la imagen se extiende por los extremos (p.ej. como un espejo).

0	0	0
0	7	6
0	5	4

2	3	3
4	7	6
2	5	4

9	6	6
8	7	6
5	5	4

5	4	4
7	7	6
8	5	4

# Filtros y convoluciones.

- Las convoluciones son una discretización de la idea de convolución usada en señales. (Repasar teoría de señales...)
- **Diferencias:** las convoluciones usadas aquí son discretas y bidimensionales.
- **Idea:** las máscaras de convolución son matrices de números → se pueden considerar, a su vez, como imágenes.
- **Propiedades:**
  - **Asociativa:**  $M2 \otimes (M1 \otimes A) = (M2 \otimes M1) \otimes A$
  - **Conmutativa:**  $M2 \otimes M1 \otimes A = M1 \otimes M2 \otimes A$
  - **Ojo:** al aplicar una convolución puede ocurrir **saturación** de píxeles. Si ocurre esto, el orden sí que puede ser importante.

# Suavizado, perfilado y bordes.

- Aplicando distintos operadores de convolución es posible obtener **diferentes efectos**:
  - **Suavizado**: o difuminación de la imagen, reducir contrastes abruptos en la imagen.
  - **Perfilado**: resaltar los contrastes, lo contrario al suavizado.
  - **Bordes**: detectar zonas de variación en la imagen.
  - **Detección** de cierto tipo de características, como esquinas, segmentos, etc.
- Suavizado y perfilado son más habituales en **restauración y mejora** de imágenes.
- Bordes y detección de características suelen usarse más en **análisis de imágenes**.

# Operadores de suavizado.

- El operador de suavizado más simple es la **convolución de media** (media aritmética).
- **Parámetros** del operador:
  - Ancho y alto de la región en la que se aplica:  $w \times h$ .
  - Posición del ancla.
- Normalmente,  $w$  y  $h$  son impares y el ancla es el píxel central.
- La máscara es un simple array de unos de tamaño  $w \times h$ .

1	1	1
1	1	1
1	1	1

Máscara de media de 3x3

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Media de 5x5

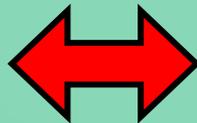
# Operadores de suavizado.

- **Ventajas** (respecto a otros suavizados):
  - Sencillo y rápido de aplicar.
  - Fácil definir un comportamiento para los **píxeles de los bordes**: tomar la media de los píxeles que quepan.
  - Recordatorio: el operador de media es **separable**.

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

**Media de 5x5**

Total: 25 sumas  $\rightarrow o(n^2)$



1	1	1	1	1
---	---	---	---	---

⊗

1
1
1
1
1

**Media de 5x1 y de 1x5**

Total: 10 sumas  $\rightarrow o(2n)$

# Operadores de suavizado.

- En algunos casos puede ser interesante aplicar **suavizados direccionales**: horizontales, verticales o en cualquier dirección.

1	1	1	1	1
---	---	---	---	---

Media horizontal 5 píxeles

1
1
1

Media vertical 3p

0	0	1
0	1	0
1	0	0

Media diagonal 3p



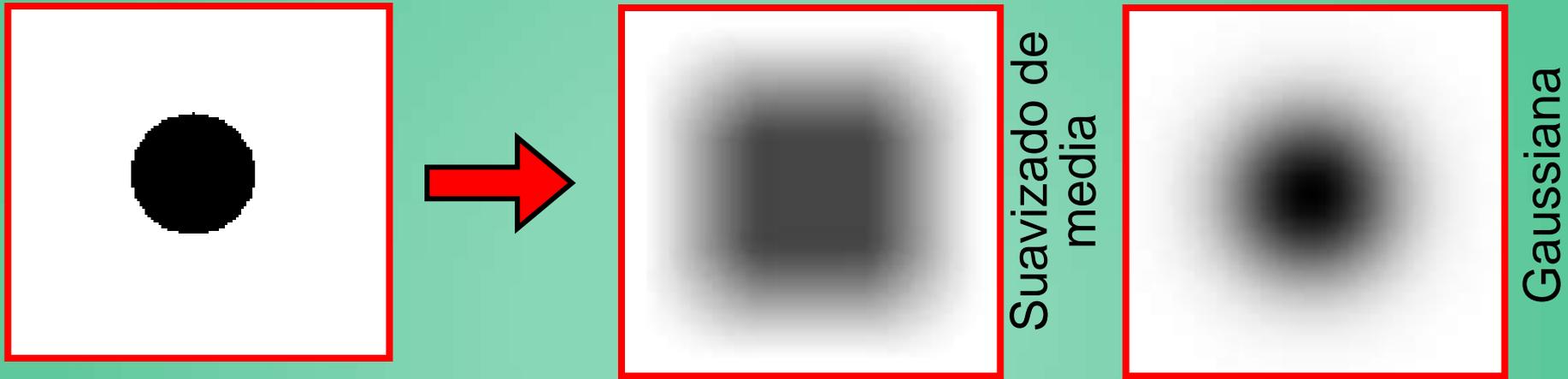
Media horiz. 31p



Media vert. 31p

# Operadores de suavizado.

- Cuando se aplica la media con tamaños grandes se obtienen resultados **artificiosos** (a menudo **indeseados**).



- **Motivo:** la media se calcula en una región cuadrada.
- Sería mejor aplicarla a una **región “redonda”**.
- O, mejor, usar suavizado gaussiano...

