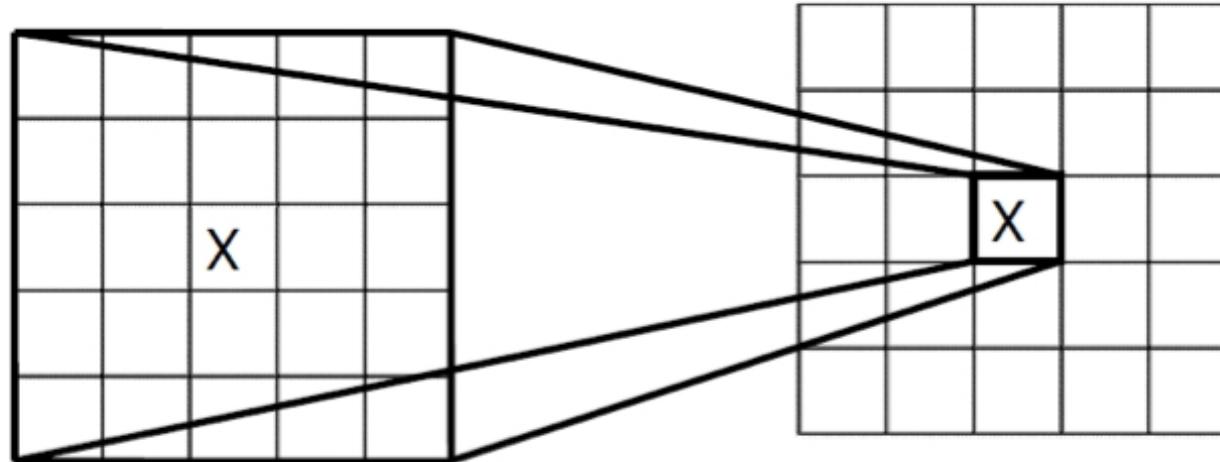


Vecindad Local: Filtros



- Referencias
- R. C. Gonzalez and R. E. Woods. Digital Image Processing. Addison-Wesley Longman Publishing Co., Inc., 2001.
- R. Szeliski, Computer Vision: Algorithms and Applications Microsoft Research, 2010. Draft online: <http://szeliski.org/Book/>



PROCESADO LOCAL

Los métodos de preprocesado con vecindad local ***combinan el nivel de gris del píxel con los niveles de gris de sus vecinos*** en la imagen original para obtener el nivel del gris del píxel en la imagen transformada.

La operación se le suele denominar **Filtrado**



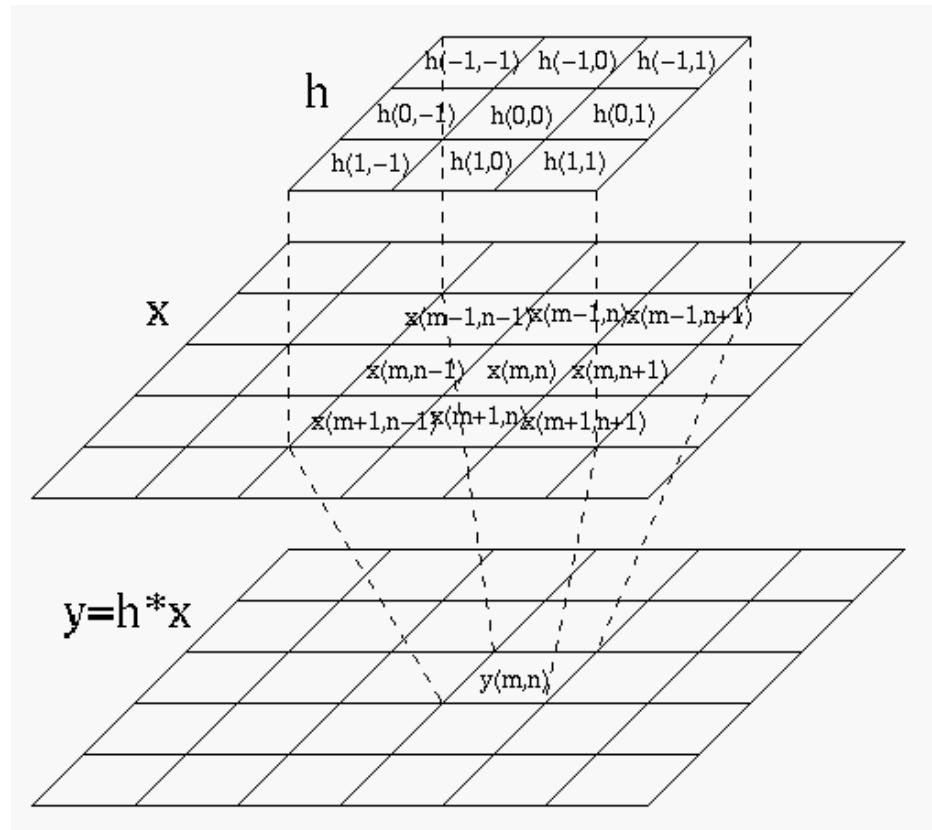
Filtros espaciales

- Elementos básicos
 - Un vecindario de tamaño $m \times n$ en la imagen centrado en el pixel (i,j)
 - Una subimagen de tamaño $m \times n$ denominada máscara, filtro o kernel
 - Los valores de la máscara se denominan coeficientes



Filtros espaciales: algoritmo

- Para cada columna i de la imagen
 - Para cada fila j de la imagen
 - La máscara del filtro se situa sobre la posición (i,j)
 - La salida del filtro se calcula según una relación predefinida entre la máscara del filtro y los valores del vecindario de la imagen





Filtros espaciales: algoritmo

45	60	98	127	132	133	137	133
46	65	98	123	126	128	131	133
47	65	96	115	119	123	135	137
47	63	91	107	113	122	138	134
50	59	80	97	110	123	133	134
49	53	68	83	97	113	128	133
50	50	58	70	84	102	116	126
50	50	52	58	69	86	101	120

$f(x,y)$

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

*

=

69	95	116	125	129	132
68	92	110	120	126	132
66	86	104	114	124	132
62	78	94	108	120	129
57	69	83	98	112	124
53	60	71	85	100	114

$h(x,y)$

$g(x,y)$



Transformada de Fourier

Las representaciones en el *dominio de la frecuencia*, en base a explicar cómo se repiten los píxeles de una imagen, consiguen representar la información de tal imagen.

Esta representación es especialmente útil, ya que teniendo la frecuencia de repetición de los elementos que componen una imagen, se pueden apreciar y alterar directamente elementos como el ruido, los bordes, las texturas, etc.

Transformadas en el dominio de la frecuencia usadas en tratamiento de imágenes:

- Transformada de Fourier (funciones base: senos y cosenos)
- Transformada del coseno (funciones base: cosenos)
- Transformadas wavelet (funciones base: Haar, Daubechies,...)



Transformada de Fourier

- Una *serie de Fourier* puede considerarse como la suma de un conjunto de funciones sinusoidales de diferentes frecuencias, promediada por unos coeficientes, con el objetivo de aproximarse a una función $f(x)$.
- Una función periódica en el tiempo, de periodo T_0 ($T_0=2\pi/\omega_0$), puede expresarse como:
$$f(x) = a_0 + \sum_{k=1}^{\infty} (a_k \cos k\omega_0 x + b_k \sin k\omega_0 x)$$
- Estos coeficientes evalúan qué peso tiene cada una de las funciones sinusoidales a la hora de construir la función $f(x)$.
- El conjunto de señales sinusoidales (senos y cosenos), constituye una base de funciones en el dominio de la frecuencia.
- La *transformada de Fourier* de una función $f(x)$ es una extensión de las series de Fourier a señales no periódicas.

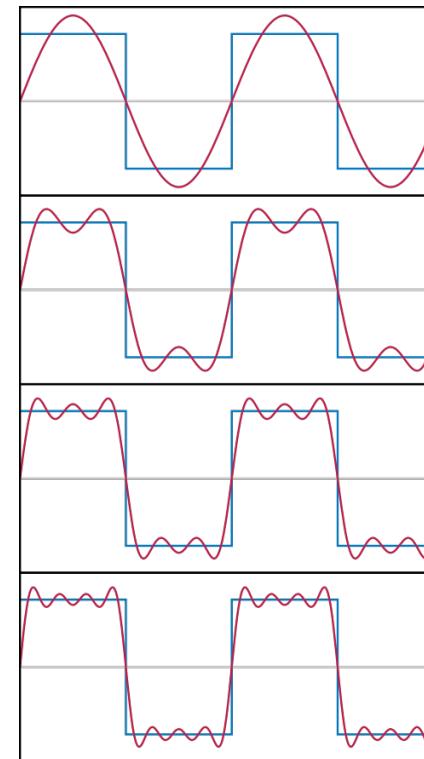
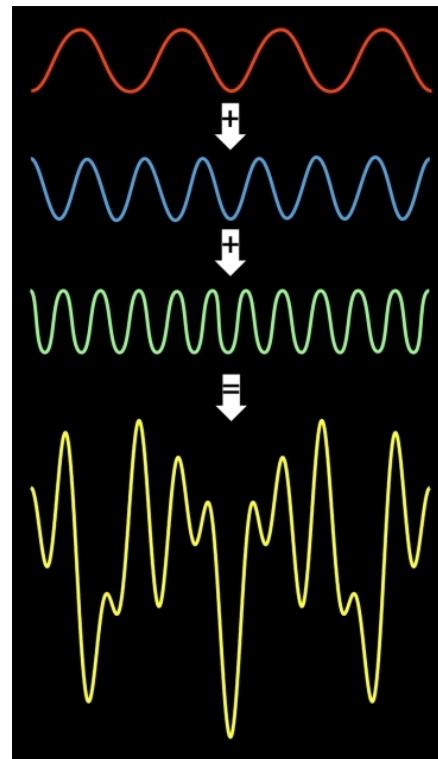
$$F(u) = \int f(x) e^{-j2\pi ux} dx \quad (\text{transformada directa})$$

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du \quad (\text{transformada inversa})$$



Series de Fourier

- Cualquier función periódica puede ser aproximada por la suma de una serie de senos/cosenos de amplitud variable





Transformada de Fourier

- La serie de Fourier puede ser generalizada para derivar la transformada de Fourier

$$F(k) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i k x} dx$$

- Transformada de Fourier inversa

$$f(x) = \int_{-\infty}^{\infty} F(k) e^{2\pi i k x} dk$$

$$e^{xi} = \cos(x) + i \sin(x)$$



Aplicabilidad

- La transformada de Fourier mapea una serie temporal a una serie de frecuencias (amplitudes y fases) que forman la serie temporal
- La transformada de Fourier inversa mapea la serie de frecuencias (amplitudes y fases) a la serie temporal original

Transformada de Fourier Discreta (DFT)



- Versión discreta de la transformada de Fourier
 - Funciones/series temporales muestreadas (discretas)
 - Directa

$$F(k) = \frac{1}{N} \sum_{k=0}^{N-1} f_k e^{-2\pi i x k / N}$$

- Inversa

$$f(x) = \sum_{n=0}^{N-1} F_n e^{-2\pi i k x / N}$$



Elementos

- TF es una función compleja

$$F(k) = R(k) + iI(k)$$

- Magnitud / Amplitud / Espectro
 - Potencia de la señal en una determinada frecuencia

$$A(k) = |F(k)| = \sqrt{R^2(k) + I^2(k)}$$

- Fase $\phi(k) = \arctan\left(\frac{I(k)}{R(k)}\right)$

- Espectro de potencia

$$P(k) = |F(k)|^2 = R^2(k) + I^2(k)$$



Transformada de Fourier en imágenes

- Uso de la transformada de Fourier discreta

$$F(k, l) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-2\pi i \left(\frac{kx}{M} + \frac{ly}{N}\right)}$$

$$f(x, y) = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} F(k, l) e^{2\pi i \left(\frac{kx}{M} + \frac{ly}{N}\right)}$$

- Se realiza una transformación a la imagen original desplazar la posición $F(0,0)$ al centro de la imagen



Direct Current (DC)

- El valor de la transformada en el punto (0,0) en el espectro
 - Media de $f(x,y)$

$$F(0,0) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$$

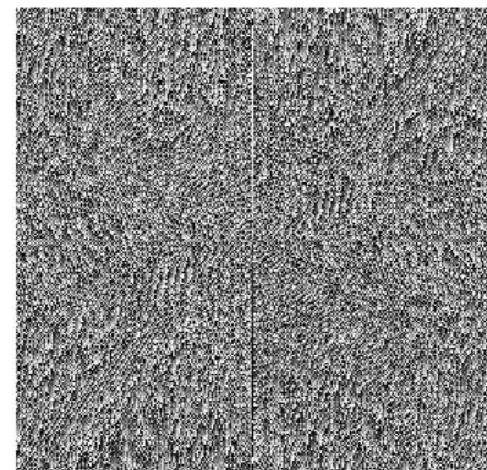


DFT en imágenes

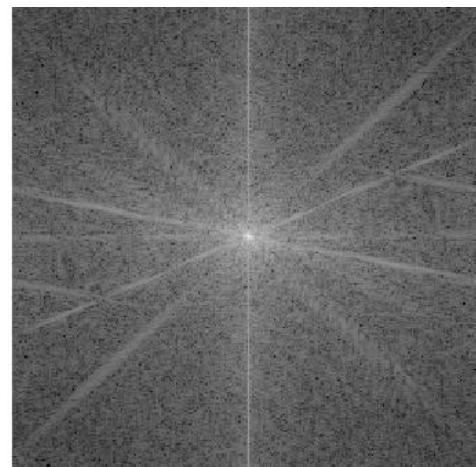
Magnitud



Fase



El rango dinámico de los coeficientes es demasiado grande para ser mostrado en una imagen



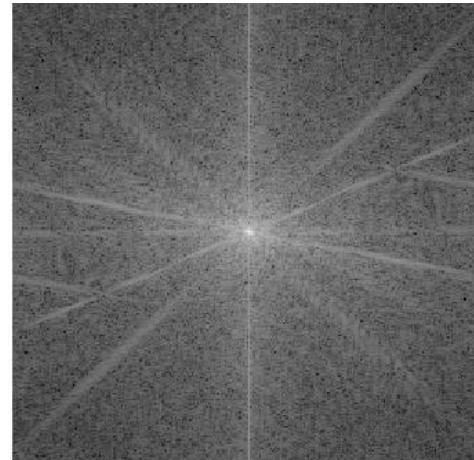
Transformación logarítmica a la imagen de magnitud



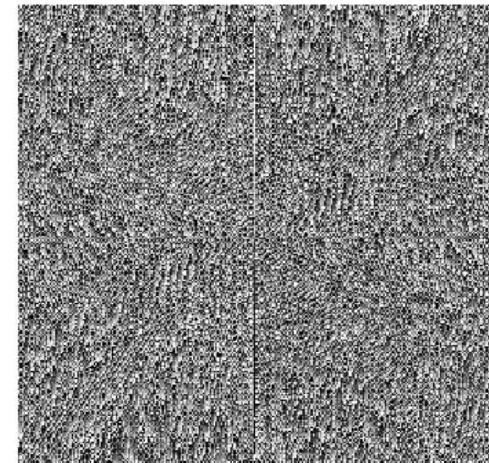
DFT en imágenes



Magnitud



Fase



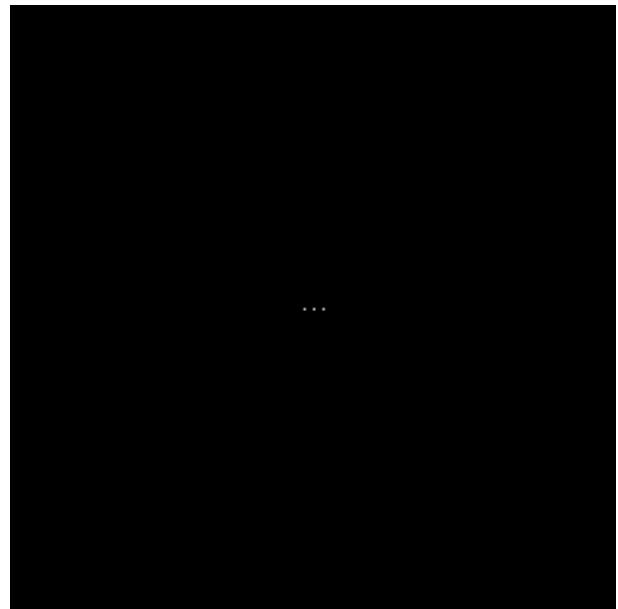
- Componentes de todas las frecuencias
- Punto central es el DC $\rightarrow F(0,0)$
- Frecuencias más altas a medida que nos alejamos del centro
- Intensidad proporcional a la magnitud

El valor de cada punto determina la fase de la frecuencia correspondiente



DFT en imágenes

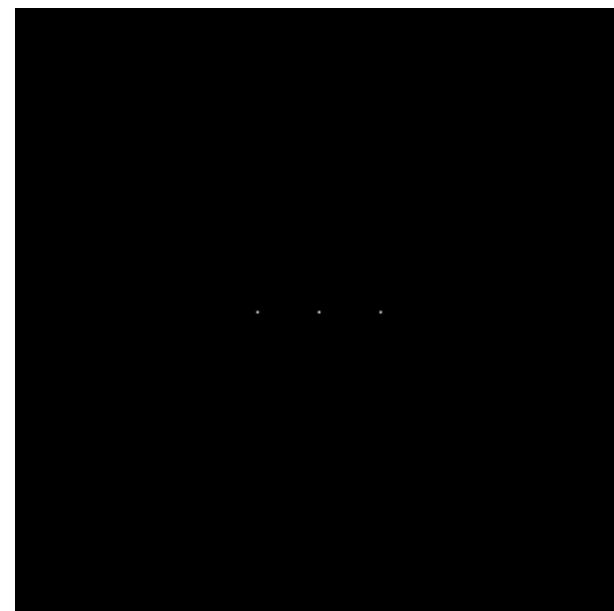
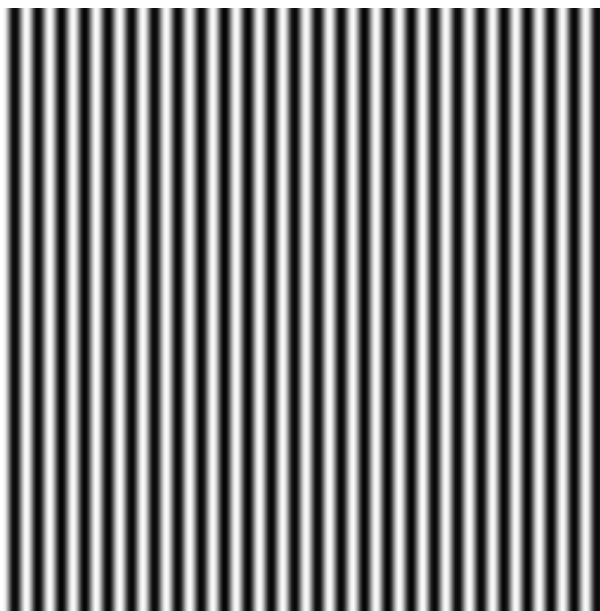
- TF de un seno en 2D
 - El punto central es el DC
 - Los otros dos puntos representan la frecuencia de la función seno
 - No hay píxeles en la dirección x porque la imagen no cambia en esa dirección

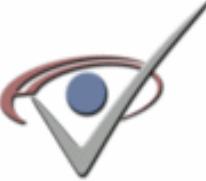




DFT en imágenes

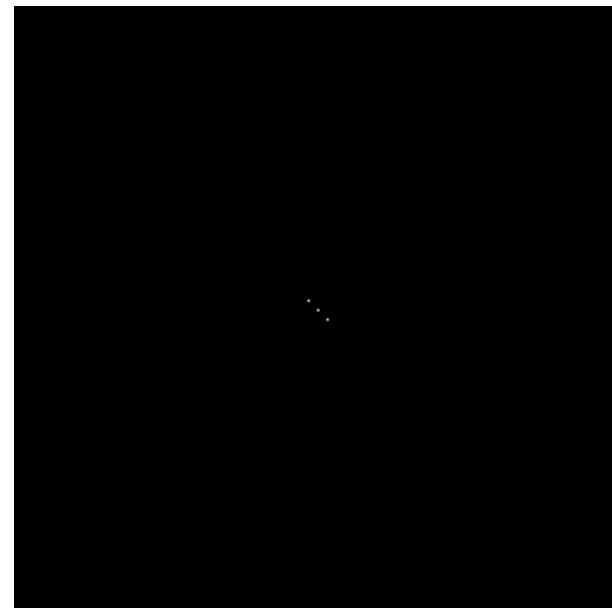
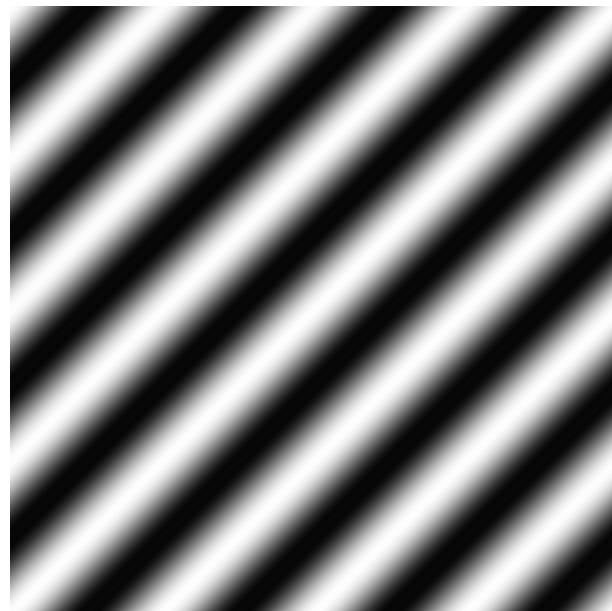
- TF de un seno en 2D con mayor frecuencia
 - Los puntos aparecen más separados → indican una frecuencia más alta