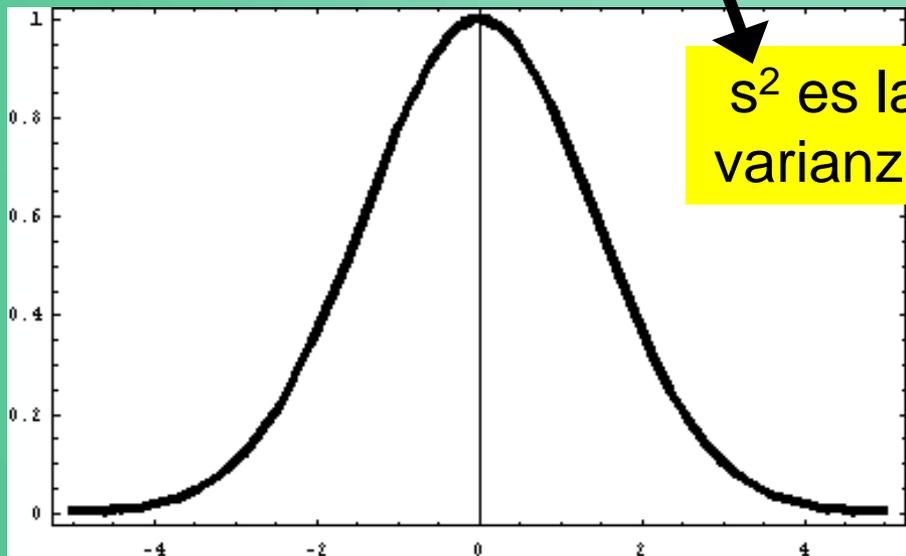
0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1	1	0	0

# Operadores de suavizado.

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

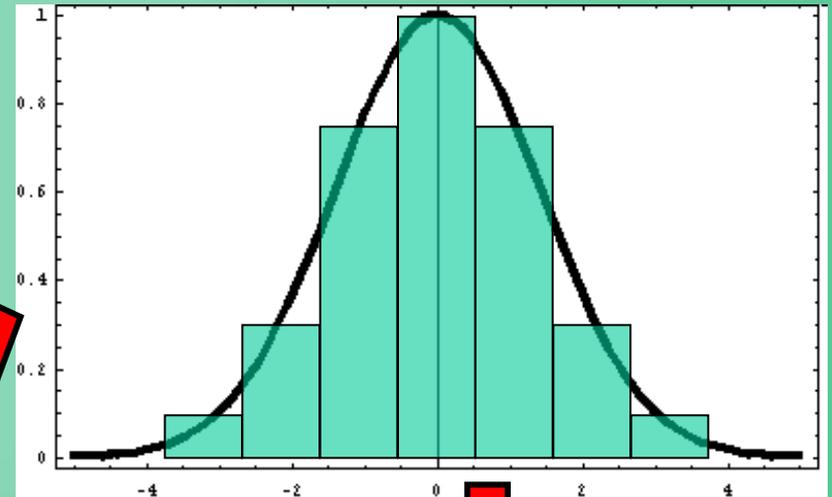
## Campana de Gauss

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



$s^2$  es la  
varianza

## Campana discreta



1/64.



# Operadores de suavizado.

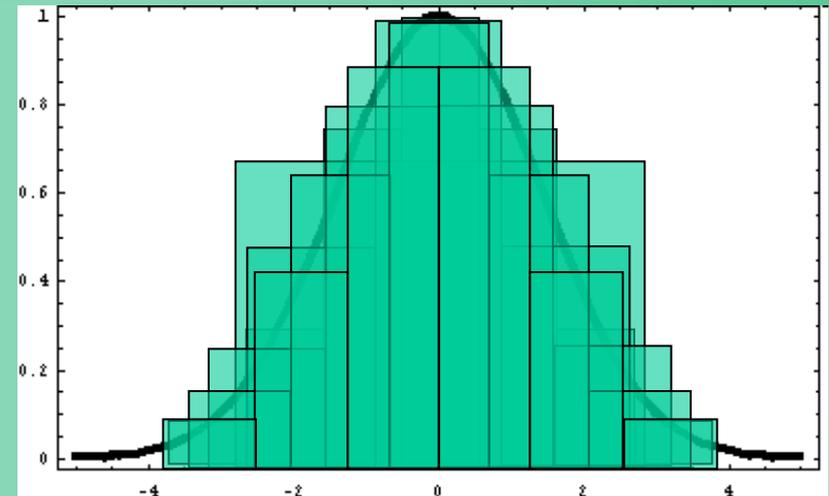
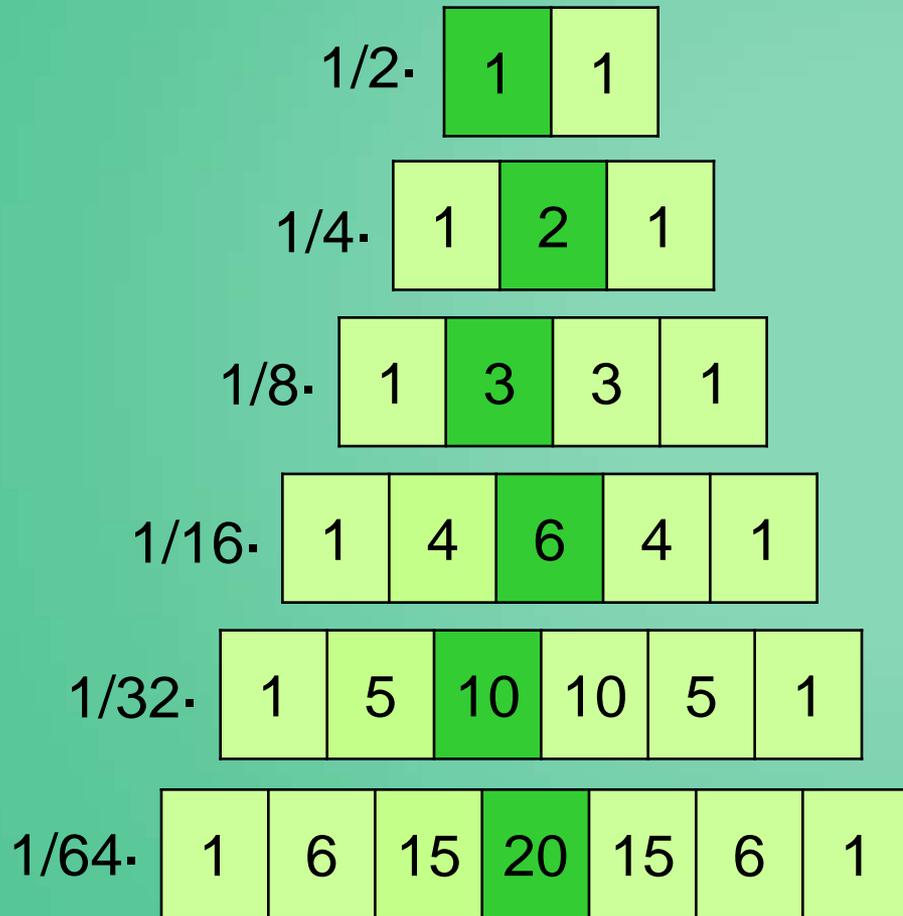
- La **varianza**,  $s^2$ , indica el nivel de suavizado.
  - **Varianza grande**: campana más ancha, más suavizado.
  - **Varianza pequeña**: campana más estrecha, menos suavizado.
  - Se mide en píxeles.
- **Cálculo de la máscara gaussiana (1D)**: calcular la función, discretizar en el rango, discretizar en el valor y calcular el multiplicador...