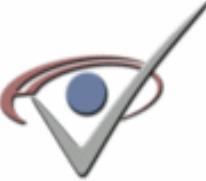




DFT en imágenes

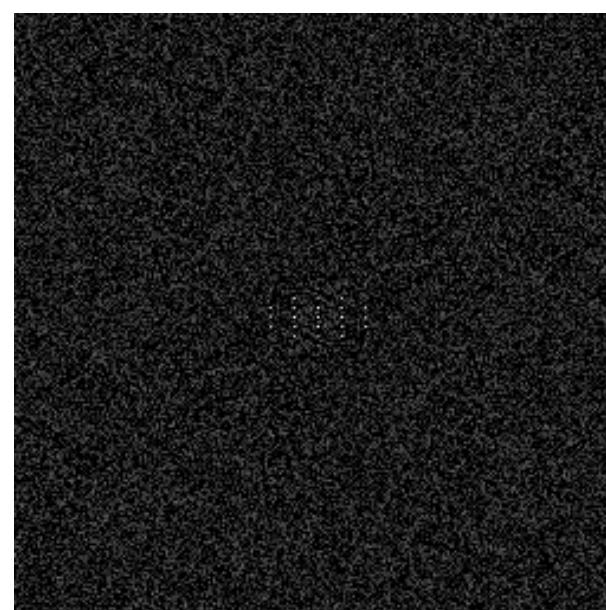
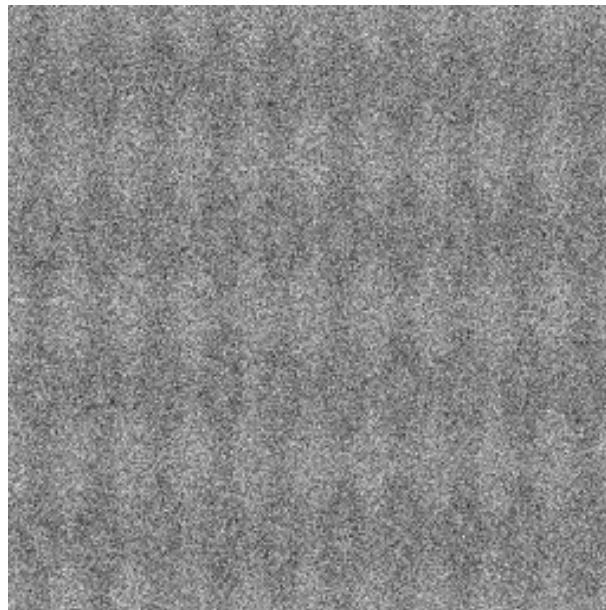
- TF de un seno en diagonal





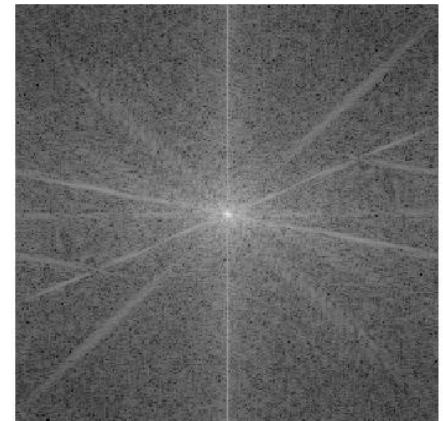
DFT en imágenes

- TF de la suma de un seno en vertical y otro en horizontal

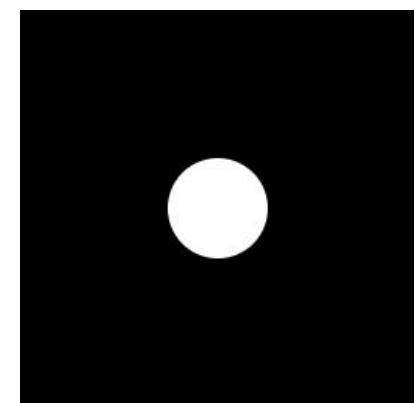


Filtros en frecuencia

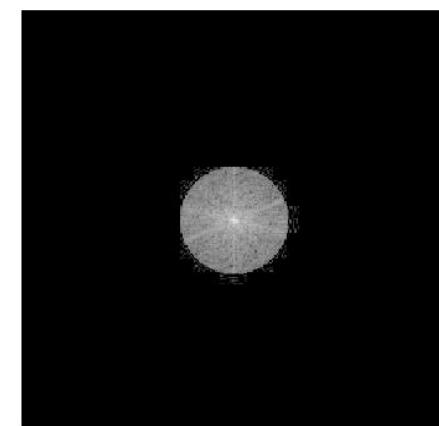
- Consiste en eliminar frecuencias
 - Multiplicar imagen en frecuencia por una máscara
 - Equivalente a una convolución en el dominio espacial

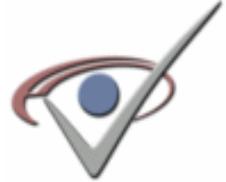


X



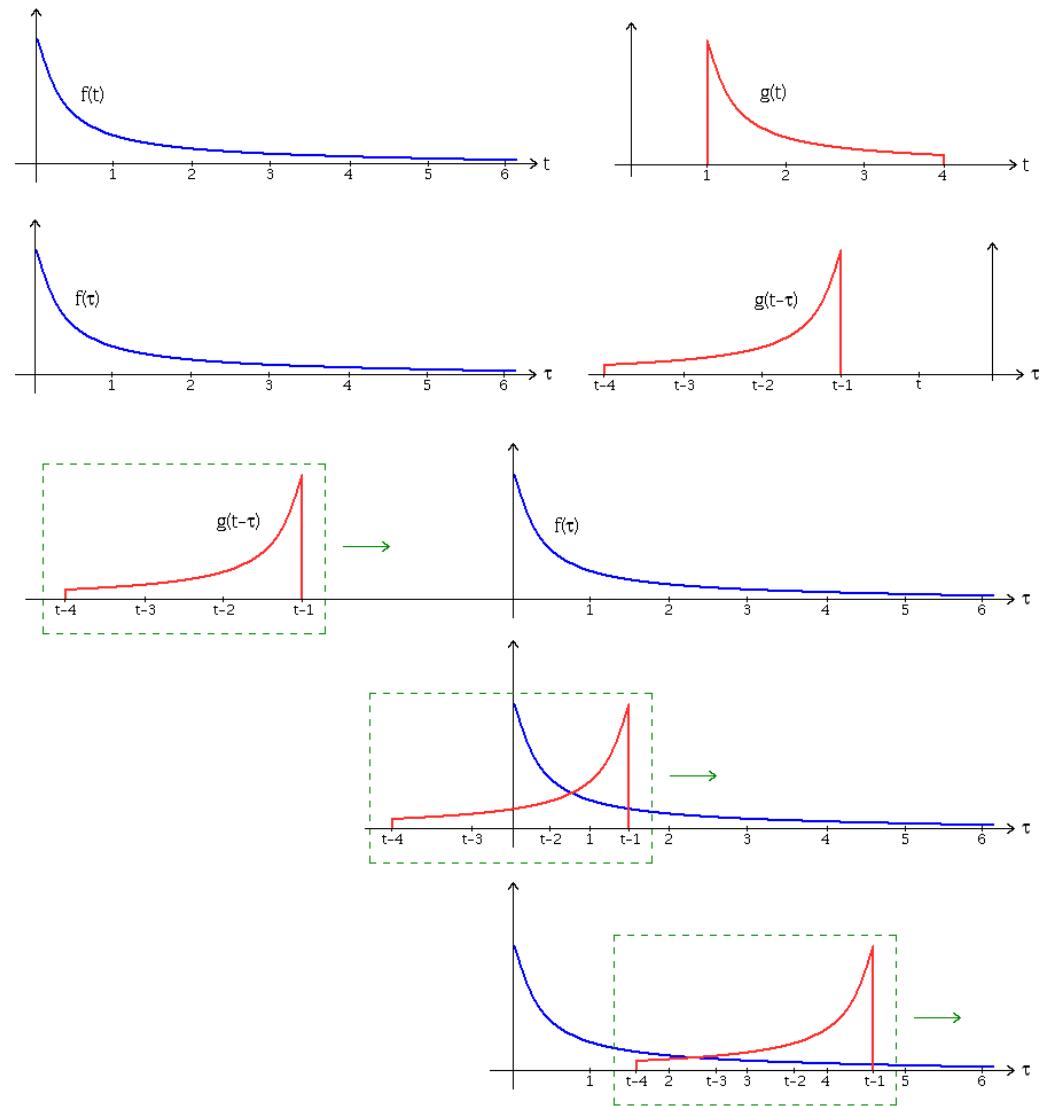
=





Convolución en 1D

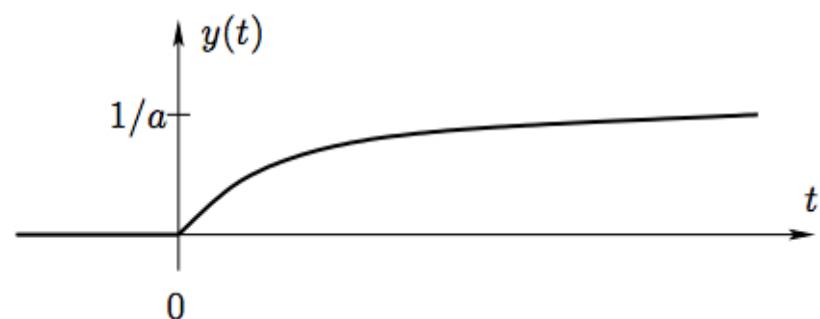
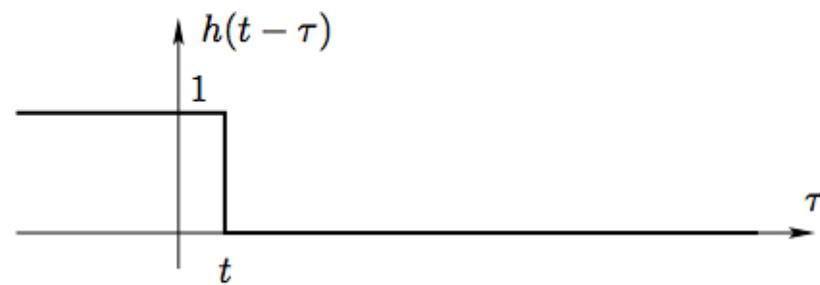
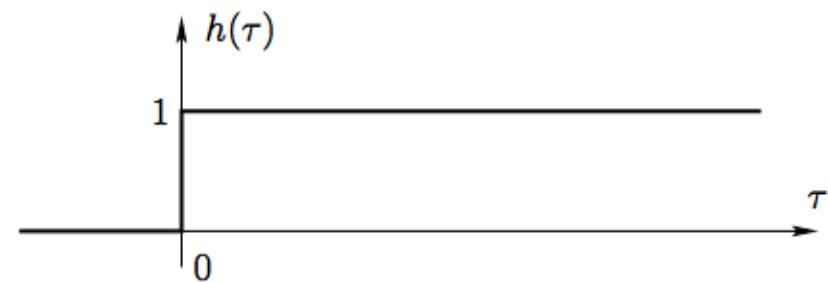
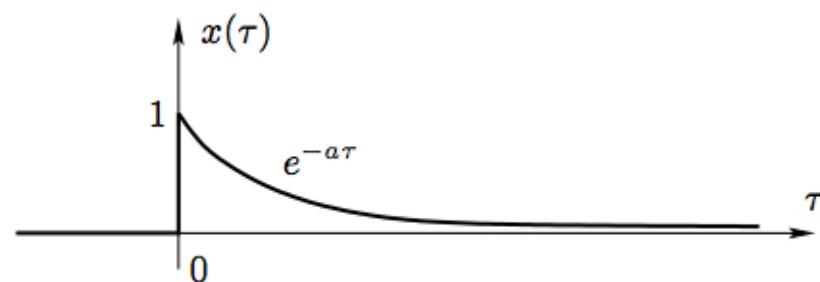
$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)d\tau$$



Ejemplo 1

$$x(t) = e^{-at} u(t), \quad a > 0$$
$$h(t) = u(t)$$

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau$$

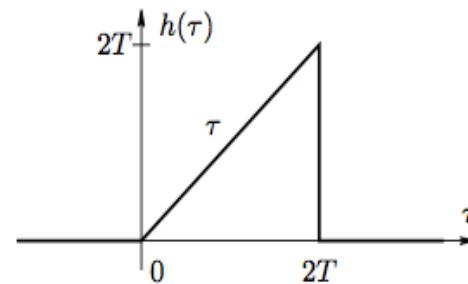
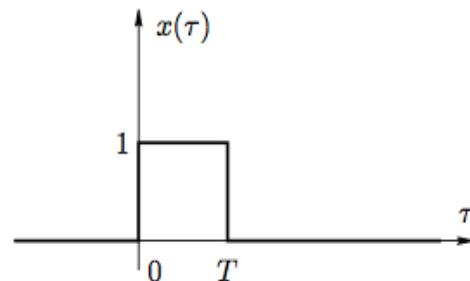


- $t < 0, \quad y(t) = 0$
- $t \geq 0, \quad y(t) = \int_0^t e^{-a\tau} d\tau = \frac{1 - e^{-at}}{a}$

$$y(t) = \frac{1 - e^{-at}}{a} u(t), \quad \forall t$$

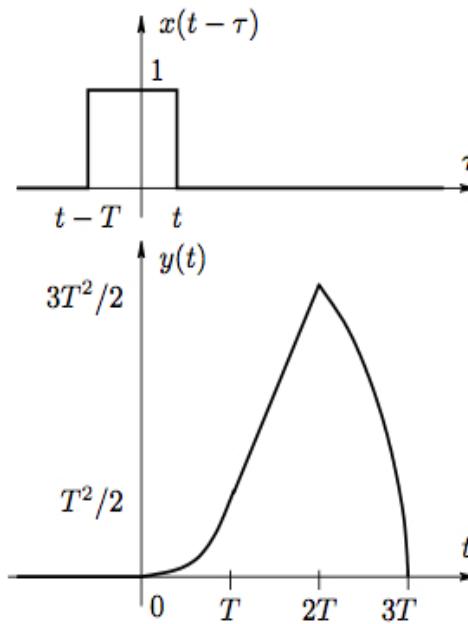
Ejemplo 2

$$x(t) = \begin{cases} 1, & 0 < t < T \\ 0, & \text{resto} \end{cases}, \quad h(t) = \begin{cases} t, & 0 < t < T \\ 0, & \text{resto} \end{cases}$$



- $t \leq 0, y(t) = 0$
- $\begin{cases} t > 0 \\ t - T \leq 0 \end{cases} \Rightarrow 0 < t \leq T, y(t) = \int_0^t \tau d\tau = \frac{t^2}{2}$
- $\begin{cases} t - T > 0, \\ t \leq 2T \end{cases} \Rightarrow T < t \leq 2T, y(t) = \int_{t-T}^t \tau d\tau = tT - \frac{1}{2}T^2$
- $\begin{cases} t > 2T, \\ t - T \leq 2T \end{cases} \Rightarrow 2T < t \leq 3T, y(t) = \int_{t-T}^{2T} \tau d\tau = tT - \frac{1}{2}t^2 + \frac{3}{2}T^2$
- $t - T > 3T \rightarrow t > 3T, y(t) = 0$

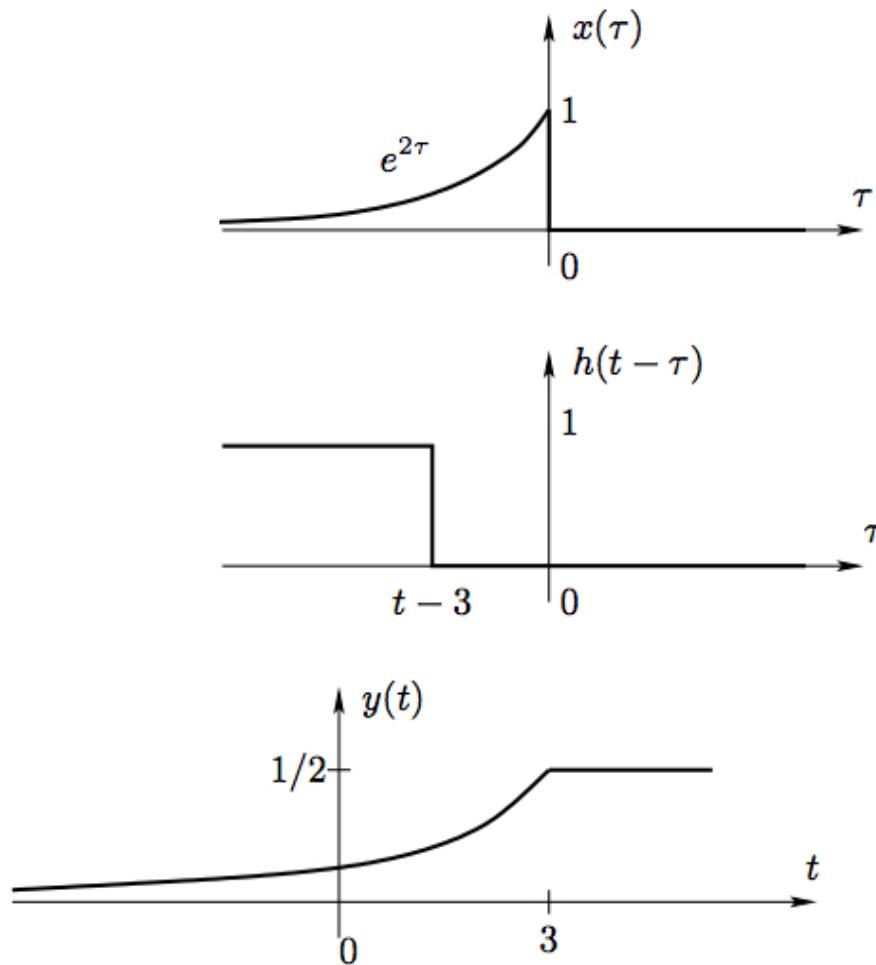
$$y(t) = \frac{t^2}{2} u(t) - \frac{1}{2}(t-T)^2 u(t-T) + \left(2T^2 - \frac{1}{2}t^2\right) u(t-2T) + \left(\frac{1}{2}t^2 - tT - \frac{3}{2}T^2\right) u(t-3T)$$



Ejemplo 3

$$x(t) = e^{2t} u(-t)$$
$$h(t) = u(t - 3)$$

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t - \tau)d\tau$$



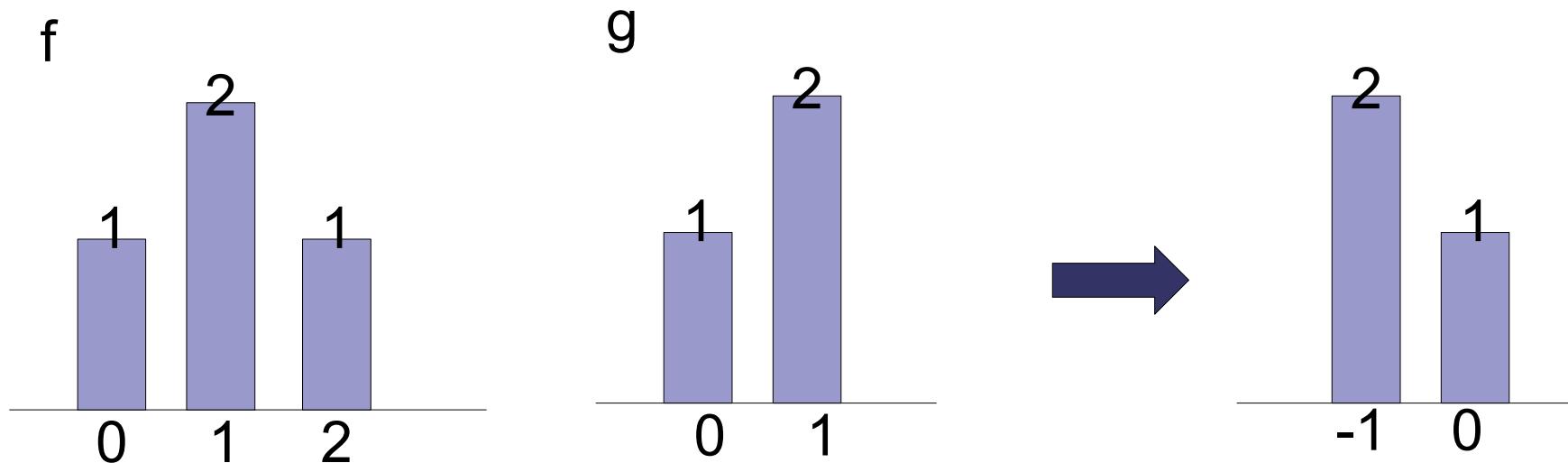
- $t - 3 < 0 \Rightarrow t < 3, y(t) = \int_{-\infty}^{t-3} e^{2\tau} d\tau = \frac{1}{2}e^{2(t-3)}$
- $t - 3 \geq 0 \Rightarrow t \geq 3, y(t) = \int_{-\infty}^0 e^{2\tau} d\tau = \frac{1}{2}$

$$y(t) = \frac{1}{2}e^{2(t-3)} + \frac{1}{2}[1 - e^{2(t-3)}]u(t-3)$$



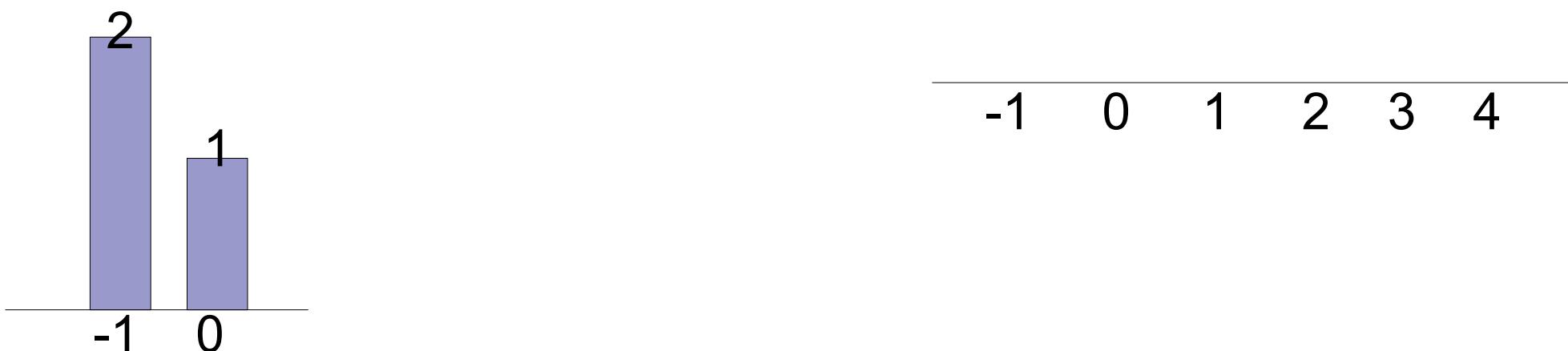
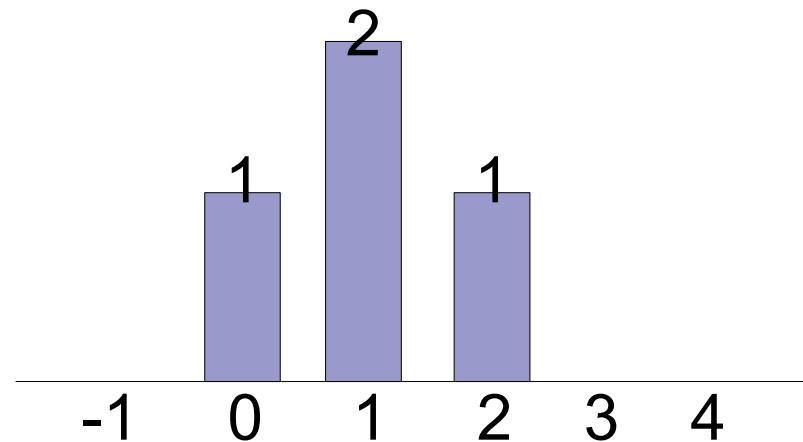
Convolución 1D: Caso discreto

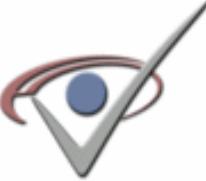
$$\begin{aligned}(f * g)(n) &= \sum_{m=-\infty}^{\infty} f(n)g(n-m) \\ &= \sum_{m=-\infty}^{\infty} f(n-m)g(m)\end{aligned}$$



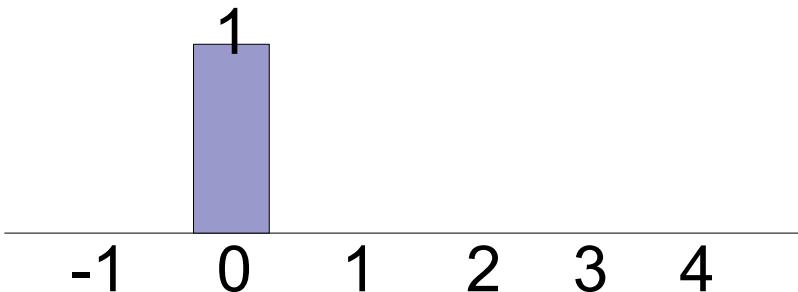
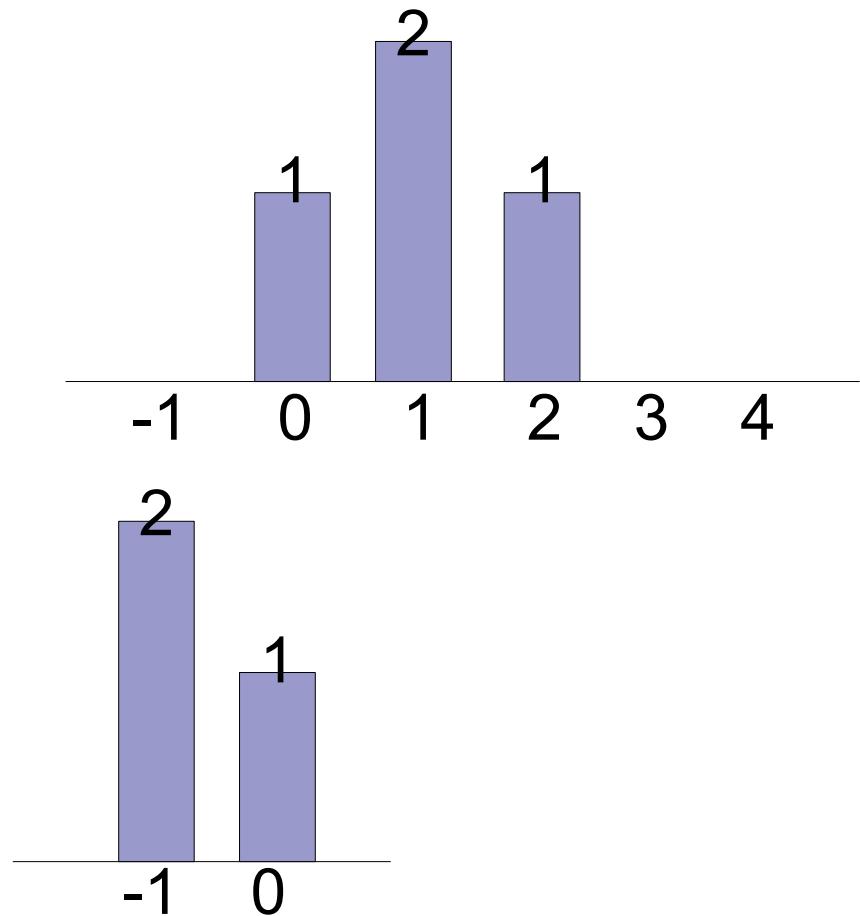


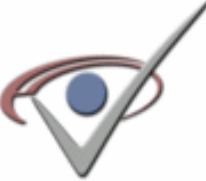
Convolución 1D: Caso discreto



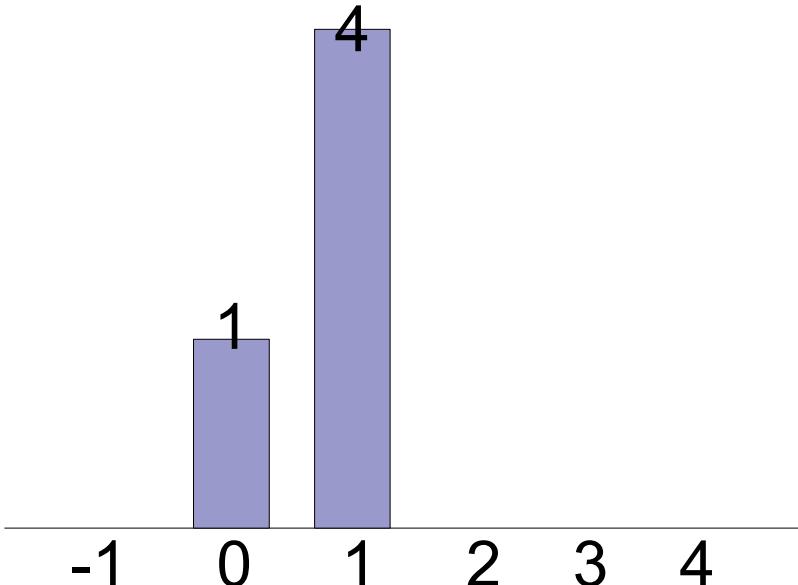
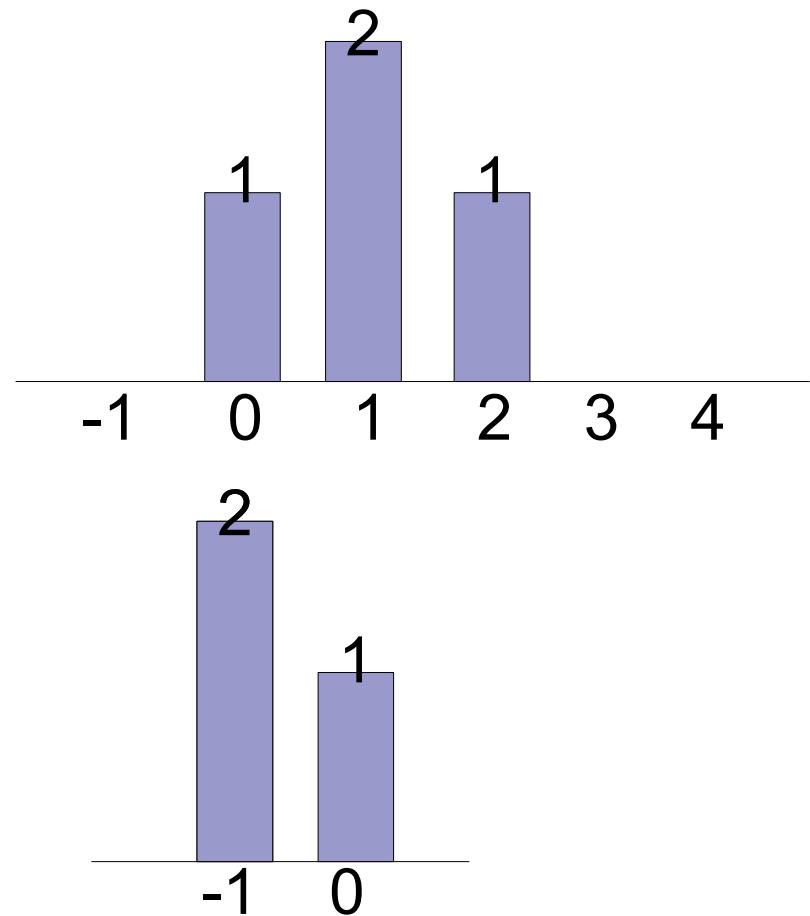


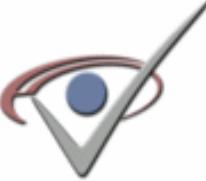
Convolución 1D: Caso discreto



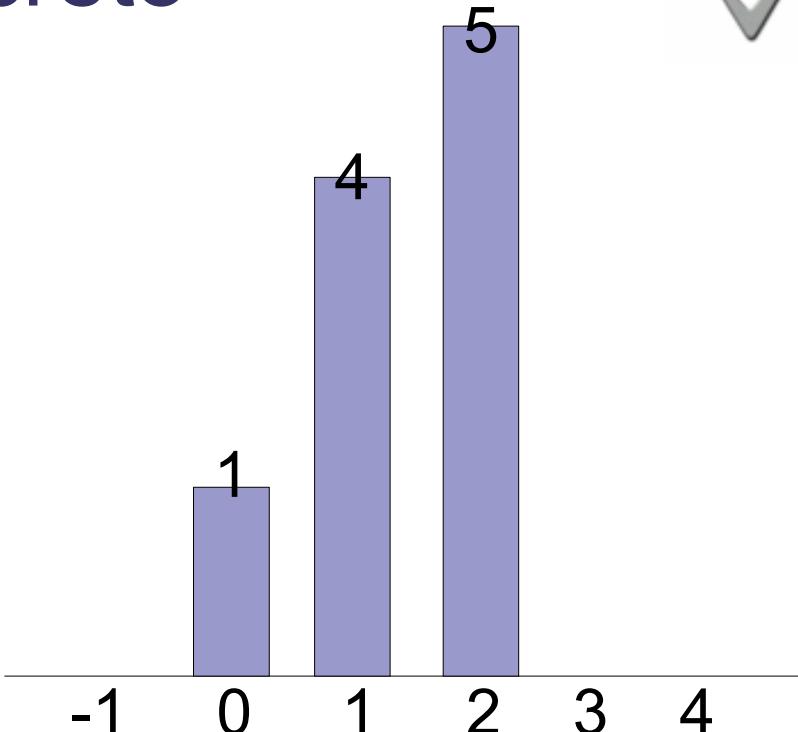
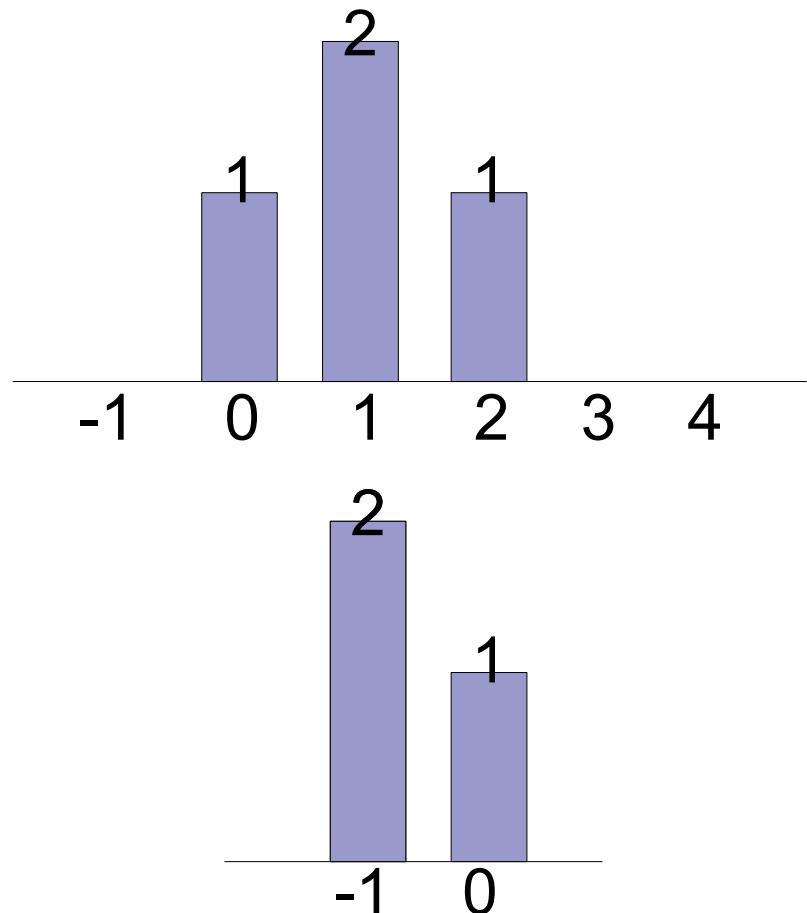


Convolución 1D: Caso discreto



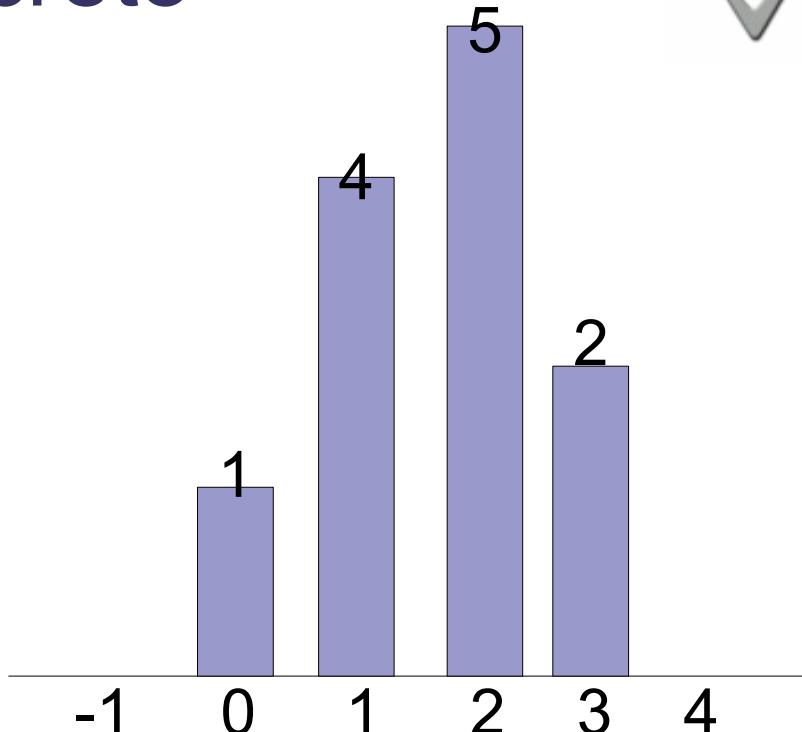
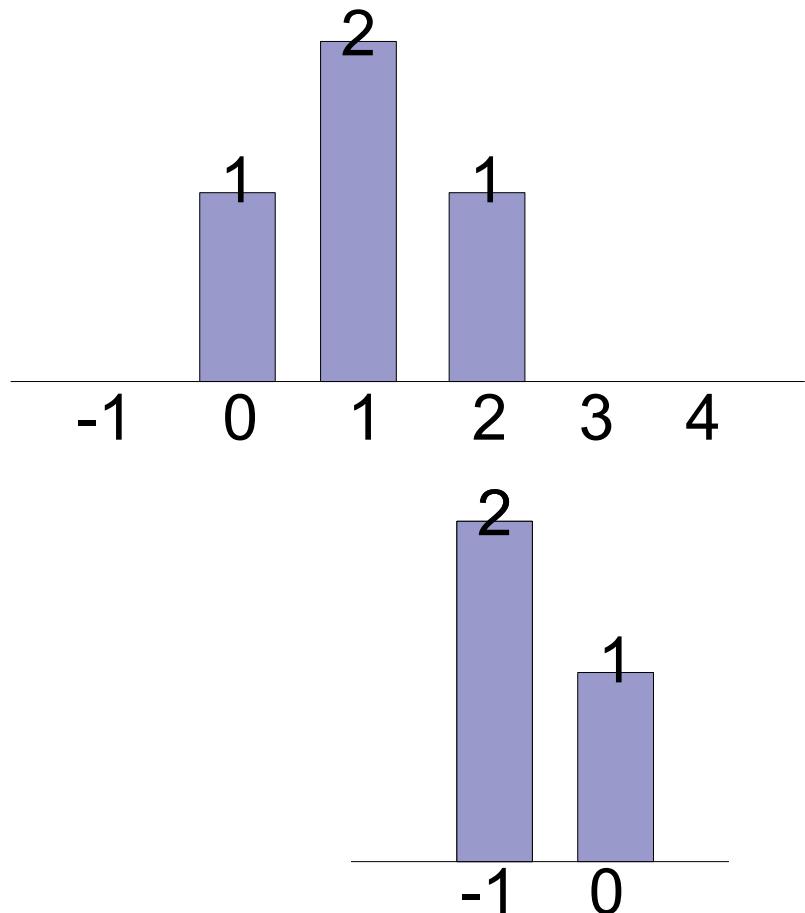


Convolución 1D: Caso discreto





Convolución 1D: Caso discreto

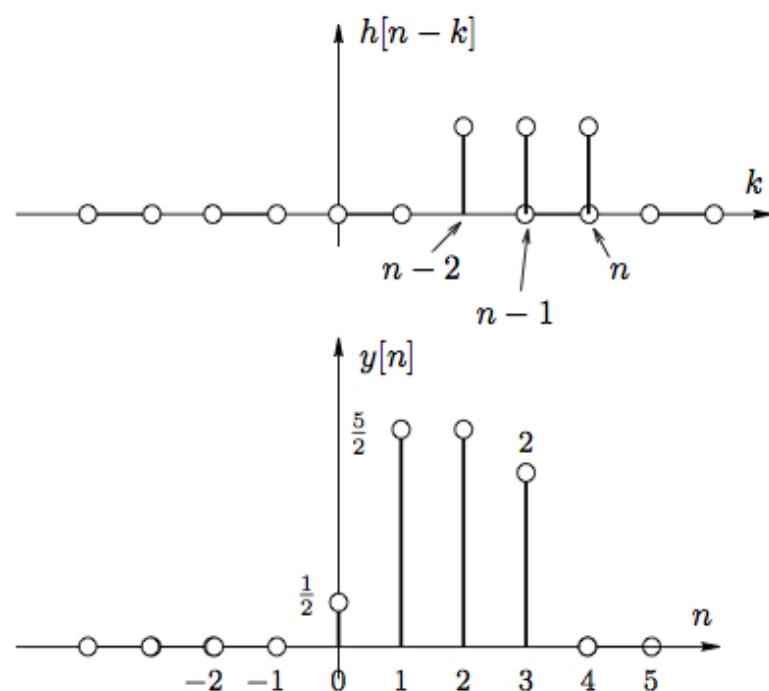
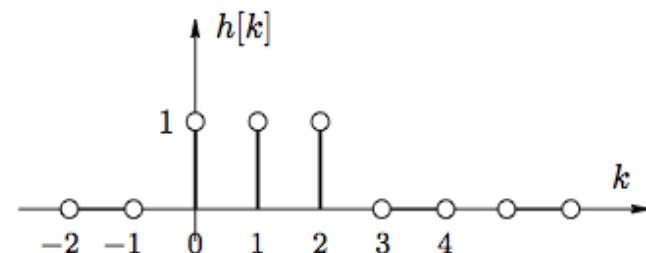
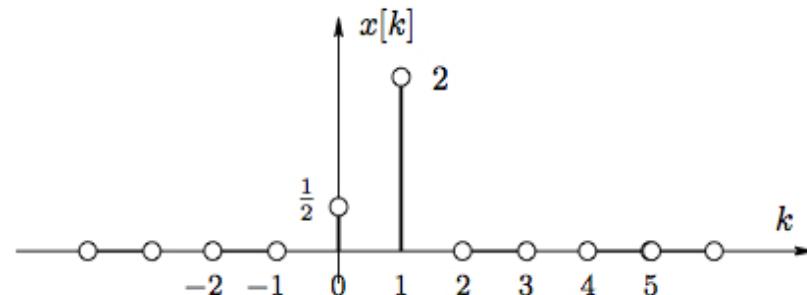