					1
				1	1
			1	2	1
		1	3	3	1
	1	4	6	4	1
1	5	10	10	5	1

- ¿No existe una forma más rápida?
- **Idea**: el triángulo de Pascal.

# Operadores de suavizado.

- ¡Magia! Las filas del triángulo de Pascal forman discretizaciones de la campana de Gauss.



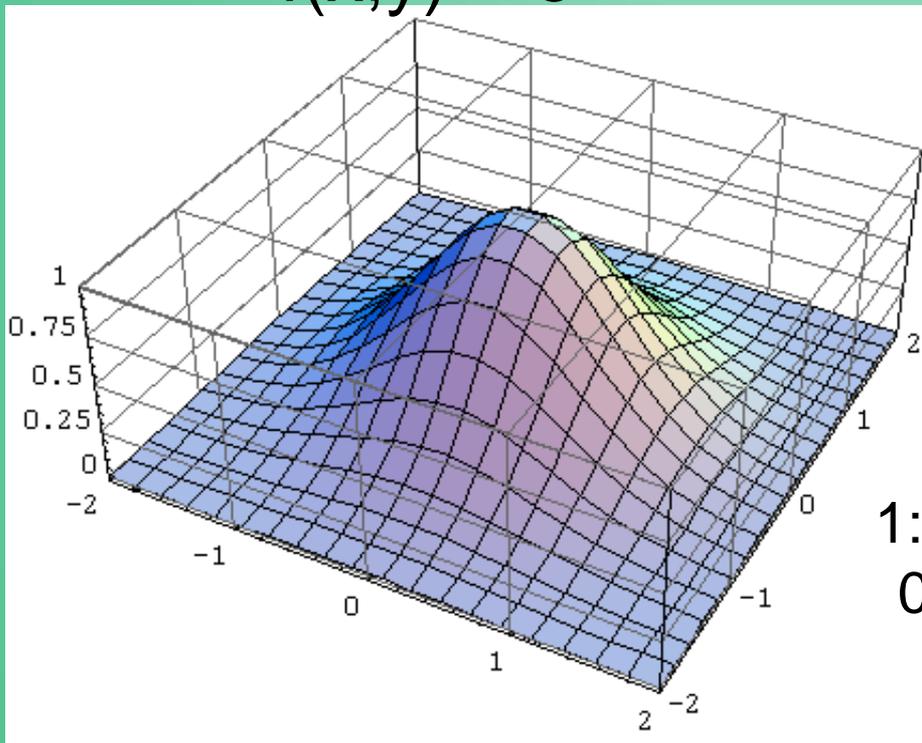
¿Por qué ocurre así?  
Recordar el **teorema central del límite...**

# 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

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



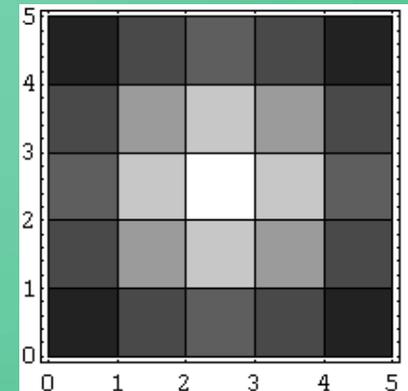
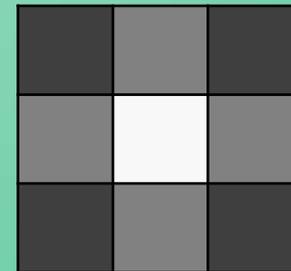
Máscara gaussiana de 3x3

1	2	1
2	4	2
1	2	1

1/16.

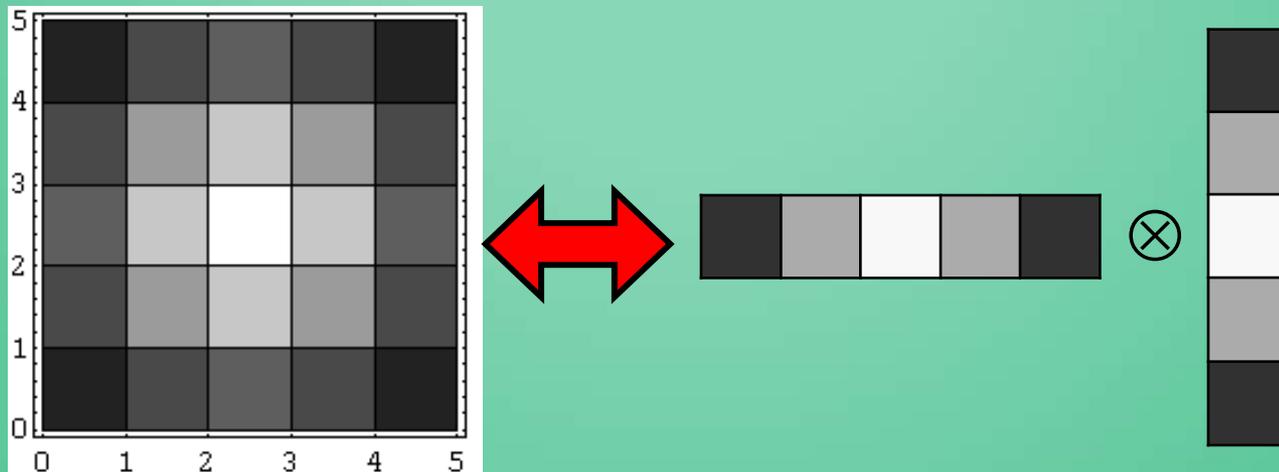
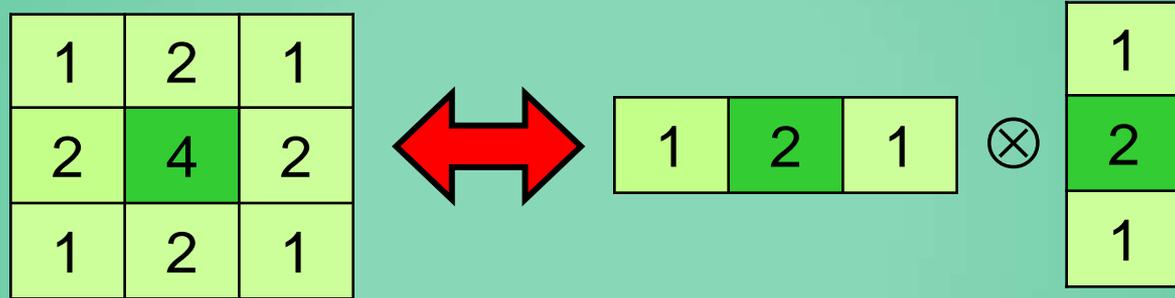