Ejemplo 1

$$x[n] = \frac{1}{2}\delta[n] + 2\delta[n - 1]$$

$$h[n] = u[n] - u[n - 3]$$

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

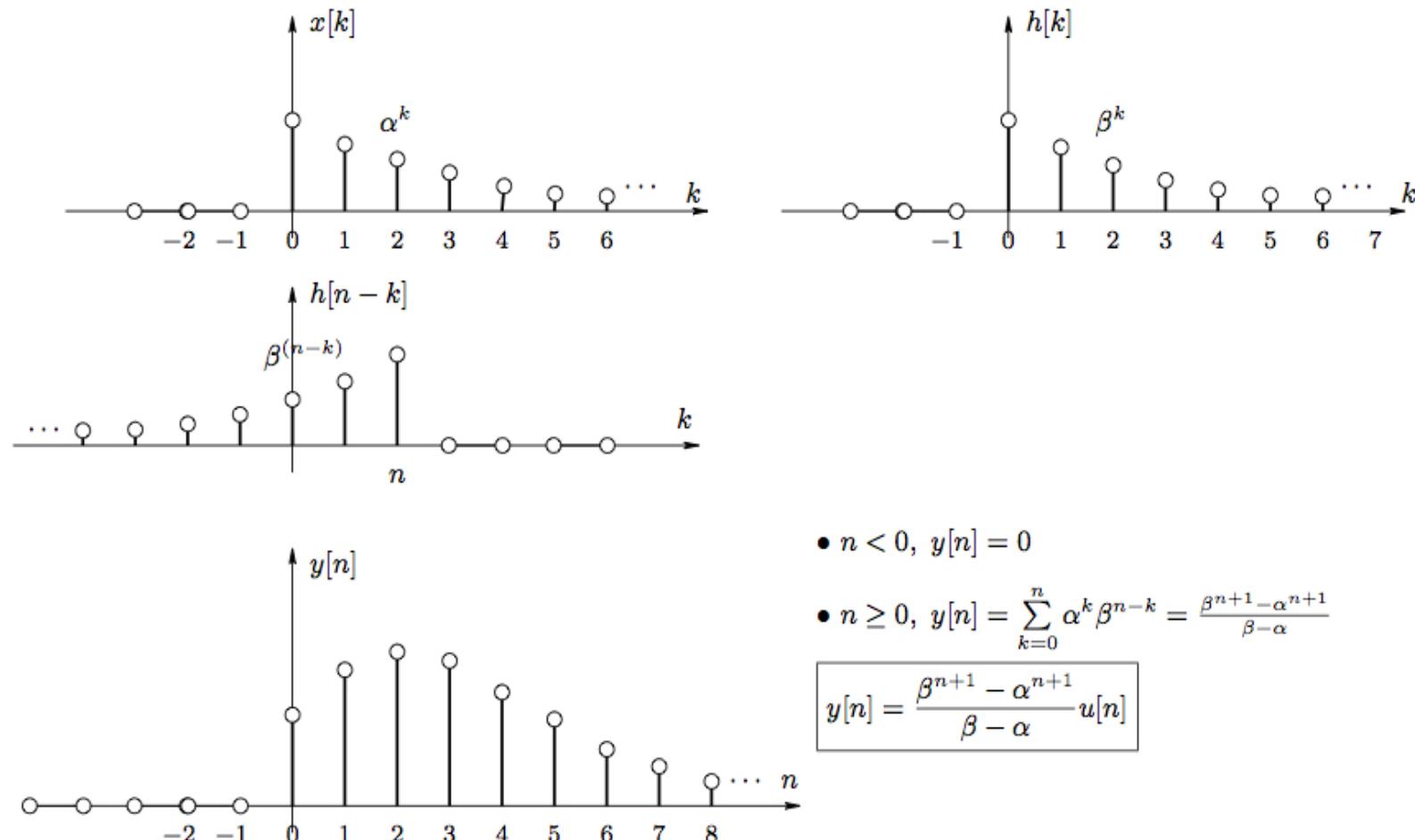


- $n < 0, y[n] = 0$
- $n = 0, y[0] = \sum_{k=-\infty}^{\infty} x[k]h[0-k] = x[0]h[0] = \frac{1}{2}$
- $n = 1, y[1] = \sum_{k=-\infty}^{\infty} x[k]h[1-k] = x[0]h[1] + x[1]h[0] = \frac{5}{2}$
- $n = 2, y[2] = \sum_{k=-\infty}^{\infty} x[k]h[2-k] = x[0]h[2] + x[1]h[1] = \frac{5}{2}$
- $n = 3, y[3] = \sum_{k=-\infty}^{\infty} x[k]h[3-k] = x[1]h[2] = 2$
- $n > 3, y[n] = 0$

Ejemplo 2

$$x[n] = \alpha^n u[n], \quad \alpha \neq \beta, \\ h[n] = \beta^n u[n], \quad 0 < \alpha, \beta < 1.$$

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \\ = \sum_{k=-\infty}^{\infty} \alpha^k u[k] \beta^{n-k} u[n-k] = \sum_{k=0}^{\infty} \alpha^k \beta^{n-k} u[n-k].$$

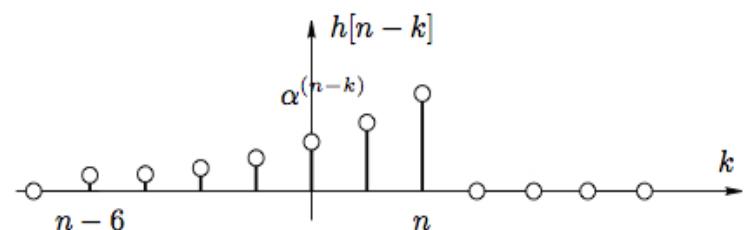
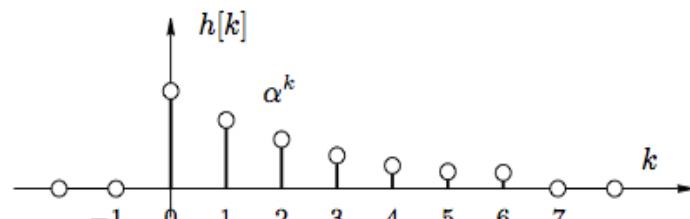
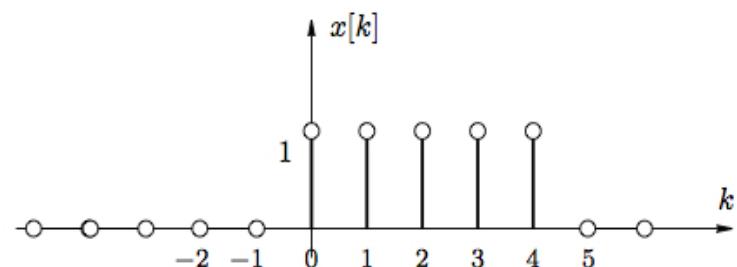




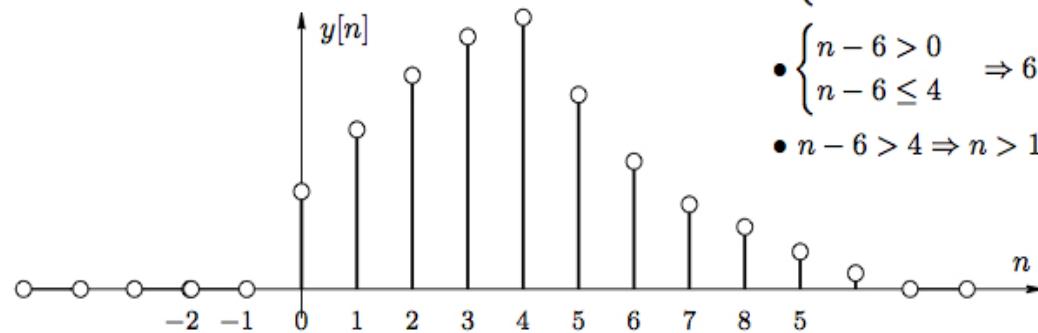
Ejemplo 3

$$x[n] = \begin{cases} 1, & 0 \leq n \leq 4 \\ 0, & \text{resto} \end{cases}, \quad h[n] = \begin{cases} \alpha^n, & 0 \leq n \leq 6 \\ 0, & \text{resto} \end{cases}$$

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$



- $n < 0, y[n] = 0$
- $0 \leq n \leq 4, y[n] = \sum_{k=0}^n \alpha^{n-k} = \frac{1-\alpha^{n+1}}{1-\alpha}$
- $\begin{cases} n > 4 \\ n-6 \leq 0 \end{cases} \Rightarrow 4 < n \leq 6, y[n] = \sum_{k=0}^4 \alpha^{n-k} = \frac{\alpha^{n-4}-\alpha^1}{1-\alpha}$
- $\begin{cases} n-6 > 0 \\ n-6 \leq 4 \end{cases} \Rightarrow 6 < n \leq 10, y[n] = \sum_{k=n-6}^4 \alpha^{n-k} = \frac{\alpha^{n-4}-\alpha^7}{1-\alpha}$
- $n-6 > 4 \Rightarrow n > 10, y[n] = 0$



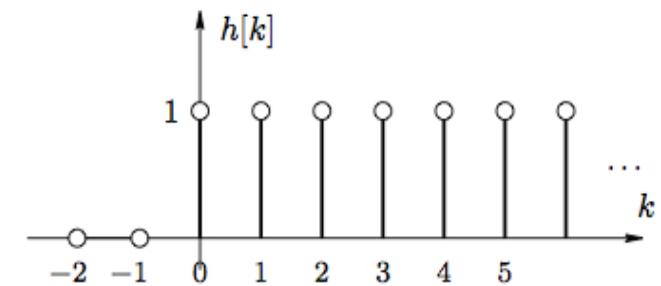
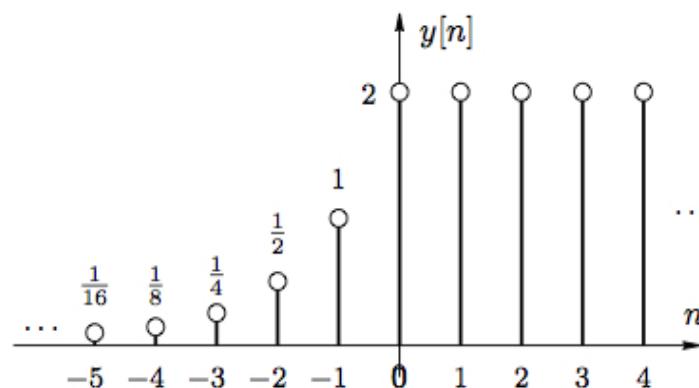
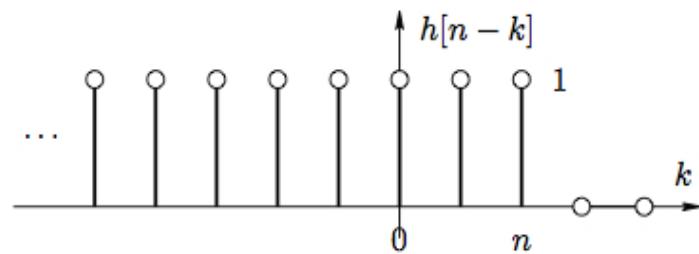
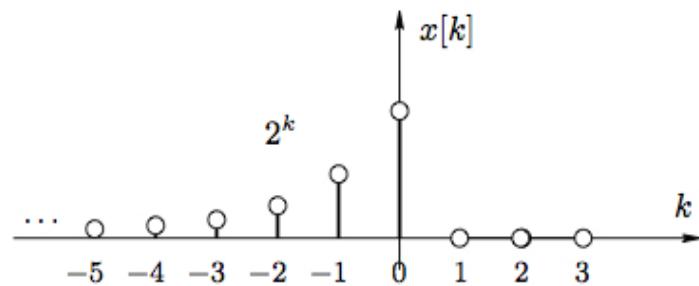


Ejemplo 4

$$x[n] = 2^n u[-n]$$

$$h[n] = u[n]$$

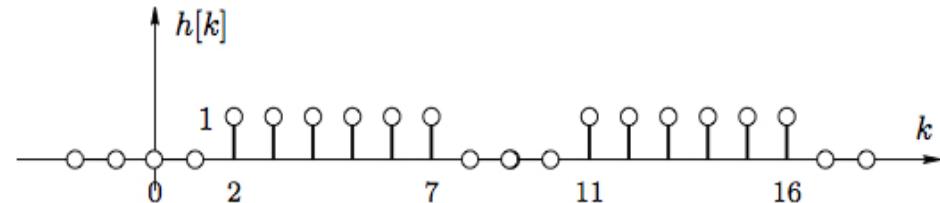
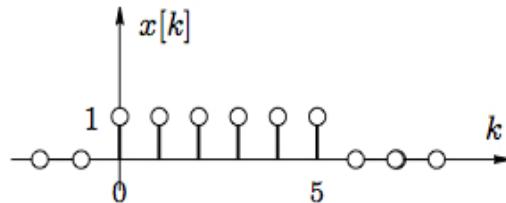
$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$



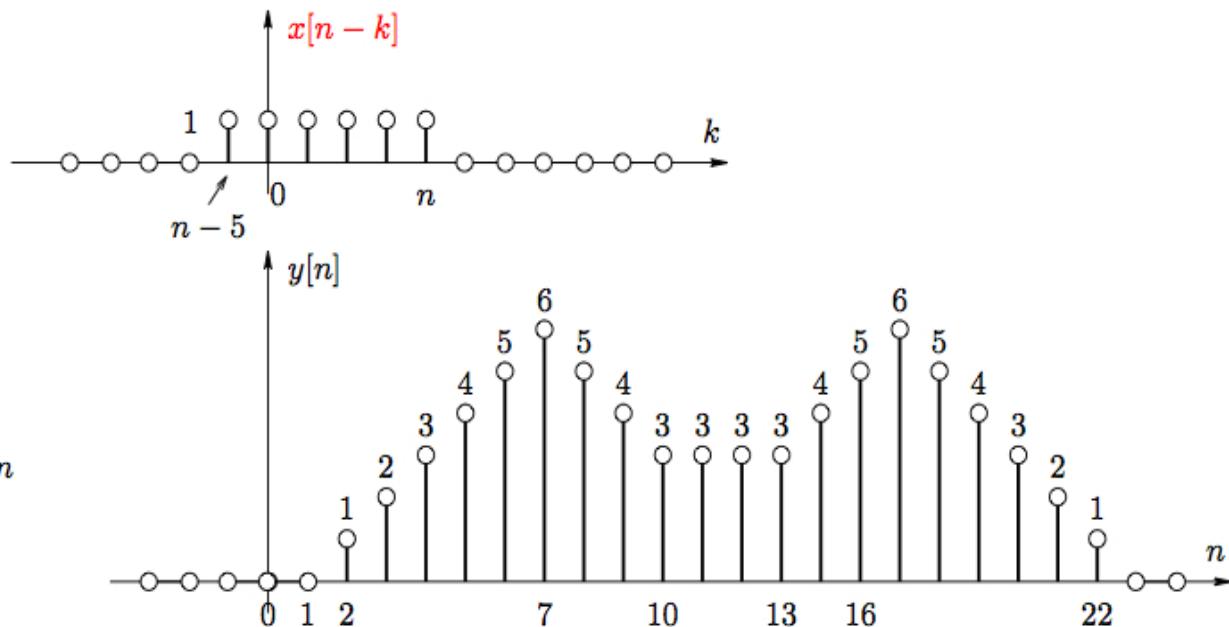
- $n < 0, y[n] = \sum_{k=-\infty}^n 2^k = 2^{n+1}$
- $n \geq 0, y[n] = \sum_{k=-\infty}^0 2^k = 2$

Ejemplo 5

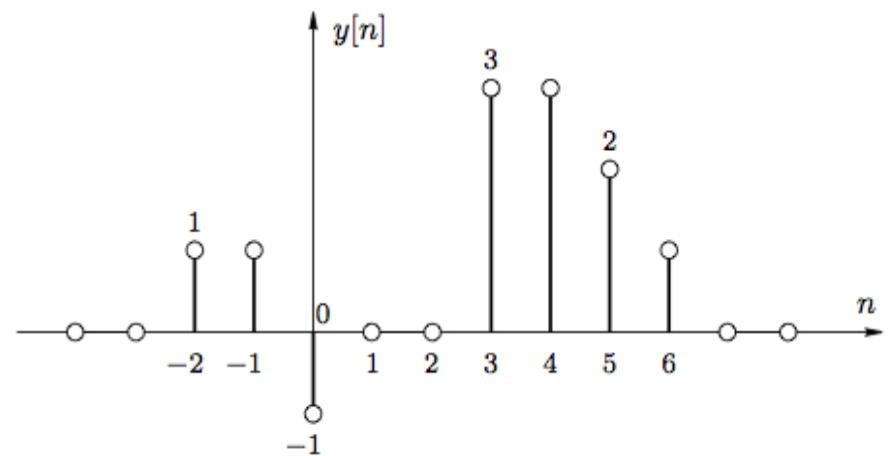
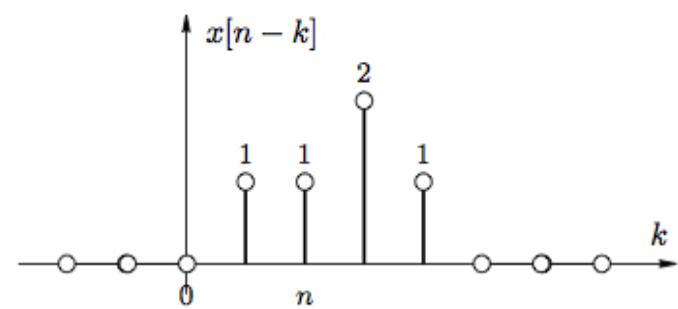
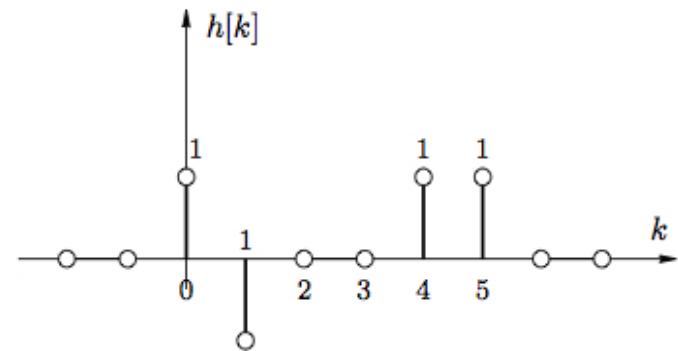
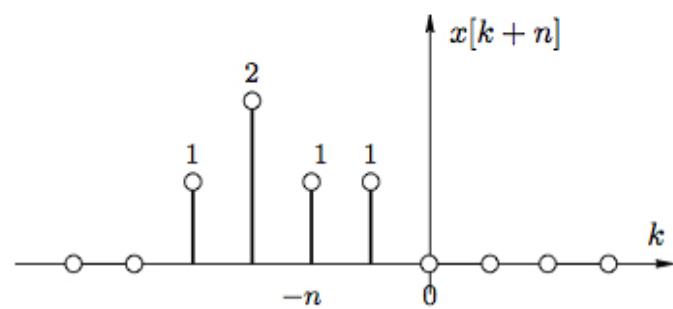
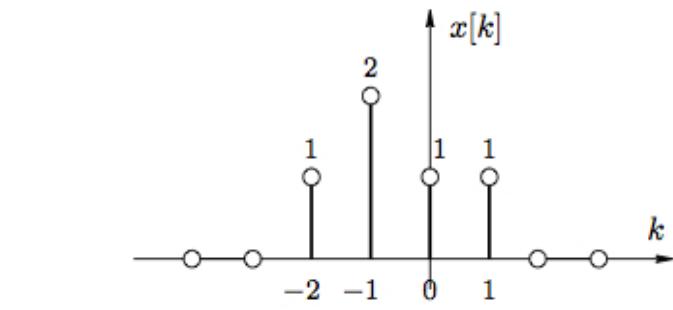
$$x[n] = \begin{cases} 1, & 0 \leq n \leq 5 \\ 0, & \text{resto} \end{cases}, \quad h[n] = \begin{cases} 1, & 2 \leq n \leq 7, 11 \leq n \leq 16 \\ 0, & \text{resto} \end{cases}$$



- $n < 2, y[n] = 0$
- $2 \leq n \leq 7, y[n] = \sum_{k=2}^n 1 = n - 1$
- $8 \leq n \leq 10, y[n] = \sum_{k=n-5}^7 1 = 13 - n$
- $11 \leq n \leq 12, y[n] = 3$
- $13 \leq n \leq 16, y[n] = \sum_{k=11}^n 1 = n - 10$
- $17 \leq n \leq 21, y[n] = \sum_{k=n-5}^{16} 1 = 22 - n$
- $n \geq 22, y[n] = 0$