1: blanco  
0: negro



# Operadores de suavizado.

- **Propiedad interesante:** el filtro gaussiano es separable.
- **Resultado:** se puede obtener un suavizado 2D aplicando dos máscaras gaussianas bidimensionales, una horizontal y otra vertical.



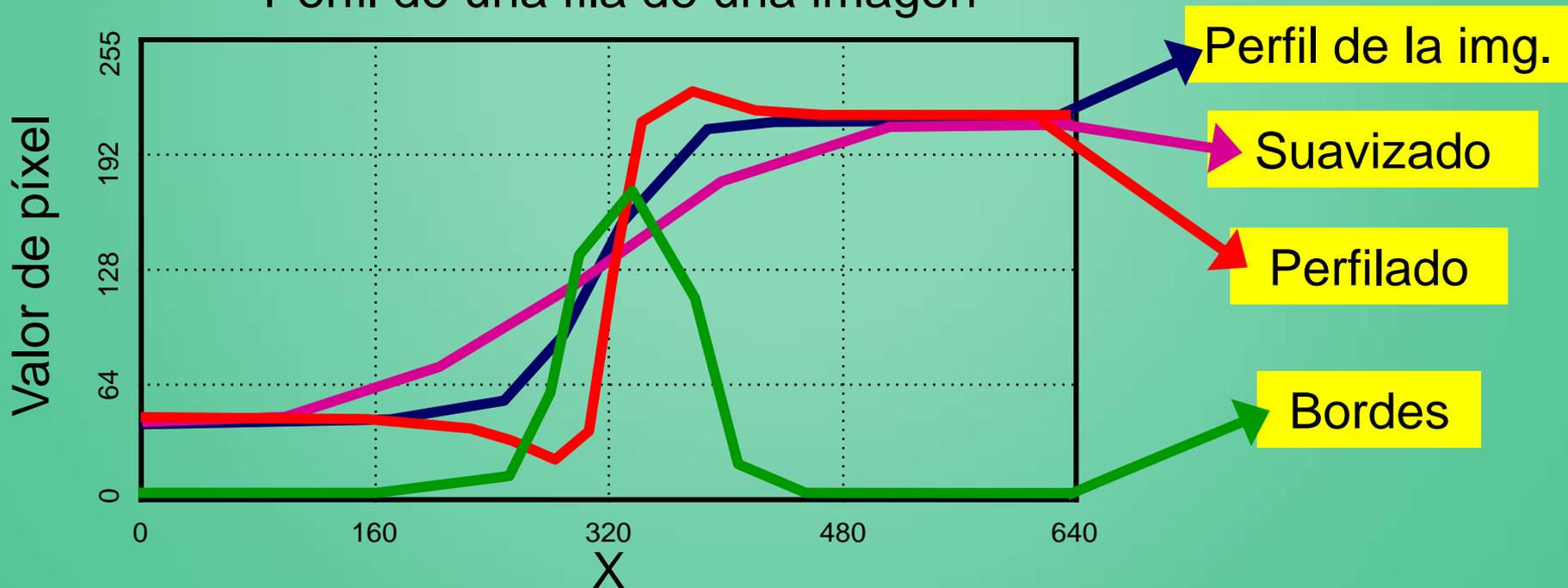
# Operadores de suavizado.

- **Resultados** de la comparación:
  - Para conseguir un mismo “**grado de suavizado**” la máscara gaussiana debe ser de mayor tamaño.
    - Se puede tomar como medida la **varianza** de la máscara correspondiente.
  - El efecto del suavizado gaussiano es más **natural** (más similar a un desenfoque) que la media.
    - Suele ser más habitual en procesamiento y análisis de imágenes.
  - Ambos filtros son **separables**.
    - Si la máscara es de  $n \times n$ , pasamos de  $O(n^2)$  a  $O(2n)$ .

# Operadores de bordes.

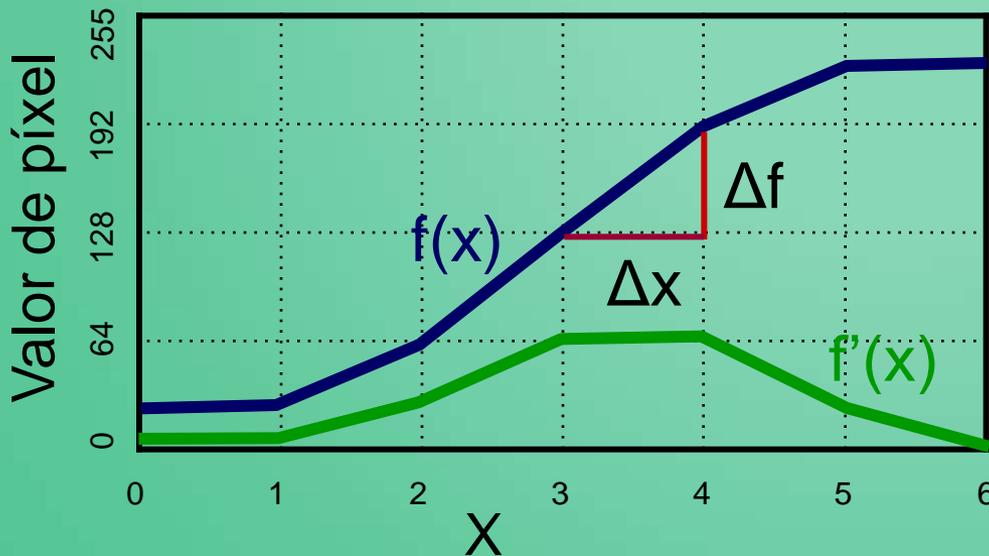
- **Perfilado y detección de bordes** están relacionados con el suavizado:
  - **Suavizado**: reducir las variaciones en la imagen.
  - **Perfilado**: aumentar las variaciones en la imagen.
  - **Bordes**: encontrar las zonas de variación.

Perfil de una fila de una imagen



# Operadores de bordes.

- Matemáticamente, la variación de una función  $f(x)$  cualquiera viene dada por la **derivada** de esa función:
  - $f'(x) > 0$  : función creciente en  $X$
  - $f'(x) < 0$  : función decreciente en  $X$
  - $f'(x) = 0$  : función uniforme en  $X$
- En nuestro caso, tenemos **funciones discretas**. La “**derivada discreta**” se obtiene calculando diferencias.



$$f'(x) = \Delta f / \Delta x$$

$$\underbrace{\Delta f = f(x) - f(x-1) \quad \Delta x = 1}_{f'(x) = f(x) - f(x-1)}$$

$$f'(x) = f(x) - f(x-1)$$

- **Conclusión:** la derivada se calculará con máscaras del tipo:

-1	1
----	---

# Operadores de bordes.

Máscara de derivada en X (M):

-1	1
----	---

Derivada en Y:

-1
1

Derivadas en diagonales:

-1	0	0	-1
0	1	1	0

- Ejemplo. Derivada en X.  $R := M \otimes A$



Imagen de entrada

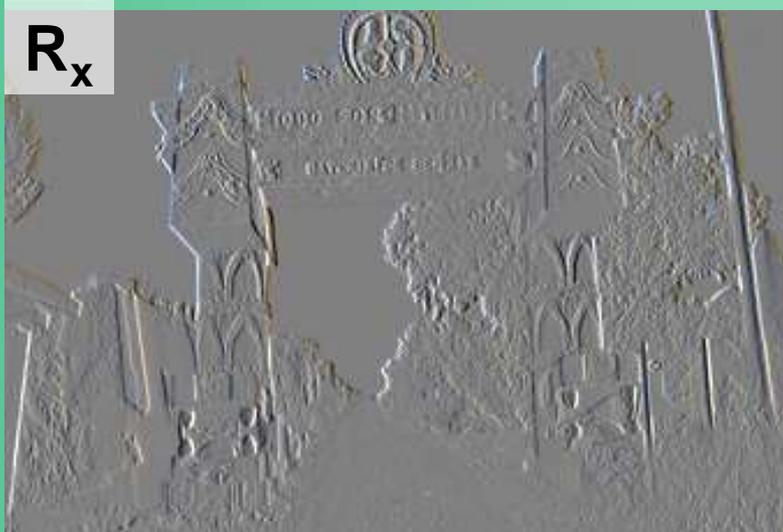


Derivada en X (x2)

$[0..255] - [0..255] = [-255..255]$

# Operadores de bordes.

- Los bordes decrecientes se saturan a 0...
- Podemos sumar 128 para apreciar mejor el resultado:
  - Gris (128): diferencia 0
  - Negro: decreciente
  - Blanco: creciente



Derivada X (+128)



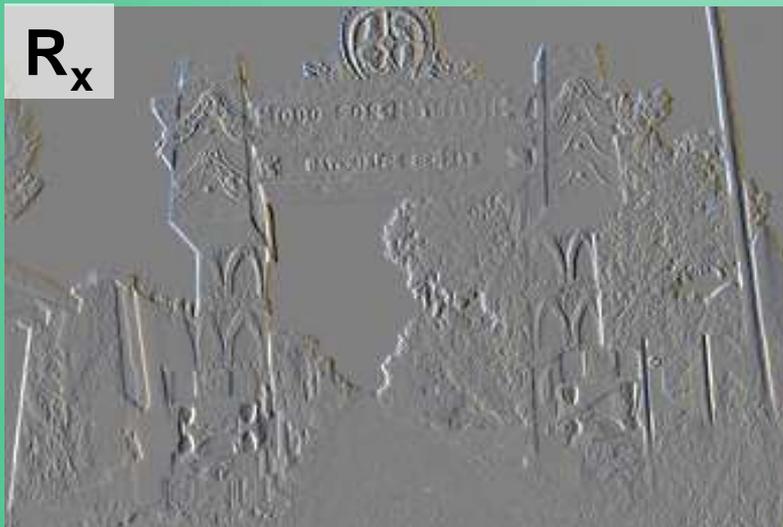
Derivada Y (+128)

- Se produce una especie de “**bajorrelieve**” (*emboss*), que puede usarse en efectos especiales.

# Operadores de bordes.

- Los operadores de bordes son muy **sensibles al ruido**.
- Es posible (y adecuado) **combinar** los operadores de **bordes con suavizados**.