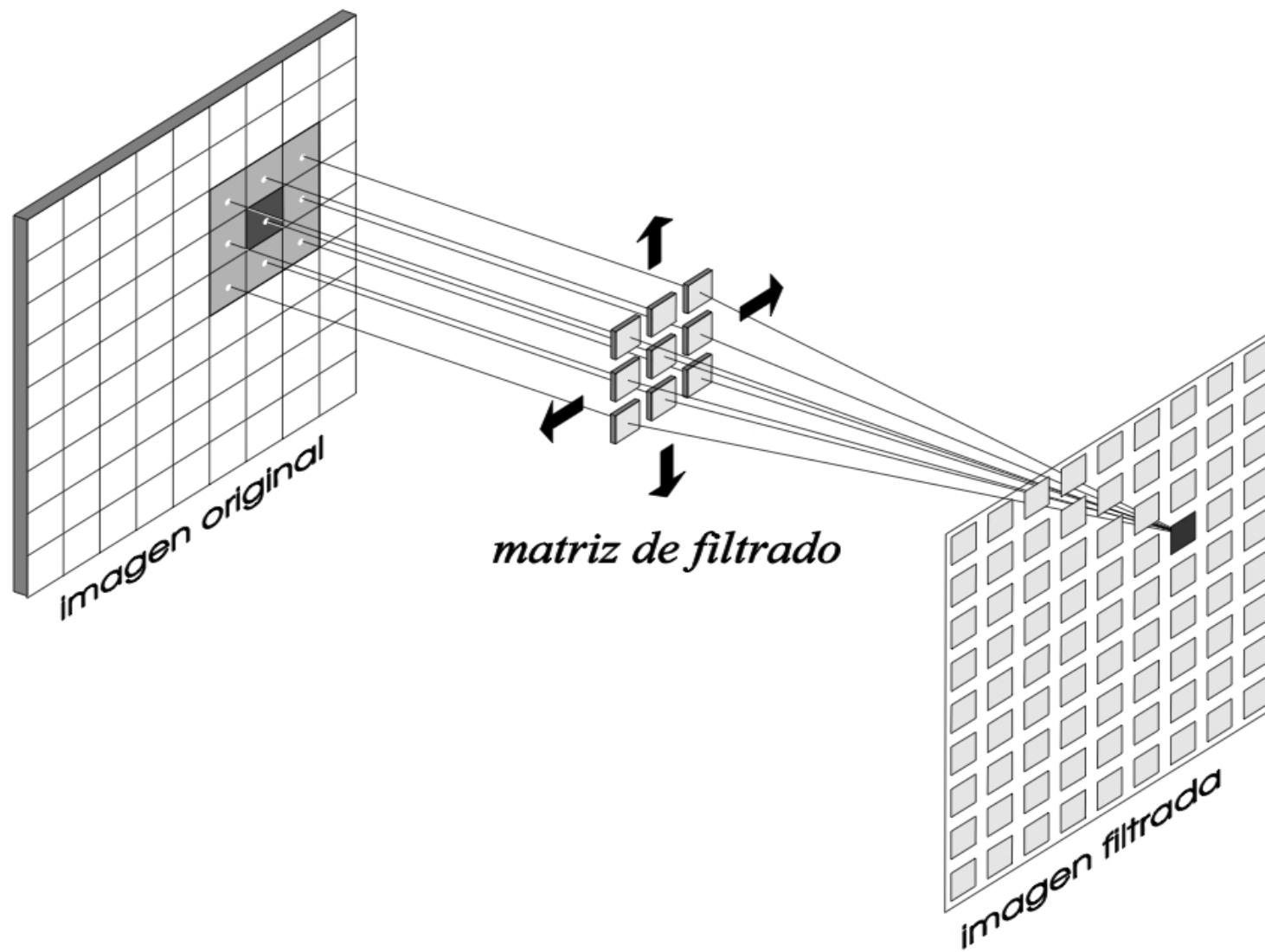


Ejemplo 6





Convolución 2D en imágenes





Convolución en imágenes 2D

$$(f * g)(n) = \sum_{m=-\infty}^{\infty} f(n-m)g(m)$$

Dada una imagen f de tamaño $M \times N$ y una máscara g de tamaño $m \times n$

$$(f * g)(i, j) = \sum_{s=-a}^a \sum_{t=-b}^b f(i+s, j+t) g(s, t), \quad i=a..M-a, j=b..N-b$$

	(i,j-a)	
(i-b,j)	(i,j)	(i+b,j)
	(i,j+a)	

$$a = \frac{m-1}{2}$$

$$b = \frac{n-1}{2}$$

	(0,-a)	
(-b,0)	(0,0)	(b,0)
	(0,a)	



Convolución en imágenes 2D

$$(f * g)(2,2) = \sum_{s=-1}^1 \sum_{t=-1}^1 f(2+s, 2+t)g(s, t) =$$

$$f(1,1)g(-1,-1) + f(2,1)g(0,-1) + f(3,1)g(1,-1) +$$

$$f(1,2)g(-1,0) + f(2,2)g(0,0) + f(3,2)g(1,0) +$$

$$f(1,3)g(-1,1) + f(2,3)g(0,1) + f(3,3)g(1,1)$$

45	60	98	127	132	133	137	133
46	65	98	123	126	128	131	133
47	65	96	115	119	123	135	137
47	63	91	107	113	122	138	134
50	59	80	97	110	123	133	134
49	53	68	83	97	113	128	133
50	50	58	70	84	102	116	126
50	50	52	58	69	86	101	120

$f(i,j)$

$$* \quad \begin{array}{|c|c|c|} \hline 0.1 & 0.1 & 0.1 \\ \hline 0.1 & 0.2 & 0.1 \\ \hline 0.1 & 0.1 & 0.1 \\ \hline \end{array} =$$

$g(s,t)$

69	95	116	125	129	132
68	92	110	120	126	132
66	86	104	114	124	132
62	78	94	108	120	129
57	69	83	98	112	124
53	60	71	85	100	114

$(f * g)(i,j)$



Filtros lineales: convolución

- Un filtro de tamaño $m \times n$ produce imágenes de tamaño $M - (m-1) \times N - (n-1)$
 - No se pueden convolucionar más allá de los bordes de la imagen
- Para conseguir imágenes con el mismo tamaño que la original
 - Utilizar “máscaras incompletas” en los bordes
 - Padding (relleno)
 - Ceros
 - Valor constante
 - Valores de los bordes de la imagen



Filtros linealmente separables

$$f * g = (f * h) * v \quad g = hv$$

donde

- g es una máscara 2D
- h es una máscara horizontal 1D
- v es una máscara vertical 1D

$$\begin{matrix} & \begin{matrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{matrix} \\ 1/16 & \end{matrix} = 1/4 \begin{matrix} & \begin{matrix} 1 & 2 & 1 \end{matrix} \\ \boxed{1} & \end{matrix} \times 1/4 \begin{matrix} & \begin{matrix} 1 \\ 2 \\ 1 \end{matrix} \\ \boxed{1} & \end{matrix}$$



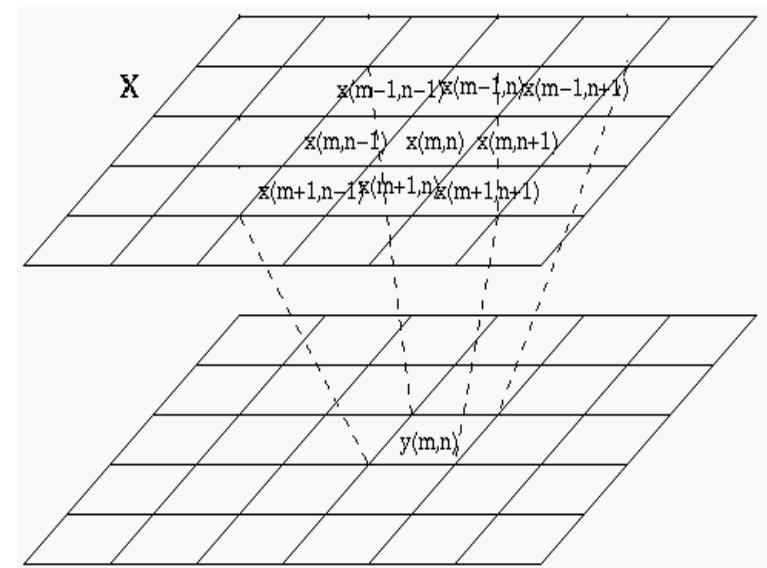
Filtros linealmente separables

- Para una máscara de tamaño $K \times K$
 - Realizar una convolución en 2D requiere K^2 operaciones por pixel
 - Realizar una convolución 1D por filas seguida de una convolución 1D por columnas requiere $2K$ operaciones por pixel
- Eficiencia computacional



Filtros no lineales

- La salida se obtiene al aplicar una operación no lineal a los pixeles de un vecindario de tamaño $m \times n$
 - Mediana
 - Máximo del vecindario
 - Mínimo del vecindario
 - α -trimmed mean
 - Media del vecindario excluyendo un porcentaje α de los valores superiores e inferiores





Aplicaciones de los filtros espaciales

- Suavizar
- Enfatizar detalles
- Acentuar bordes
- Eliminar ruido



Mejora de la Imagen

Referencias

- R. C. Gonzalez and R. E. Woods. Digital Image Processing. Addison-Wesley Longman Publishing Co., Inc., 2001.
- E. Trucco and A. Verri. Introductory techniques for 3D computer Vision, Prentice-Hall, 1998.
- R. Szeliski, Computer Vision: Algorithms and Applications Microsoft Research, 2010. Draft online: <http://szeliski.org/Book/>
- M. Sonka, V. Hlavac, and R. Boyle. Image Processing: Analysis and Machine Vision. Thomson-Engineering, 1998.



Mejora de Imágenes

- Los procesos de mejora de imágenes se centran normalmente en dos tipos de operaciones:
 - **Suavizado:** Eliminar ruido presente en la imagen
 - **Realce:** Resaltar características de la imagen tales como bordes o zonas de poco contraste, previo a un análisis de más alto nivel



Ruido

Definición (Ruido)

- *Error que puede degradar la calidad de una imagen*
- *El ruido se puede producir durante la adquisición, la transmisión o el procesamiento de la imagen.*

Definición (Robustez)

*Un sistema de visión artificial es **robusto** si genera los mismos resultados con ruido y en ausencia de ruido*



Ruido

Tipos de ruido

Las características probabilísticas permiten distinguir dos **clases de ruido**

- **Ruido local determinístico**

- Presenta una forma y apariencia constante para cada sistema en particular.
- Se manifiesta como una distribución constante y determinada dentro de la imagen.
- Un ejemplo típico es la aparición en la imagen de una franja con unos niveles de intensidad sensiblemente diferentes a los del resto de la imagen.

- **Ruido aleatorio:** es producido por fuentes imprevistas y su distribución es aleatoria y cambiante con el tiempo.



Ruido

Ruido semántico

- Ruido intrínseco provocado por la **textura** de los objetos o por un **entorno especial** de la escena
- Ejemplo: imágenes subacuáticas o con presencia de humo o vapor.

Ruido: origen

El ruido se puede producir en la imagen durante su

- Su adquisición
- Su transmisión
- Su procesamiento



Ruido provocado durante la adquisición de la imagen (1/5)

Categorías de los dispositivos de adquisición de imágenes

- Sensores fotoquímicos: películas o diapositivas fotográficas.
- Sensores fotoelectrónicos: tubos *vidicon*, dispositivos de inyección de carga (CID, *Charge Injection Devices*) y dispositivos de carga acoplada (CCD, *Charge Coupled Devices*).



Nota (Ruido provocado durante la adquisición de la imagen (2/5))

- *Aunque los dispositivos fotoquímicos pueden detectar y grabar la imagen al mismo tiempo, la imagen que producen no puede ser fácilmente digitalizada.*
- *Los dispositivos fotoelectrónicos superan a las películas en que pueden digitalizar la imagen al mismo tiempo que la adquieren.*



Ruido

Ruido provocado durante la adquisición de la imagen (3/5)

Tipos de ruido en las imágenes adquiridas con sensores fotoquímicos:

- **Ruido de grano de película (*film grain noise*):**
 - Provocado por los granos de plata que precipitan durante la exposición de la película.
 - Ruido es muy dominante cuando las imágenes son adquiridas con una velocidad alta.
 - Puede ser modelado mediante distribuciones de Poisson o gaussianas
- Ruido provocado por el **pollo** de los componentes ópticos o presente en el revelado de las películas.



Ruido

Ruido provocado durante la adquisición de la imagen (4/5)

Ruido en las imágenes adquiridas con sensores fotoelectrónicos

- **Ruido térmico (*thermal noise*):**

- Provocado por el calor originado por los componentes electrónicos.
- Suele ser modelado como ruido aditivo blanco gaussiano de media cero.

- **Ruido fotoelectrónico:**

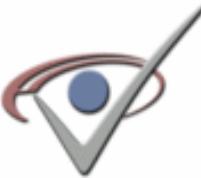
- Producido por la fluctuación aleatoria del número de fotones de la luz.
- Fluctuación baja: ruido modelado por una distribución de Poisson
- Fluctuación alta: ruido modelado como un proceso gaussiano ($\sigma = \sqrt{\mu}$).



Ruido

Ruido provocado durante la adquisición de la imagen (5/5)

- **Ruido de cuantificación:** otro tipo de ruido estrechamente relacionado con la adquisición
- Se produce cuando son usados insuficientes niveles de cuantificación para digitalizar la imagen.
- En tales situaciones aparecen contornos falsos.
- Se puede eliminar mediante una transformación en la escala de niveles de intensidad (por ejemplo, ecualización del histograma de frecuencias).



Ruido provocado durante la transmisión de la imagen (1/4)

- La **transmisión** de una imagen digital entre dos dispositivos remotos (una cámara y un ordenador, dos ordenadores o un monitor y un ordenador) puede **incrementar el ruido** de la imagen.
- Tipos de ruido
 - **Ruido aditivo**
 - **Ruido multiplicativo**
 - **Ruido impulsivo o ruido de sal y pimienta (*salt-pepper noise*)**.



Ruido

Ruido provocado durante la transmisión de la imagen (2/4)

- **Ruido aditivo**

- Ruido que es independiente de la señal de la imagen.

$$g(x, y) = f(x, y) + \eta(x, y)$$

donde la imagen original f y el ruido η son dos variables independientes.



Ruido

Ruido provocado durante la transmisión de la imagen (3/4)

- **Ruido multiplicativo**

- Ruido dependiente de la magnitud de la señal.
- Si la magnitud del ruido es mucho mayor que la de la señal entonces

$$\begin{aligned}g(x, y) &= f(x, y) + \eta(x, y)f(x, y) \\&= f(x, y)(1 + \eta(x, y)) \approx f(x, y)\eta(x, y)\end{aligned}$$

- Ejemplo:
 - Ruido de la pantalla de un televisor analógico.
 - Este ruido depende de la líneas del televisor: en el área de una línea, este ruido es máximo, y entre dos líneas es mínimo.



Ruido

Ruido provocado durante la transmisión de la imagen (4/4)

- **Ruido impulsivo o ruido de sal y pimienta (salt-pepper noise).**
 - También aparece con frecuencia durante la transmisión de la imagen
 - Hay puntos de la imagen que poseen unos niveles de intensidad que difieren bastante de los niveles de los puntos de su entorno.
 - Suele aparecer en forma de impulsos blancos o negros.

$$g(x, y) = \begin{cases} \eta(x, y) & \text{con probabilidad } p \\ f(x, y) & \text{con probabilidad } 1 - p \end{cases}$$

- Los impulsos pueden tener valores prefijados (v.g.: 0 ó 255) o pueden tener una distribución de probabilidad de “cola larga”



Ruido

Ruido provocado durante el procesamiento de la imagen (1/4)

- Se genera para valorar la **robustez** de un algoritmo en presencia de ruido.
- Se introduce ruido **impulsivo**, **aditivo** o **multiplicativo**.
- Se suele utilizar es el **ruido blanco** (*white noise*)
 - Tien una potencia espectral constante, es decir, su intensidad no decrece cuando aumenta la frecuencia.
 - Este ruido tiene de media cero y no está correlado.
 - Es una aproximación ideal del ruido que puede degradar las imágenes, porque su uso simplifica los cálculos.



Ruido

- Cualquier entidad, en imágenes, datos o resultados intermedios, que no es interesante para los objetivos del proceso principal
 - Signal-to-Noise ratio (SNR)

$$SNR = \frac{\sigma_s}{\sigma_n}, \left\{ \begin{array}{l} \sigma_s = \text{desviación típica de } f(i, j) \\ \sigma_n = \text{desviación típica de } n(i, j) \end{array} \right\}$$

- Mide lo fuerte que es el ruido con respecto a la señal de interés
- Se suele medir en dB

$$SNR_{dB} = 10 \log_{10} \frac{\sigma_s}{\sigma_n}$$



Modelo de degradación de una imagen

$$f(i, j) = h(i, j) * g(i, j) + n(i, j)$$

Imagen resultante	Función de degradación	Imagen original	Término de ruido
-------------------	------------------------	-----------------	------------------

- Consideraciones
 - Función de degradación = función identidad
 - Ruido independiente de la imagen
 - Aditivo
 - Multiplicativo



Generación de ruido en imágenes

- Modelos matemáticos para describir ruido
- Algoritmos para añadir ruido a las imágenes y simular el efecto del canal
 - Testear otros algoritmos para eliminar ruido
 - Demostrar que otros algoritmos son resistentes al ruido



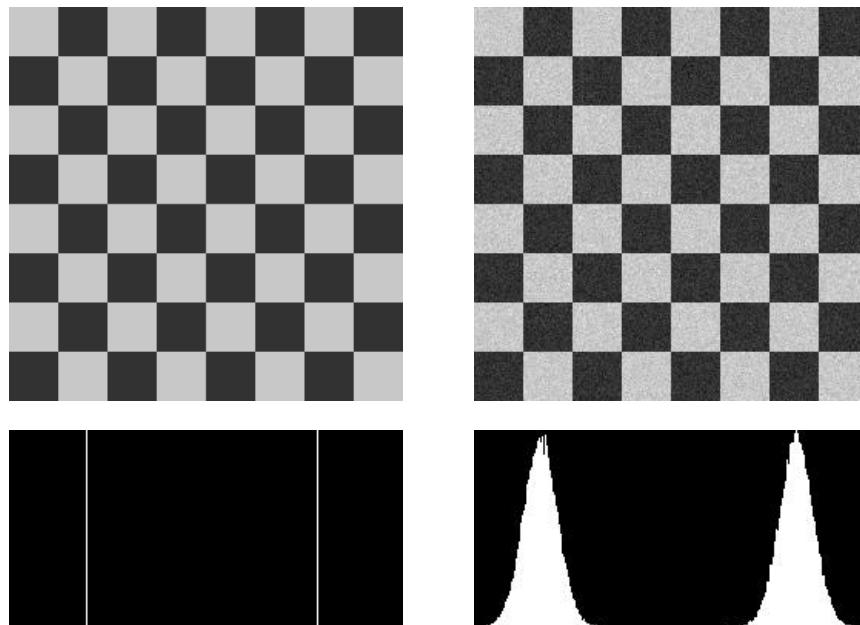
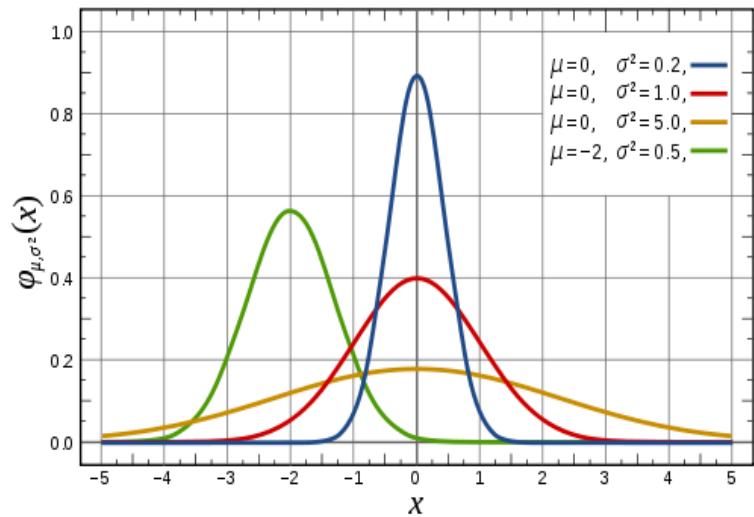
Modelos de ruido: Gaussiano

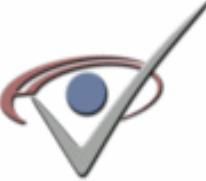
- En ausencia de información, $n(i,i)$ se suele modelar como ruido Gaussiano
 - Para cada la posición (i,j) de la imagen, $n(i,j)$ es una variable aleatoria con la siguiente función de densidad de probabilidad (PDF)

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2}$$



Modelos de ruido: Gaussiano





Modelos de ruido: Rayleigh

- PDF de ruido Rayleigh

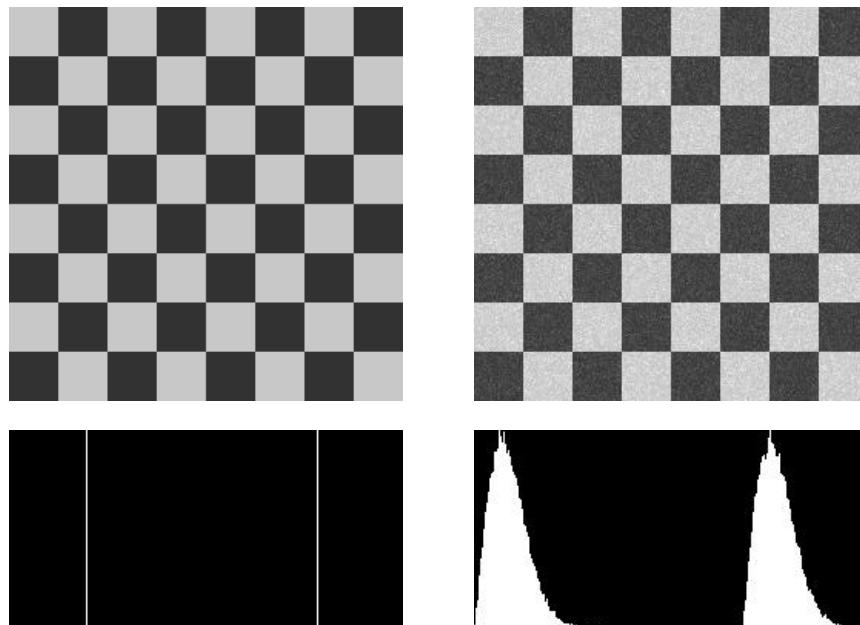
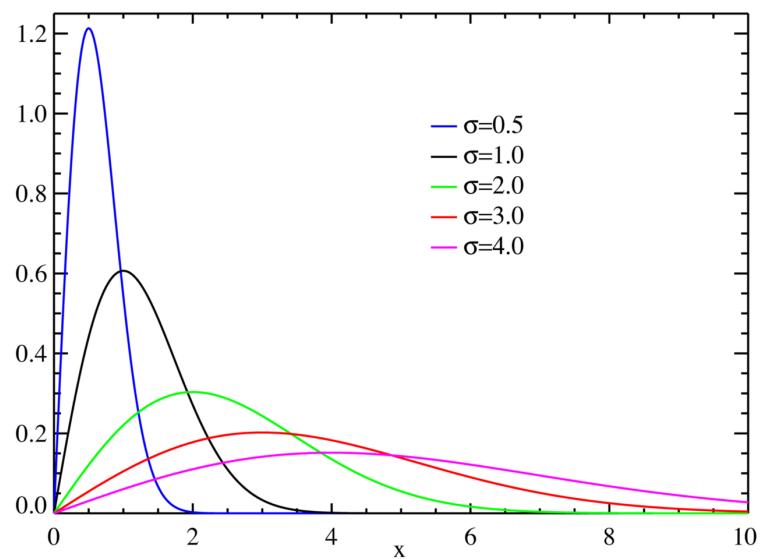
$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{(-z-a)^2/b} & \text{para } z \geq a \\ 0 & \text{para } z < a \end{cases}$$

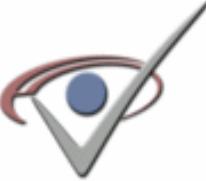
$$\mu = a + \sqrt{\pi b/4}$$

$$\sigma^2 = \frac{b(4-\pi)}{4}$$



Modelos de ruido: Rayleigh





Modelos de ruido: Gamma

- PDF de ruido Gamma

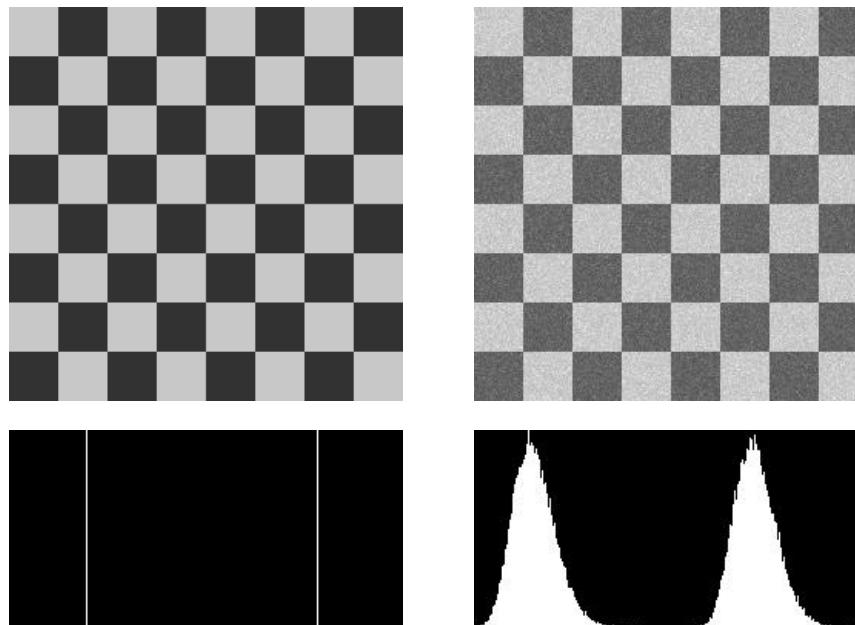
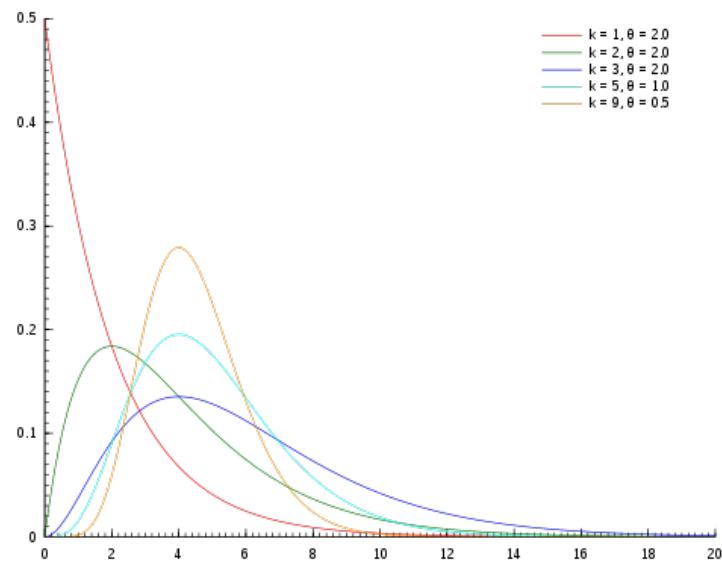
$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az} & \text{para } z \geq 0 \\ 0 & \text{para } z < 0 \end{cases}$$

$$\mu = \frac{b}{a}$$

$$\sigma^2 = \frac{b}{a^2}$$



Modelos de ruido: Gamma





Modelos de ruido: Exponencial

- PDF de ruido Exponencial

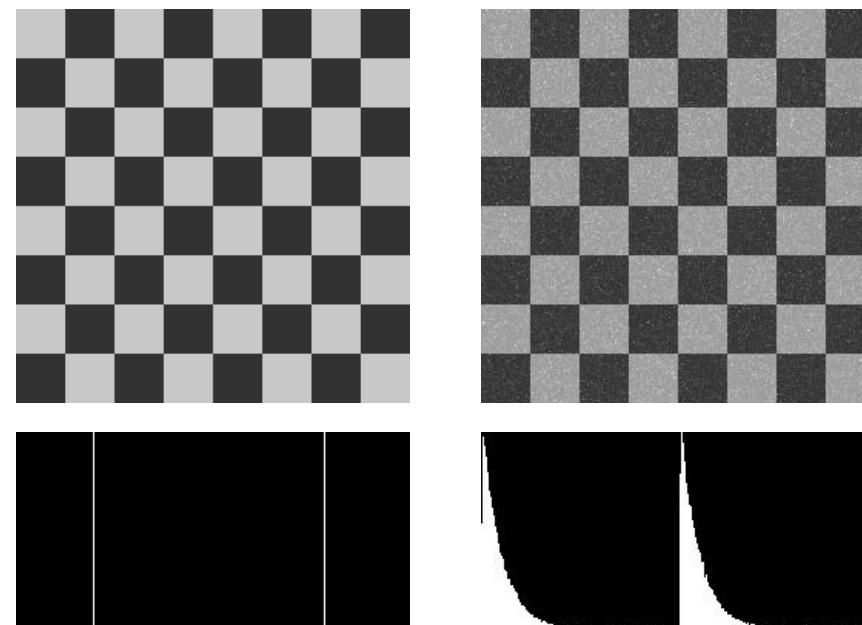
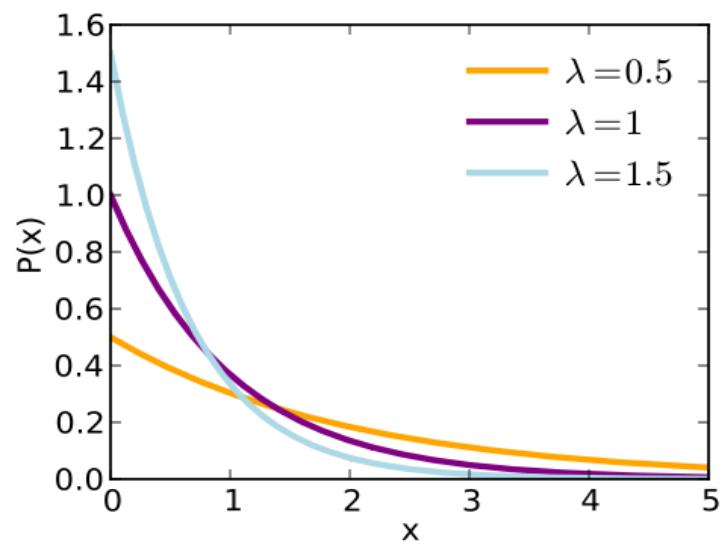
$$p(z) = \begin{cases} a e^{-az} & \text{para } z \geq 0 \\ 0 & z < 0 \end{cases} \quad a > 0$$

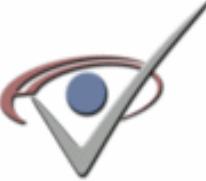
$$\mu = \frac{1}{a}$$

$$\sigma^2 = \frac{1}{a^2}$$



Modelos de ruido: Exponencial





Modelos de ruido: Uniforme

- PDF de ruido Uniforme

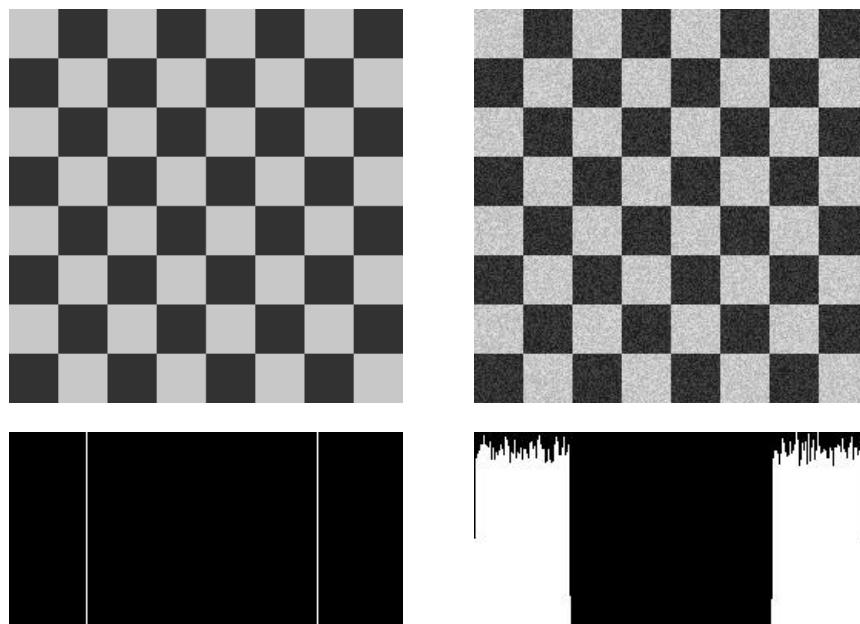
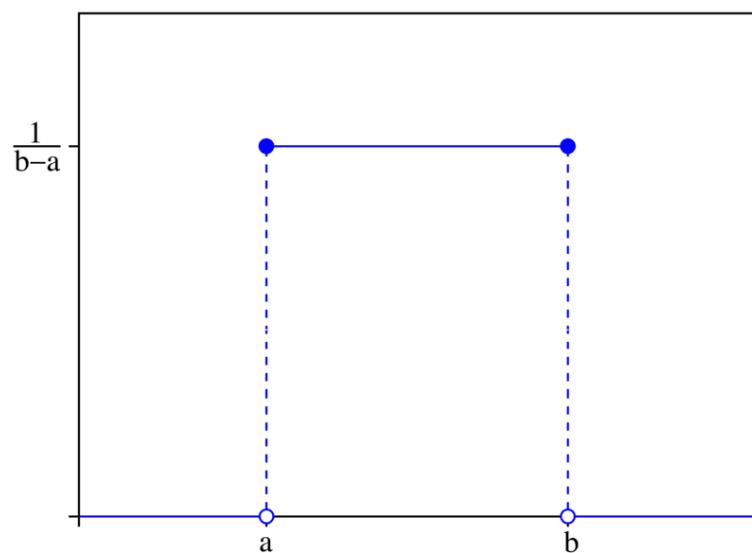
$$p(z) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq z \leq b \\ 0 & \text{en otro caso} \end{cases}$$

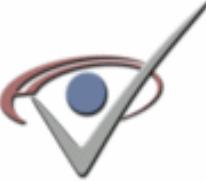
$$\mu = \frac{a+b}{2}$$

$$\sigma^2 = \frac{(b-a)^2}{12}$$



Modelos de ruido: Uniforme

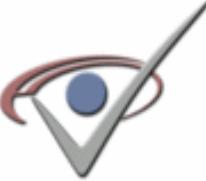




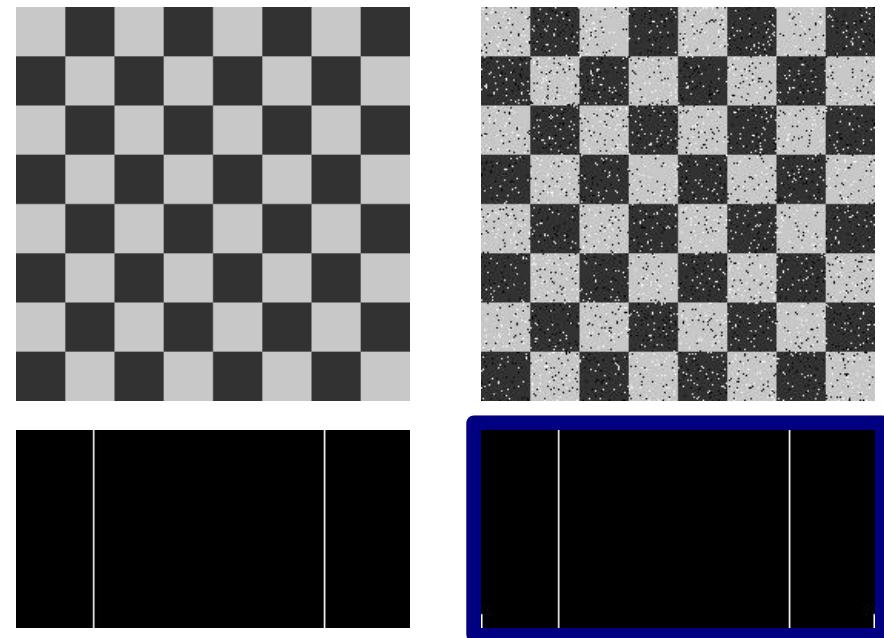
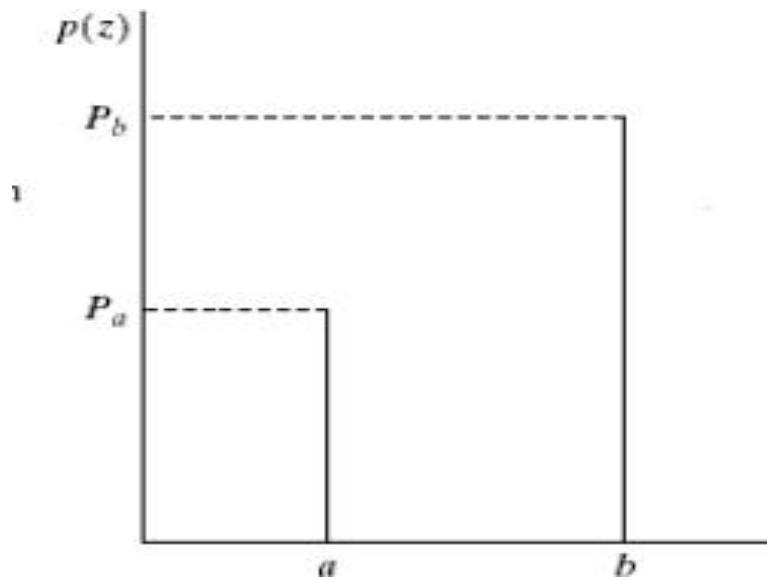
Modelos de ruido: Impulsional

- PDF de ruido Impulsional
 - Sal y pimienta

$$p(z) = \begin{cases} P_a & \text{para } z=a \\ P_b & \text{para } z=b \\ 0 & \text{en otro caso} \end{cases}$$



Modelos de ruido: Impulsional



Generación de ruido artificial en imágenes



- Ruido aditivo gaussiano ($\mu = 0$)

1. Seleccionar un valor para σ

Valores bajos generan menos ruido

2. Si las intensidades van en el rango $[0, N-1]$, calcular

$$p[k] = \frac{1}{\sqrt{2\pi}\sigma} e^{-k^2/2\sigma^2}, \quad k=0..N-1$$

3. Para cada píxel (i,j) de intensidad $g(i,j)$

1. Generar un número aleatorio q_1 en el rango $[0,1]$

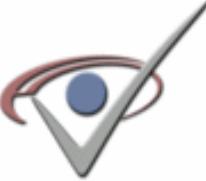
2. Calcular $n = \text{argmin}(q_1, p[k])$

3. Generar aleatoriamente un número q_2 en el conjunto $\{-1,1\}$ (signo)

4. $f'(i,j) = g(i,j) + q_2 n$

5. Calcular el valor en la nueva imagen con ruido

$$f(x,y) = \begin{cases} 0 & \text{si } f'(x,y) < 0 \\ N-1 & \text{si } f'(x,y) > N-1 \\ f'(x,y) & \text{en otro caso} \end{cases}$$



Filtros de suavizado

- Usos
 - Obtener una versión borrosa de la imagen original
 - Eliminar pequeños detalles de la imagen
 - Operación previa a extracción de características de interés en la imagen
 - Eliminar ruido



Filtros espaciales

- Lineales
 - Sea I una imagen de tamaño $M \times N$
 - Sea A un kernel de un filtro lineal de tamaño $N \times N$

$$I_A(i, j) = I * A$$



Filtros espaciales lineales

- Filtro de 3x3

$$A = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix}$$

- Coeficientes de A

- $a_{ij} \geq 0 \rightarrow$ suavizado
- $\sum a_{ij} > 1 \rightarrow$ la imagen filtrada será más clara
- $\sum a_{ij} = 0 \rightarrow$ la imagen filtrada conservará la misma intensidad
- $\sum a_{ij} < 0 \rightarrow$ la imagen filtrada será más oscura

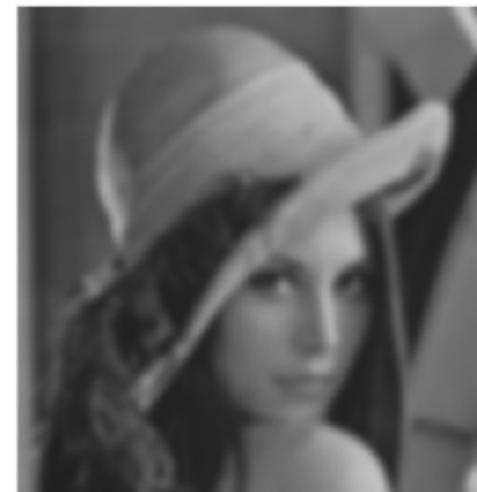


Filtro de Medias

- El filtrado espacial lineal paso bajo se basa en el promediado de los píxeles adyacentes al píxel que se evalúa.
- Se usa para quitar ruido y eliminar pequeños detalles de la imagen.
- El filtro paso bajo más simple que se puede diseñar es una matriz de 3x3 con todos los elementos a 1.



$$\frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$





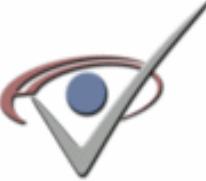
Filtro de Medias

Otros filtros de paso bajo:

$$h = \frac{1}{10} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

$$h = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$$

- Atenuan ruido de tipo gaussiano/uniforme/...
- Ruido impulsional sólo es difuminado
- La imagen se emborrona
 - Se agrava a medida que se aumenta el tamaño de la máscara



Filtro de medias

Ruido Gaussiano



Filtro 3x3



Filtro 7x7



Ruido Impulsional





Filtro de la Media Geométrica

- Definido como el producto de los valores de los píxeles dentro de la ventana, elevados a la potencia $1/nm$

$$Mg = \prod_{(x,y) \in W} [f(x, y)]^{1/nm}$$

donde nm es el número de píxeles en la ventana W de dimensiones $n \times m$

- Bueno para eliminar el ruido Gaussiano. Falla con el ruido impulsional



Filtro de la Media Armónica

- Definido como la división del tamaño de la ventana entre la suma de la inversa de los píxeles de la ventana

$$Mar = \frac{nm}{\sum_{(x,y) \in W} \frac{1}{f(x, y)}}$$

donde $n m$ es el número de píxeles en la ventana W de dimensiones $n \times m$

- Trabaja bien con ruido Gaussiano. Falla con ruido de pimienta pero trabaja bien con ruido de tipo sal



Imagen corrompida con ruido gaussiano



Filtrado con media geométrica



Filtrado con un pasa bajo espacial



Filtrado con media armónica



Filtro de la Media Contra-Armónica

- Definido por

$$Mca = \frac{\sum_{(x,y) \in W} f(x, y)^{R+1}}{\sum_{(x,y) \in W} f(x, y)^R}$$

- $R > 0 \rightarrow$ elimina el ruido tipo ‘pimienta’.
- $R < 0 \rightarrow$ elimina el ruido tipo ‘sal’.
- $R = 0 \rightarrow$ realiza la media aritmética.

donde: W es la ventana de dimensiones $n \times m$ tenida en cuenta
 R la tendencia del valor final



Imagen corrompida con ruido tipo sal



Imagen corrompida con ruido tipo pimienta



Filtrado con contra-armónico
 $R=-3$



Filtrado con contra-armónico
 $R=3$



Filtros espaciales lineales

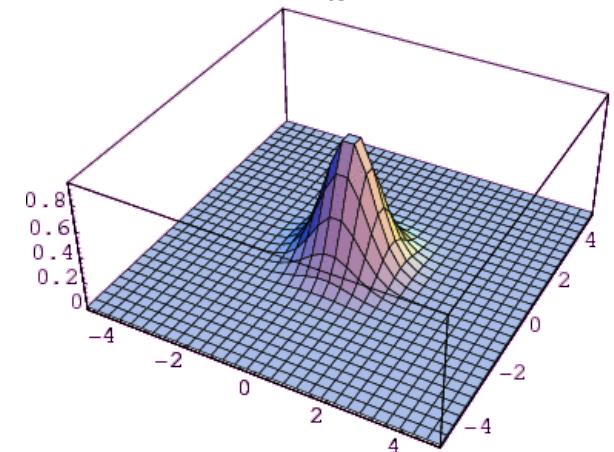
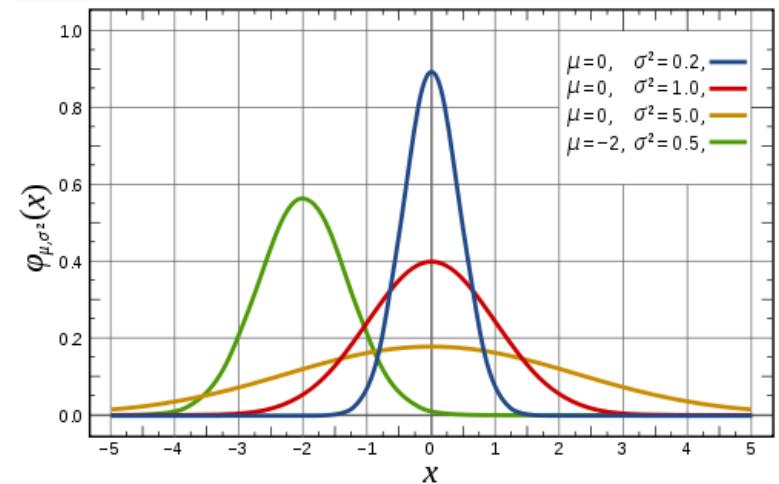
- Filtro Gaussiano

- Distribución Gaussiana en 1D

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$$

- Distribución Gaussiana en 2D

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

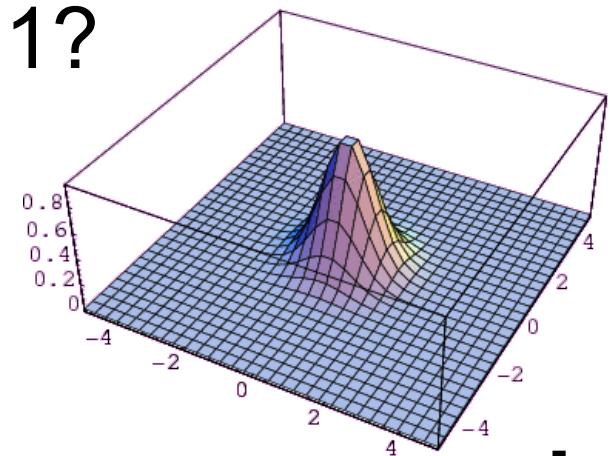




Ejercicio

- ¿Cómo se calcularía la máscara para un filtro Gaussiano de tamaño 5x5 y $\sigma = 1$?