-1	1	$\otimes$	1	2	1	=	1	1	-1	-1
			2	4	2		2	2	-2	-2
			1	2	1		1	1	-1	-1



Derivada X (+128)



Suaviz. + Deriv. X

# Operadores de bordes.

- Existen algunos **operadores** de bordes **estándar**.
- **Filtros de Prewitt:**

Filtro de Prewitt 3x3, derivada en X

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

Filtro de Prewitt 3x3, derivada en Y

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

- **Filtros de Scharr:**

Filtro de Scharr 3x3, derivada en X

-3	0	3
-10	0	10
-3	0	3

Filtro de Scharr 3x3, derivada en Y

-3	-10	-3
0	0	0
3	10	3

# Operadores de bordes.

- **Filtros de Sobel:** se construyen usando la derivada de la gaussiana.

Filtro de Sobel 3x3, derivada en X

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

Filtro de Sobel 3x3, derivada en Y

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

- Además, el filtro de Sobel permite calcular derivadas conjuntas en X e Y, derivadas segundas, terceras, etc.
- **Ejemplo.** Derivada segunda en X.

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

# Operadores de bordes.

- Ejemplos.



Imagen de entrada



Prewitt Y (3x3)



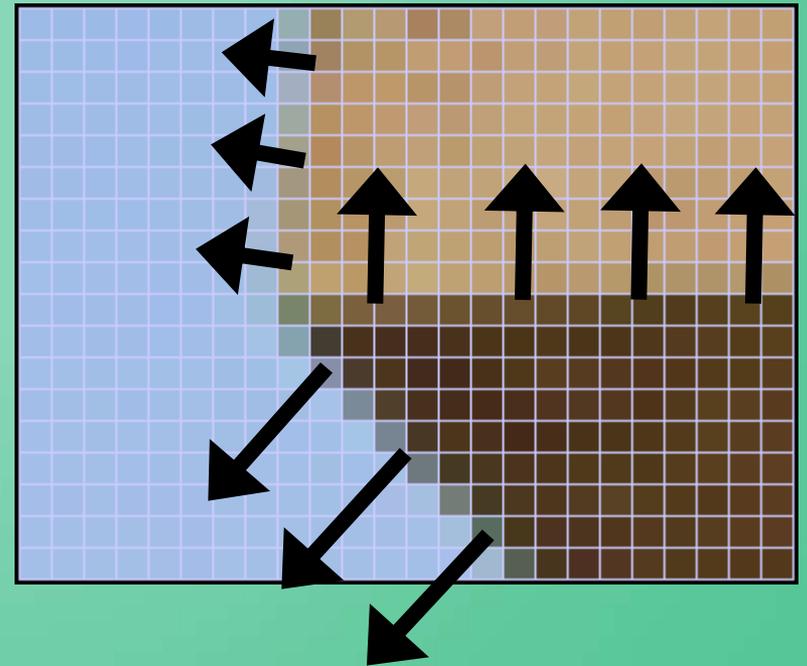
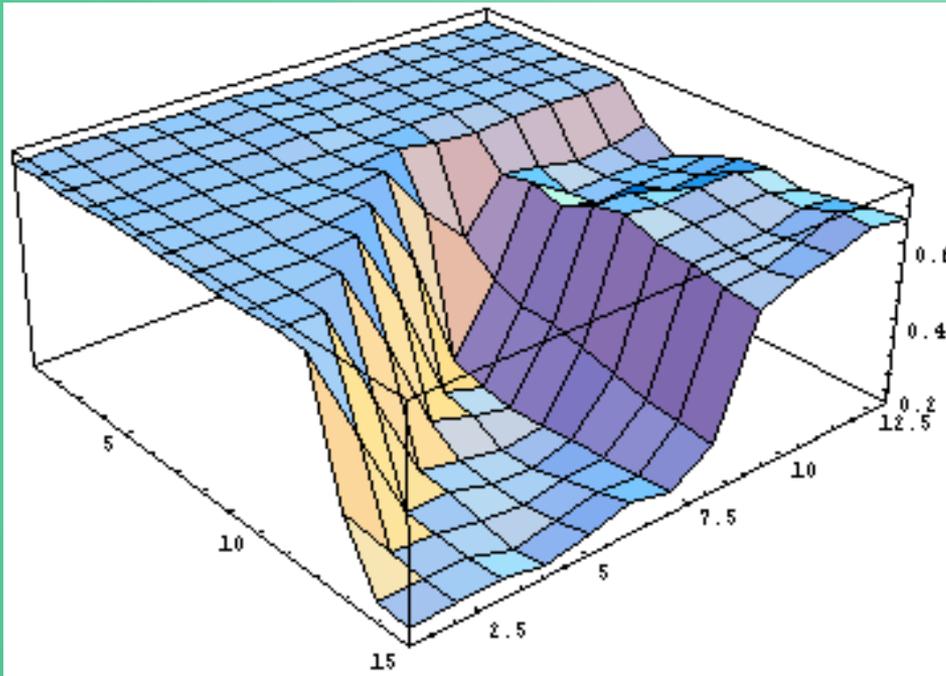
Sobel Y (3x3)



Sobel 2ª deriv. Y

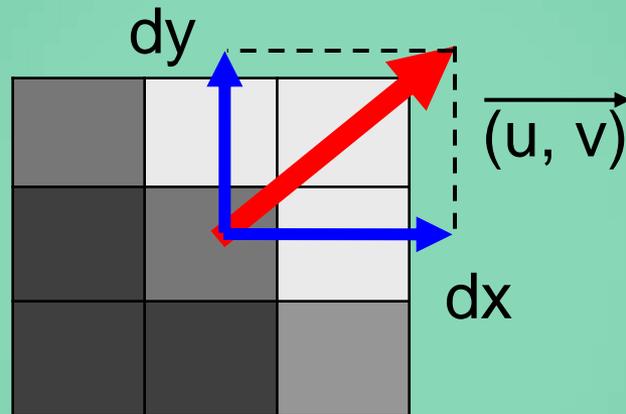
# Operadores de bordes.

- Realmente, en dos o más dimensiones, en lugar de la derivada tiene más sentido el concepto de **gradiente**.
- ¿Qué es el gradiente? → Repasar cálculo...
- **El gradiente** indica la dirección de máxima variación de una función (en 2D, la máxima pendiente).