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$



$$A = \begin{bmatrix} G(-2,2) & G(-1,2) & G(0,2) & G(1,2) & G(2,2) \\ G(-2,1) & G(-1,1) & G(0,1) & G(1,1) & G(2,1) \\ G(-2,0) & G(-1,0) & G(0,0) & G(1,0) & G(2,0) \\ G(-2,-1) & G(-1,-1) & G(0,-1) & G(1,-1) & G(2,-1) \\ G(-2,-2) & G(-1,-2) & G(0,-2) & G(1,-2) & G(2,-2) \end{bmatrix}$$



Filtro de Gauss

- Propiedades
 - Simetría rotacional. Tiene el mismo efecto en todas las direcciones

$$\sigma = 0,625 \text{ píxeles}$$

1	2	3	2	1
2	7	11	7	2
3	11	17	11	3
2	7	11	7	2
1	2	3	2	1



Filtro de Gauss

- Propiedades
 - Simetría rotacional. Tiene el mismo efecto en todas las direcciones
 - Un único lóbulo (pico)
 - El peso de los píxeles decrece con la distancia al centro
 - Cuanto más alejado está un píxel, menos significativo es
 - Preserva las bajas frecuencias y tiende a eliminar las altas.
 - El grado de filtrado es controlado por σ
 - A mayor $\sigma \rightarrow$ mayor suavizado y se tiene en cuenta los puntos más alejados de la media
- Filtro previo a la detección de bordes



Filtro de Gauss

- Ejemplos

Si $\sigma = 1$ píxeles

3	6	7	6	3
6	9	11	9	6
7	11	12	11	7
6	9	11	9	6
3	6	7	6	3

Si $\sigma = 1,6$ píxeles

7	8	9	8	7
8	10	11	10	8
9	11	12	11	9
8	10	11	10	8
7	8	9	8	7



Filtro Gaussiano

- Consigue un efecto de media ponderada del vecindario del pixel
 - Grado de suavizado está determinado por la desviación típica de la Gaussiana
 - Suavizado menos brusco
 - Preserva bordes mejor
- Linealmente separable



Filtro Gaussiano

Ruido Gaussiano



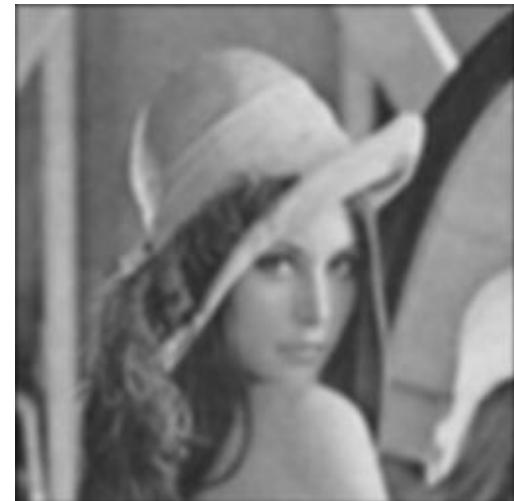
Ruido Impulsional



Filtro 7x7, $\sigma=1$



Filtro 7x7, $\sigma=2$



Filtros espaciales no lineales: filtros de orden

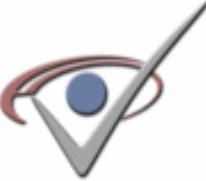
- Filtro de mediana

- Consiste en reemplazar el valor de un pixel en la imagen por la mediana del vecindario

$$J(i, j) = \underset{(s, t) \in \mathcal{N}(i, j)}{\text{median}}(I(s, t))$$

←
Vecindario del punto (i,j)

- La mediana es el valor central de un conjunto ordenado de puntos
- Ventajas
 - Elimina ruido impulsional
 - No afecta a los bordes
 - Puede ser aplicado iterativamente y no disminuye la calidad de la imagen



Filtro de mediana

Ruido Gaussiano



Ruido Impulsional

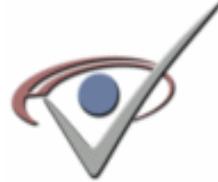


Filtro 3x3



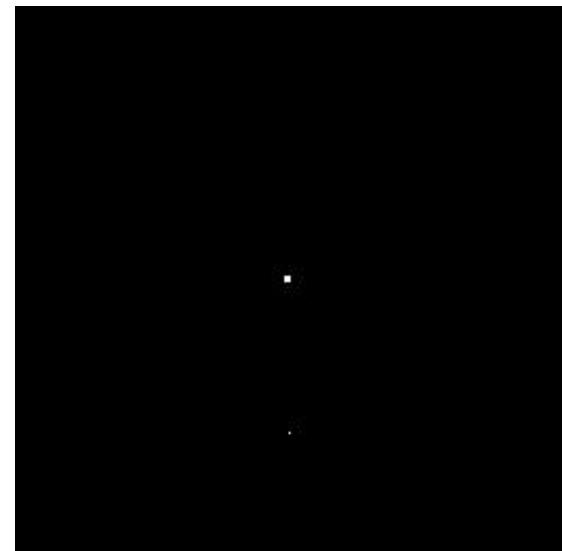
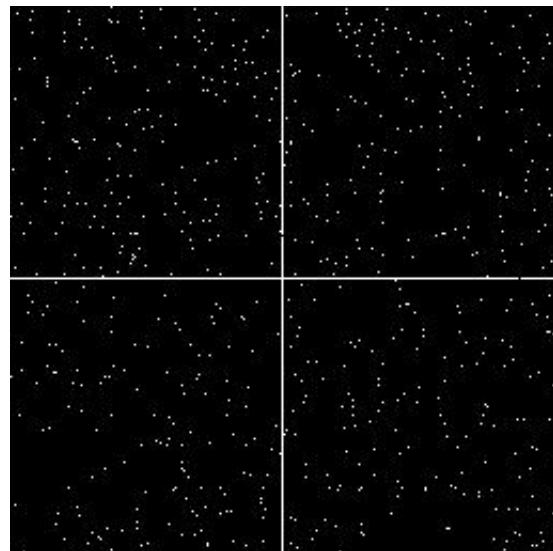
Filtro 7x7





Filtro de mediana

- Desventajas
 - Problema en imágenes con líneas finas o esquinas





Ejercicio

- Definir un filtro de mediana de tamaño 3x3 que no elimine líneas de 1 pixel de grosor

Imagen de prueba

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

Filtros espaciales no lineales: filtros de orden

- Filtro de máximos

$$J(i, j) = \max_{(s, t) \in \mathcal{N}(i, j)} (I(s, t))$$

- Filtro de mínimos

$$J(i, j) = \min_{(s, t) \in \mathcal{N}(i, j)} (I(s, t))$$

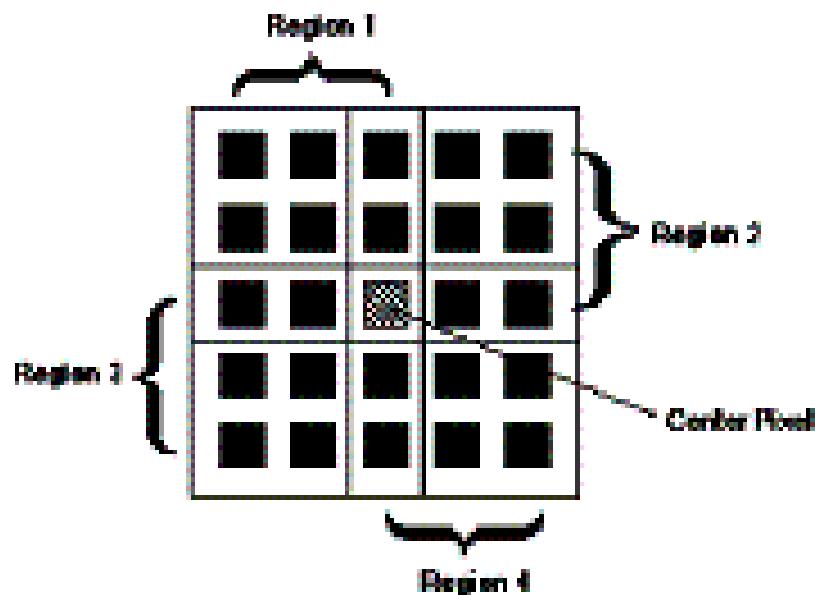
- Filtro del punto medio

$$J(i, j) = \frac{1}{2} \left(\max_{(s, t) \in \mathcal{N}(i, j)} (I(s, t)) + \min_{(s, t) \in \mathcal{N}(i, j)} (I(s, t)) \right)$$



Filtro de Kuwahara

- Tiene como objetivos suavizar imágenes sin distorsionar los detalles y si es posible la posición de los bordes. Filtro de suavizado perservando bordes.
- Se parte de una ventana de tamaño $J=K=4L+1$ con L entero. La ventana se parte en cuatro regiones



- Para cada una de las regiones se mide el valor medio de gris y la varianza (desviación típica)
- El valor de salida asociado al pixel central de la ventana es el valor medio de la región que tiene una varianza más pequeña

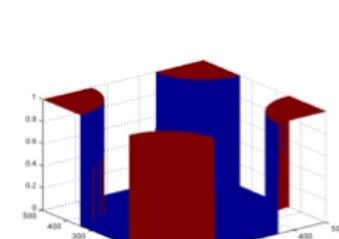
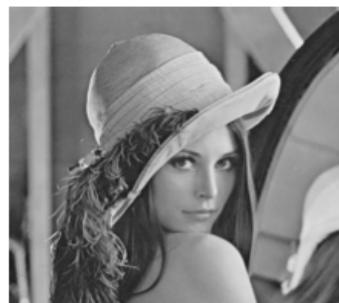
$$m_b = \frac{1}{N} \sum_{m,n \in R} a(m,n) \quad s_b = \sqrt{\frac{1}{N-1} \sum_{m,n \in R} (a(m,n) - m_b)^2}$$



Filtros en frecuencia

- Filtro paso bajo
 - Se establece una frecuencia de corte
 - Se filtran las frecuencias superiores a la frecuencia de corte
 - Filtro ideal produce “rebotes” en los bordes
 - Filtro Butterworth ofrece mejores resultados
 - Efecto similar a un suavizado gaussiano en el dominio espacial

$$H(u, v) = \begin{cases} 1 & \text{si } \sqrt{u^2 + v^2} < f_c \\ 0 & \text{si } \sqrt{u^2 + v^2} > f_c \end{cases}$$

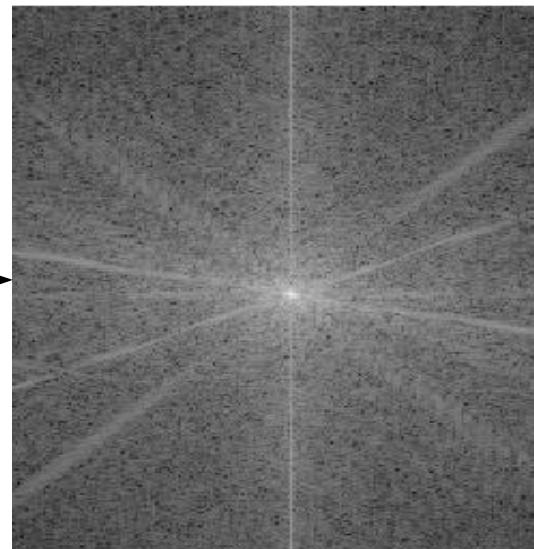




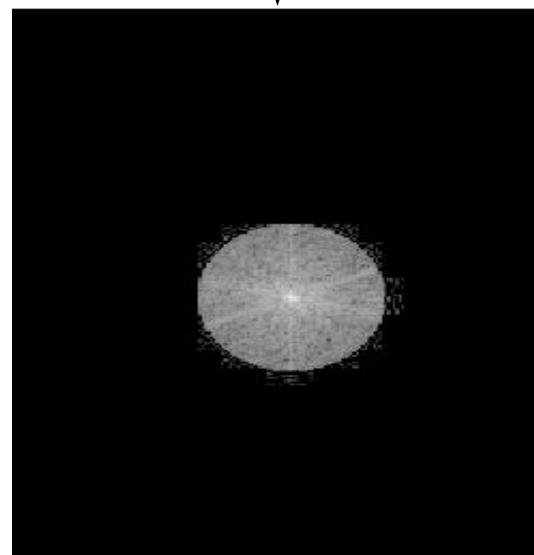
Filtros en frecuencia



DFT →



Filtro en Frecuencia



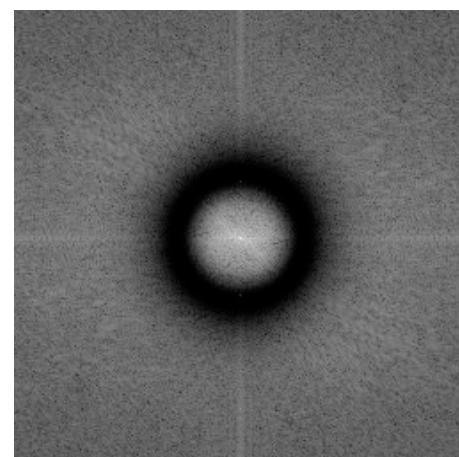
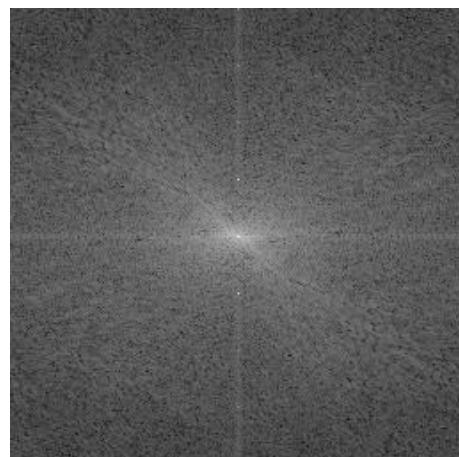
DFT⁻¹ →





Filtros en frecuencia

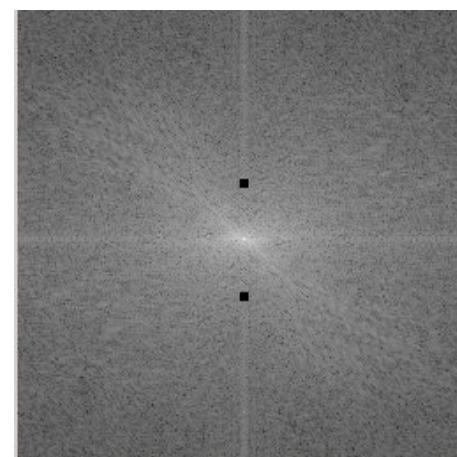
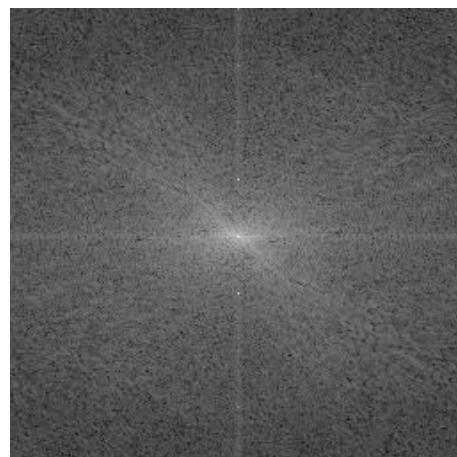
- Filtro banda eliminada
 - Elimina las frecuencias comprendidas entre dos frecuencias umbrales
 - Útil para eliminar ruido en una determinada banda de frecuencias preservando el resto de frecuencias





Filtros en frecuencia

- Filtro Notch
 - Elimina una frecuencia específica del espectro
 - Sirve para eliminar ruido en una determinada frecuencia preservando el resto de frecuencias





Realce

- Objetivos
 - Subrayar los detalles finos en una imagen
 - Mejorar los detalles que han sido difuminados
- Metodología
 - Realzar los bordes y otras discontinuidades
 - Desenfatizar áreas donde la intensidad varía de forma suave



Realce en el dominio espacial

- Suavizado en el dominio espacial
 - Media ponderada de los píxeles en un vecindario
 - Convolución
 - Integración
- Realce en el dominio espacial
 - Realzar bordes/discontinuidades
 - La respuesta del operador de derivación es proporcional al grado de discontinuidad de la imagen
 - Derivadas



Derivadas en la imagen

- Operador de primera derivada
 - 0 en zonas con valores constantes de intensidad
 - Distinto de 0 en un cambio brusco de intensidad
 - Distinto de 0 a lo largo de un cambio gradual de intensidad
- Operador de segunda derivada
 - 0 en zonas con valores constantes de intensidad
 - Distinto de 0 en un cambio brusco de intensidad
 - 0 a lo largo de un cambio gradual de intensidad





Aproximación a las derivadas

- Primera derivada

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

- Segunda derivada

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

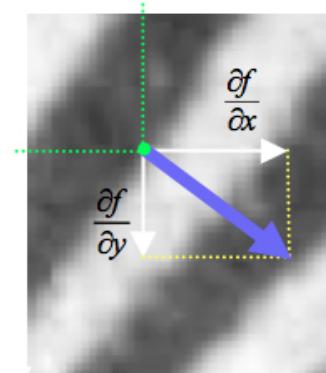


Aproximación a las derivadas

Operador Gradiente. El gradiente de una función continua $f(x,y)$ es un vector y su magnitud expresa el ratio de cambio por unidad de distancia en la dirección del vector. El objetivo de los operadores gradiente es detectar cambios en los niveles de gris que tienen lugar en zonas o regiones reducidas. El gradiente y la magnitud se expresan por:

$$\text{grad}[f(x, y)] = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]^T$$

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



A la magnitud del gradiente se la conoce como *fuerza del borde* y junto con la *orientación del gradiente* constituyen datos cruciales en la extracción de este tipo de características.