# Operadores de bordes.

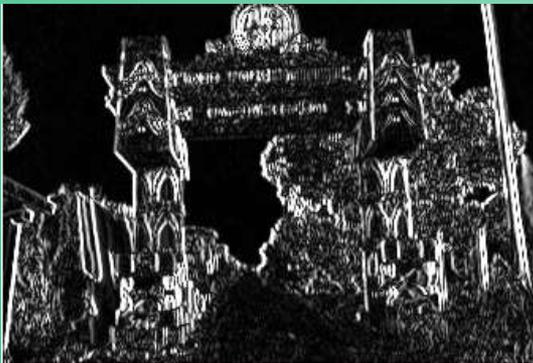
- El **gradiente** en un punto es un vector  $\overrightarrow{(u, v)}$ :
  - **Ángulo**: dirección de máxima variación.
  - **Magnitud**: intensidad de la variación.



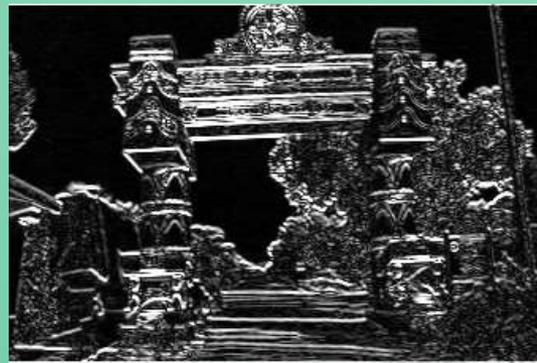
- El **gradiente** está relacionado con las **derivadas**:
  - $u$  = Derivada en X del punto
  - $v$  = Derivada en Y del punto
  - Teniendo  $dy$  y  $dx$ , ¿cuánto vale el ángulo y la magnitud?

# Operadores de bordes.

- **Cálculo del gradiente:**
  - Calcular **derivada en X: Dx** (por ejemplo, con un filtro de Sobel, Prewitt,...)
  - Calcular **derivada en Y: Dy**
  - **Magnitud** del gradiente:  $\sqrt{Dx^2 + Dy^2}$
  - **Ángulo** del gradiente:  $\text{atan2}(Dy, Dx)$



Valor absoluto de derivada en X  
(Sobel de 3x3)



Valor absoluto de derivada en Y  
(Sobel de 3x3)



Magnitud del gradiente

# Operadores de bordes.

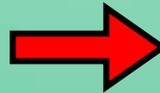
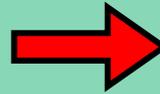
- El gradiente da lugar al concepto de **borde**.
- Un **borde** en una imagen es una curva a lo largo de la cual el gradiente es máximo.



El borde es perpendicular a la dirección del gradiente.

# Operadores de bordes.

- Los bordes de una escena son **invariantes** a cambios de luminosidad, color de la fuente de luz, etc. → En **análisis de imágenes** usar los bordes (en lugar de las originales).



# Operadores de bordes.

- **Otras formas de calcular los bordes:**

1. Calcular la derivada en diferentes direcciones:  $D_1$ ,  $D_2$ ,  $D_3$ ,  $D_4$ .
2. Para cada punto, la magnitud del gradiente es la derivada de máximo valor absoluto:

$$G(x,y) := \max \{ |D_1(x,y)|, |D_2(x,y)|, |D_3(x,y)|, |D_4(x,y)| \}$$

3. La dirección del gradiente viene dada por el ángulo que ha producido el máximo:

$$A(x,y) := \operatorname{argmax} \{ |D_1(x,y)|, |D_2(x,y)|, |D_3(x,y)|, |D_4(x,y)| \}$$

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

$D_1$ : N-S

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

$D_2$ : NE-SO

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

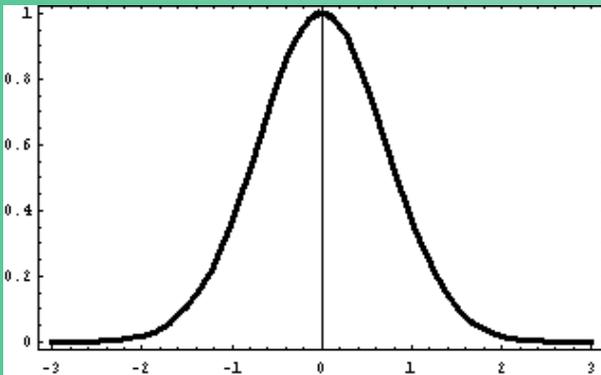
$D_3$ : E-O

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

$D_4$ : SE-NO

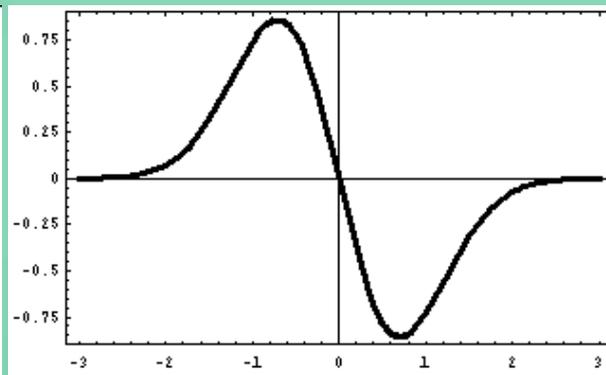
# Operadores de bordes.

- Otra forma más sencilla (aproximada) es usar máscaras de convolución adecuadas, por ejemplo de **Laplace**.
- La **función de Laplace** es la segunda derivada de la gaussiana.



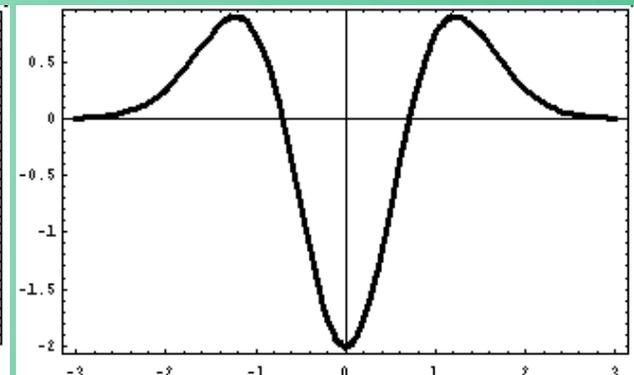
$$f(x) = e^{-x^2/s^2}$$

**Másc. Gaussiana**  
Operador de suavizado



$$df(x)/dx$$

**Másc. Sobel**  
Operador de derivación



$$d^2f(x)/dx^2$$

**Másc. Laplaciana**  
Operador de gradiente

# Operadores de bordes.

- La máscara laplaciana se define usando la función de Laplace.
- Ejemplos de máscaras de Laplace.

0	1	0
1	-4	1
0	1	0

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

“Diferencia entre el píxel central y la media de sus vecinos...”

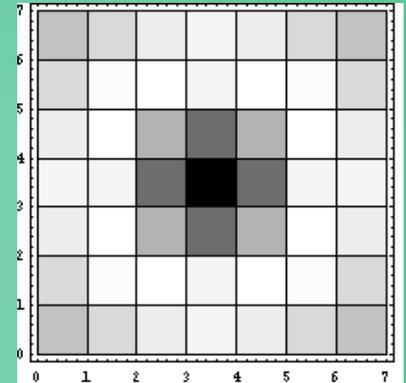


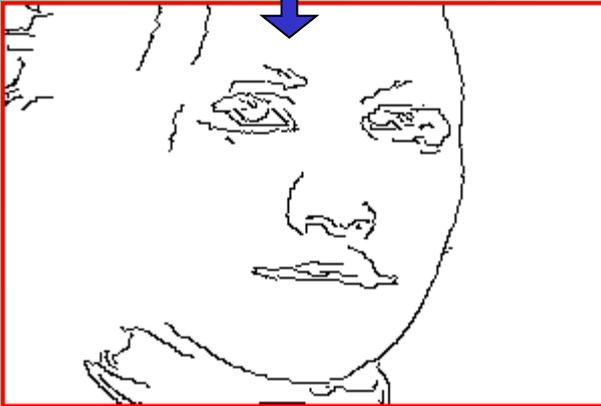
Imagen de entrada



Laplaciana 2 (3x3)

# Operadores de bordes.

- **Detector de bordes de Canny:**
  - No sólo usa convoluciones (operadores de gradiente), sino que busca el **máximo gradiente** a lo largo de un borde.
  - El resultado es una **imagen binaria** (borde/no borde), ajustable mediante un umbral.



# Operadores de perfilado.

- **Perfilado:** destacar y hacer más visibles las variaciones y bordes de la imagen. Es lo contrario al suavizado.
- Permite eliminar la apariencia borrosa de las imágenes, debida a imperfecciones en las lentes.
- ... aunque tampoco se pueden hacer milagros...



← Suavizado

Original

Perfilado →

# Operadores de perfilado.

- El perfilado se puede conseguir sumando a la imagen **original**, la **laplaciana** ponderada por cierto factor.
- Lo cual equivale a usar una máscara de convolución adecuada:

	Laplaciana		Identidad		Perfilado																											
1.	<table border="1"><tr><td>-1</td><td>-1</td><td>-1</td></tr><tr><td>-1</td><td>8</td><td>-1</td></tr><tr><td>-1</td><td>-1</td><td>-1</td></tr></table>	-1	-1	-1	-1	8	-1	-1	-1	-1	+	<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	1	0	0	0	0	=	<table border="1"><tr><td>-1</td><td>-1</td><td>-1</td></tr><tr><td>-1</td><td>9</td><td>-1</td></tr><tr><td>-1</td><td>-1</td><td>-1</td></tr></table>	-1	-1	-1	-1	9	-1	-1	-1	-1
-1	-1	-1																														
-1	8	-1																														
-1	-1	-1																														
0	0	0																														
0	1	0																														
0	0	0																														
-1	-1	-1																														
-1	9	-1																														
-1	-1	-1																														

- Más o menos perfilado dando distintos pesos, **a**.

	<table border="1"><tr><td>0</td><td>-1</td><td>0</td></tr><tr><td>-1</td><td>4</td><td>-1</td></tr><tr><td>0</td><td>-1</td><td>0</td></tr></table>	0	-1	0	-1	4	-1	0	-1	0	+	<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	1	0	0	0	0	=	<table border="1"><tr><td>0</td><td>-a</td><td>0</td></tr><tr><td>-a</td><td>4a+1</td><td>-a</td></tr><tr><td>0</td><td>-a</td><td>0</td></tr></table>	0	-a	0	-a	4a+1	-a	0	-a	0
0	-1	0																														
-1	4	-1																														
0	-1	0																														
0	0	0																														
0	1	0																														
0	0	0																														
0	-a	0																														
-a	4a+1	-a																														
0	-a	0																														

**Ojo:** la función Laplacian usa máscaras “invertidas”, luego **a** debe ser  $< 0$

# Operadores de perfilado.

- Ejemplos. Variando pesos y tamaño de la laplaciana.



Imagen de entrada



Perfilado 33%, 3x3



Perfilado 60%, 1x1



Perfilado 15%, 7x7

# Operadores de perfilado.

- **Cuidado con el perfilado.** La operación de perfilado aumenta el nivel de ruido de la imagen.



Imagen con ruido  
por interferencias  
TV



Perfilado 33%, 3x3



Imagen con ruido  
por compresión  
JPEG



Perfilado 60%, 3x3

# Suavizado, perfilado y bordes.

## Conclusiones:

- Las **convoluciones** son una herramienta fundamental en procesamiento de imágenes.
  - **Una misma base común:** combinaciones lineales de una vecindad local de los píxeles (de cierto tamaño).
  - **Diversos usos:** según los valores de los coeficientes: suavizado, eliminación de ruido, bordes, perfilado, etc.
- Se pueden definir **operaciones similares** sobre **vídeo** (usando la dimensión temporal, por ejemplo, suavizado a lo largo del tiempo), y sobre **audio digital** (por ejemplo, suavizado de la señal o introducción de eco).
- Es importante conocer el **significado matemático** de los procesos aplicados (derivadas, gradientes, integrales,...).