Sustituyendo las diferencias obtenidas en la ecuación anterior por las derivadas se obtiene el gradiente de la imagen:

$$\text{grad}[f(x, y)] = \left[(f(x+1, y) - f(x, y))^2 + (f(x, y+1) - f(x, y))^2 \right]^{1/2}$$

En algunas ocasiones se aproxima por:

$$\text{grad}[f(x, y)] \approx |f(x+1, y) - f(x, y)| + |f(x, y+1) - f(x, y)|$$

Propiedades de las derivadas en la imagen



- Primera derivada
 - Bordes más gruesos
 - Mejor respuesta ante escalones de intensidad graduales
- Segunda derivada
 - Bordes más finos
 - Mejor respuesta a detalles finos
 - Líneas finas
 - Puntos aislados (incluso ruido)
 - Doble borde en escalones de intensidad graduales

Operador Segunda Derivada - Laplaciano



- Máscara discreta de la segunda derivada
- Máscara isotrópica
 - Respuesta independiente a la dirección de las discontinuidades de la imagen
- Laplaciano
 - Operador de derivada isotrópico más sencillo

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



Laplaciano

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

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

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

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

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

Resultados isotrópicos 90°

Resultados isotrópicos 45°



Aplicación del Laplaciano

1. Convolución de la imagen con la máscara

1	1	1
1	-8	1
1	1	1

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

- Escalar resultado

2. Añadir el resultado a la imagen original

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) & \text{coeficiente central} < 0 \\ f(x, y) + \nabla^2 f(x, y) & \text{coeficiente central} > 0 \end{cases}$$

- Escalar resultado



Laplaciano

Laplaciano

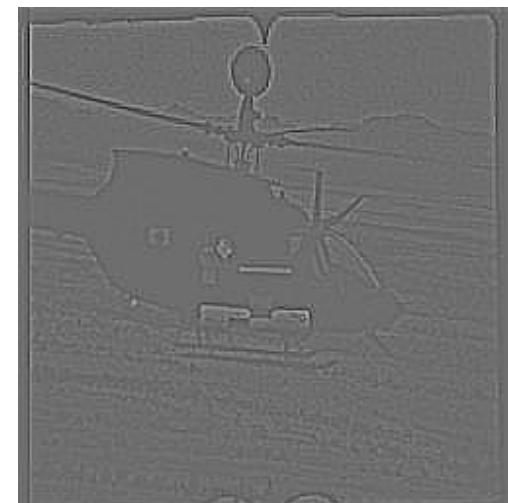


Imagen + laplaciano escalada

Laplaciano escalado



Algoritmos de realce

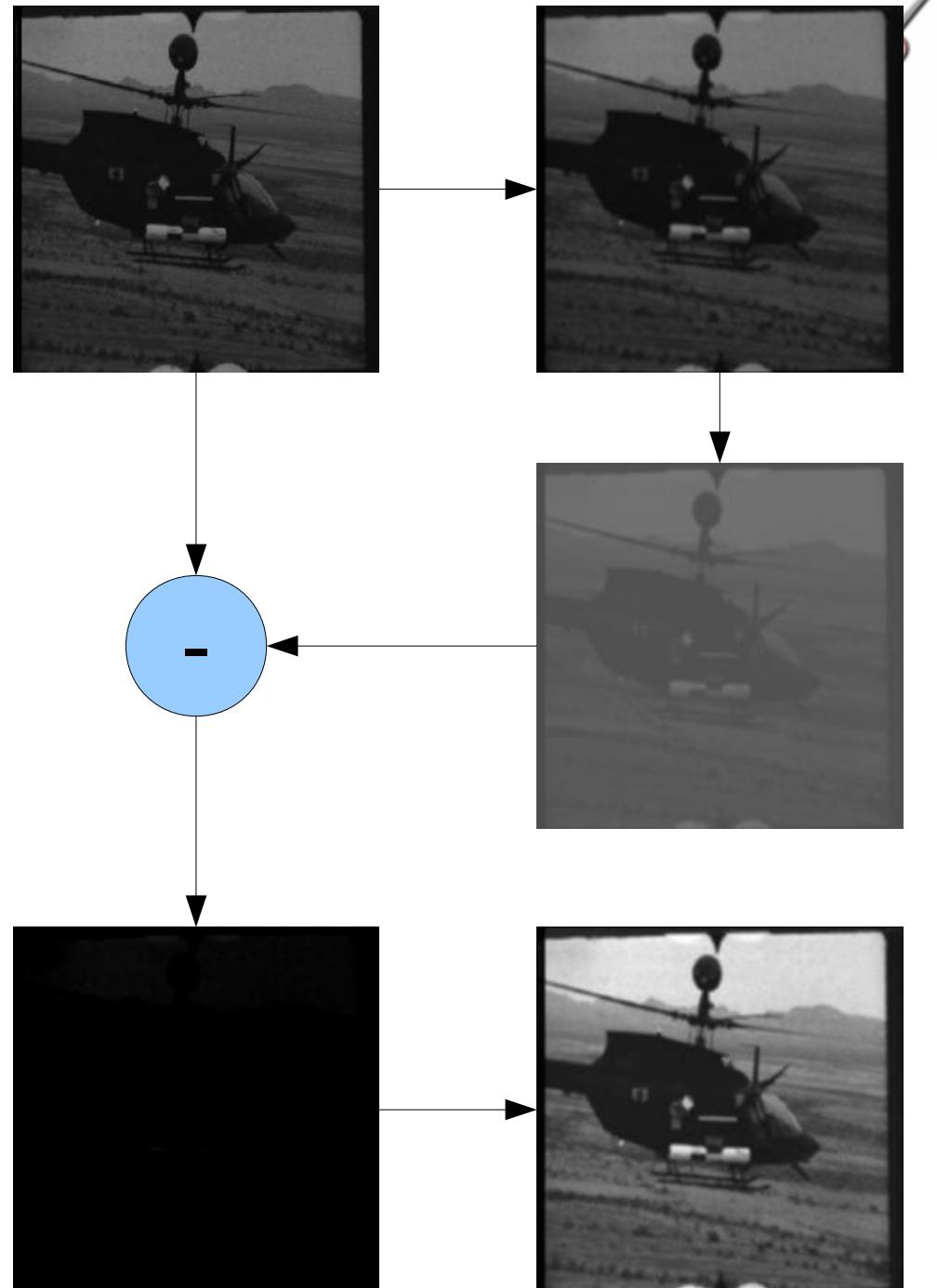
- Unsharp masking
 - Basado en sustraer una versión suavizada de la imagen a la propia imagen

$$g(x, y) = f(x, y) - f_{smooth}(x, y)$$

- Existen varias versiones de este algoritmo

Unsharp masking

1. Aplicar un filtro paso bajo a la imagen original
 - Filtro de medias 3x3
2. Aplicar una compresión de histograma a la imagen suavizada
3. Sustraer la imagen comprimida a la imagen original
4. Aplicar un estiramiento de histograma a la imagen resultante





Algoritmos de realce

- High boost
 - Generalización del algoritmo Unsharp Masking

$$g_{hb}(x, y) = Af(x, y) - f_{smooth}(x, y), \quad A \geq 0$$

$$g_{hb}(x, y) = (A - 1)f(x, y) + f(x, y) - f_{smooth}(x, y)$$

$$g_{hb}(x, y) = (A - 1)f(x, y) + f_{sharpen}(x, y)$$

- Cuanto mayor sea A, más información de la imagen original se retiene



High boost

- Puede calcularse mediante máscaras de convolución
 - Si $A = 0 \rightarrow$ Laplaciano
 - Si $A < 0 \rightarrow$ Bordes
 - Si $A \geq 1 \rightarrow$ Realce

-1	-1	-1
-1	$A + 8$	-1
-1	-1	-1



High boost



$A = 1$



$A = 3$



$A = 0$

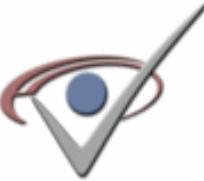


$A = 1$

+ Estiramiento de histograma

$A = 3$





Realce en frecuencia

- Filtro paso alto

- Ideal

$$G(k, l) = F(k, l) H(k, l)$$

$$H(k, l) = \begin{cases} 0 & \text{si } |F(k, l)| < F_0 \\ 1 & \text{si } |F(k, l)| \geq F_0 \end{cases} \quad F_0 = \text{frecuencia de corte}$$

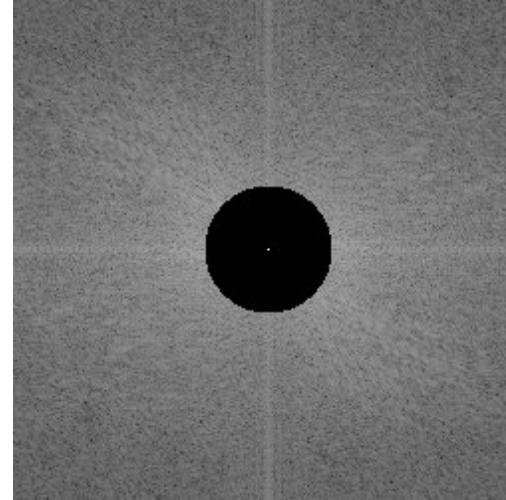
- Butterworth

$$H(k, l) = \frac{1}{1 + [F_0 / |F(k, l)|]^{2n}}$$



Filtro paso alto ideal

Filtrado en frecuencia



Altas frecuencias

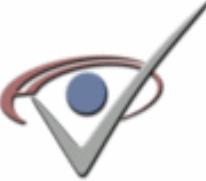


Imagen original + altas frecuencias



Estiramiento histograma





Filtro paso alto Butterworth

Ideal



Butterworth





Énfasis de altas frecuencias

- Multiplicar componentes de alta frecuencia por una constante
- Sumar un offset para preservar frecuencias bajas

$$H_{hfe}(k, l) = a + bH_{hp}(k, l) \quad a \geq 0, \quad b > a$$





Filtrado homomórfico

- Las imágenes pueden representadas como el producto de la iluminación y la reflectancia

$$f(i, j) = il(i, j)r(i, j)$$

- Iluminación → componentes de baja frecuencia
- Reflectancia → componentes de alta frecuencia
- Objetivo
 - Filtrar de forma distinta ambas componentes



Filtrado homomórfico

$$f(i, j) = il(i, j)r(i, j)$$

- La transformada de Fourier del producto de dos funciones no es separable

$$F(k, l) \neq IL(k, l)R(k, l)$$

- Se define una función intermedia

$$\begin{aligned}z(i, j) &= \ln(f(i, j)) = \ln(il(i, j)r(i, j)) \\z(i, j) &= \ln(il(i, j)) + \ln(r(i, j))\end{aligned}$$



Filtrado homomórfico

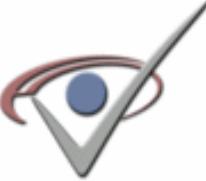
- Su TF es

$$Z(k, l) = TF(\ln(il(i, j))) + TF(\ln(r(i, j)))$$
$$Z(k, l) = IL(k, l) + R(k, l)$$

- Aplicamos un filtro a ambas componentes y calculamos su TF inversa

$$S(k, l) = H(k, l)IL(k, l) + H(k, l)R(k, l)$$

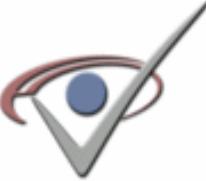
$$s(i, j) = TF^{-1}[H(k, l)IL(k, l)] + TF^{-1}[H(k, l)R(k, l)]$$
$$s(i, j) = il'(i, j) + r'(i, j)$$



Filtrado homomórfico

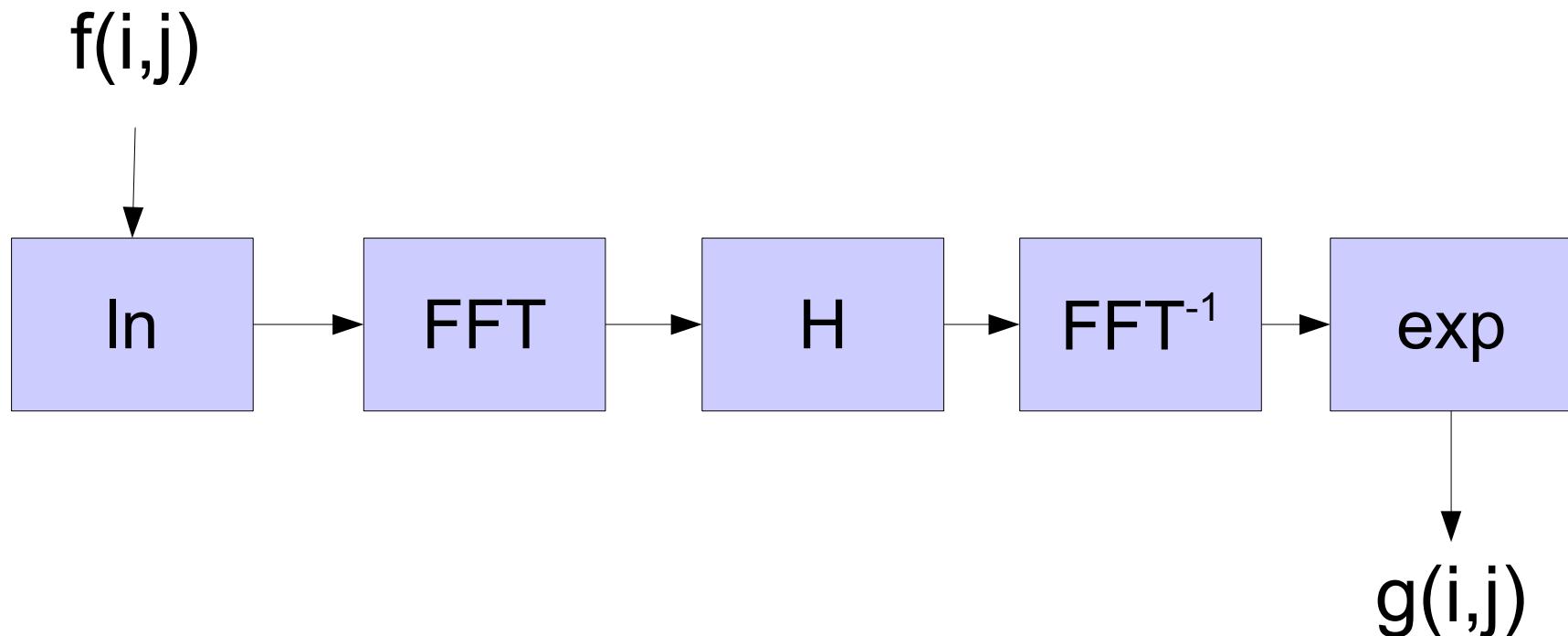
- Como z se obtuvo mediante el logaritmo, aplicamos la transformación inversa (exponencial)

$$g(i, j) = e^{s(i, j)} = e^{il'(i, j) + r'(i, j)} = e^{il'(i, j)} e^{r'(i, j)}$$
$$g(i, j) = il_0(i, j) r_0(i, j)$$



Filtrado homomórfico

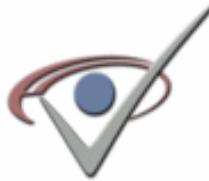
- Gráficamente





Filtrado homomórfico





Operadores morfológicos

Referencias

- R. C. Gonzalez and R. E. Woods. Digital Image Processing. Addison-Wesley Longman Publishing Co., Inc., 2001.
- M. Sonka, V. Hlavac, and R. Boyle. Image Processing: Analysis and Machine Vision. Thomson-Engineering, 1998.



Operadores morfológicos

- Operadores no lineales que actúan en la forma de los objetos
- Principales usos
 - Preprocesado de imágenes (eliminación de ruido, simplificación de formas)
 - Realce de la estructura de los objetos (esqueletización, adelgazamiento/engrosamiento líneas)
 - Segmentación de objetos
 - Descripción cuantitativa de los objetos (área, perímetro)

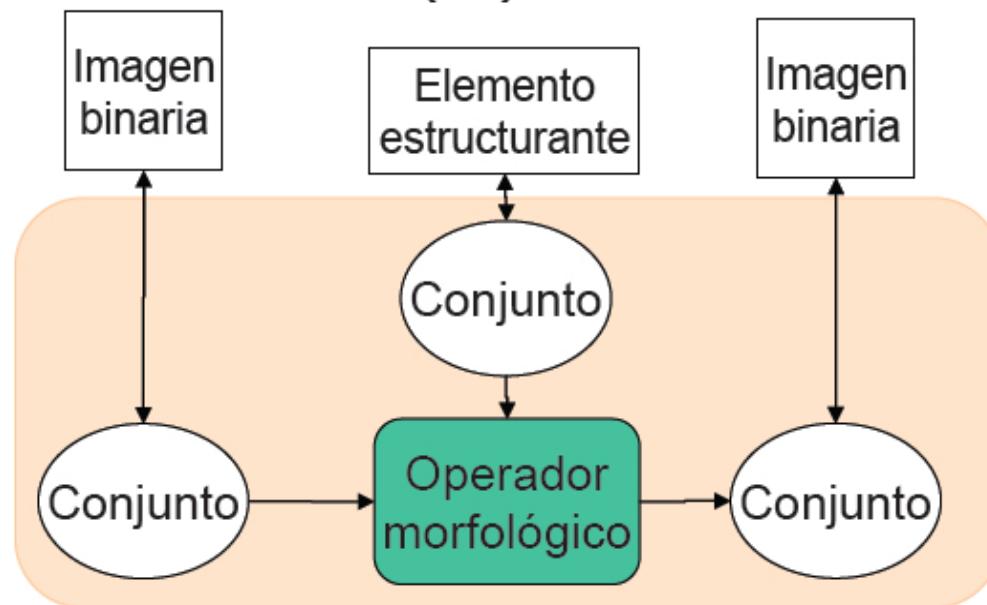


Procesado Morfológico

- Los fundamentos del análisis y procesado morfológico se basan en el álgebra de conjuntos y en la topología

Tres elementos:

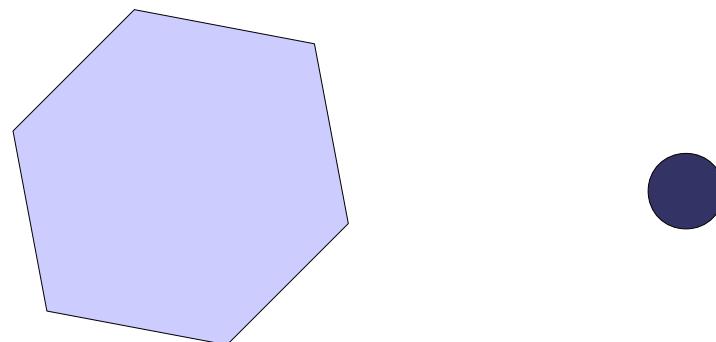
- Conjuntos (\Leftrightarrow Imágenes)
- Operadores Morfológicos (dilatación, erosión, apertura/cierre)
- Elementos Estructurantes (EE)





Procesado Morfológico

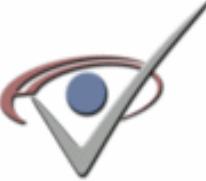
- Desde un punto de vista geométrico
 - Comparación de los objetos a analizar con otro objeto de forma conocida denominado elemento estructurante





Procesado Morfológico

- **Imagen:** Subconjunto del espacio 2D de todos los enteros \mathbb{Z}^2
 - Un punto de la imagen se representa con un par de coordenadas enteras
 - El origen de coordenadas $(0,0)$ se encuentra en una esquina de la imagen
 - Puntos que representan un objeto → conjunto X
 - $I(i,j) = 1$
 - Puntos que representan fondo → conjunto X^c
 - $I(i,j) = 0$



Imagen

1	0	0	0
1	0	0	0
0	1	1	0
0	1	0	0
(0,0) → 0	1	0	0

$$X = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$$



Elemento estructurante

- Matriz binaria de tamaño $k \times k$
 - Conjunto de 1's determinan la forma del elemento estructurante

1	1	1
1	1	1
1	1	1

0	1	0
1	1	1
0	1	0

0	0	0
1	1	1
0	0	0

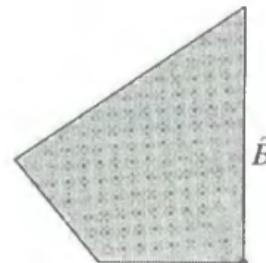
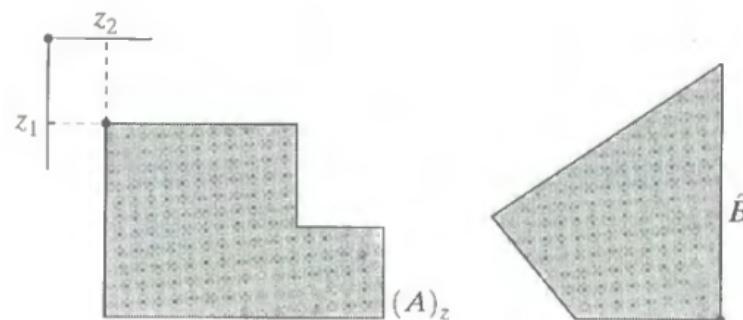
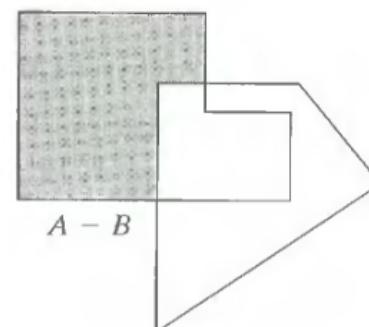
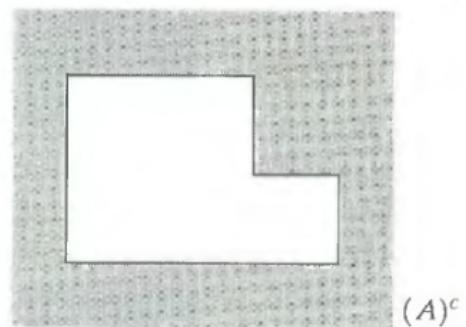
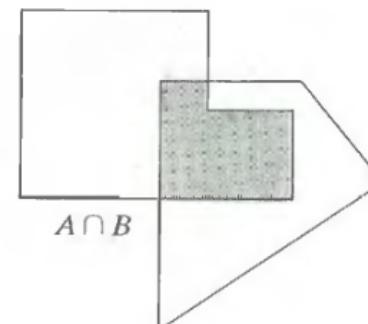
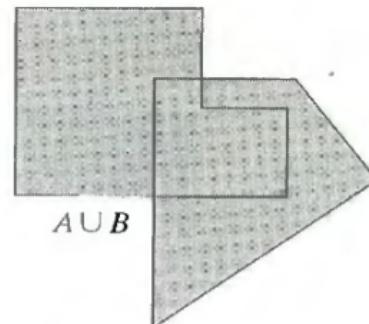
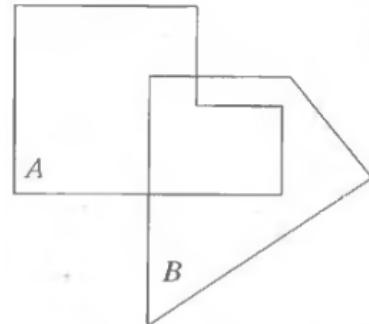


Teoría de conjuntos

- Unión $C = A \cup B$
 - Intersección $D = A \cap B$
 - Complemento $A^c = \{w \mid w \notin A\}$
 - Diferencia $A-B = \{w \mid w \in A, w \notin B\} = A \cap B^c$
-
- Reflexión $\hat{A} = \{w \mid w = -a, \forall a \in A\}$
 - Traslación por el vector $z = (z_1, z_2)$
 $(A)_z = \{c \mid c = a + z, \forall a \in A\}$



Teoría de conjuntos





Teoría de conjuntos

- Traslación al punto $z = (1, 0)$

$$(A)_z = \{c \mid c = a + z, \quad \forall a \in A\}$$

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

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

$$\begin{aligned} A &= \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\} \\ (A)_z &= \{(2,0), (2,1), (2,2), (3,2), (1,3), (1,4)\} \end{aligned}$$



Elemento Estructurante

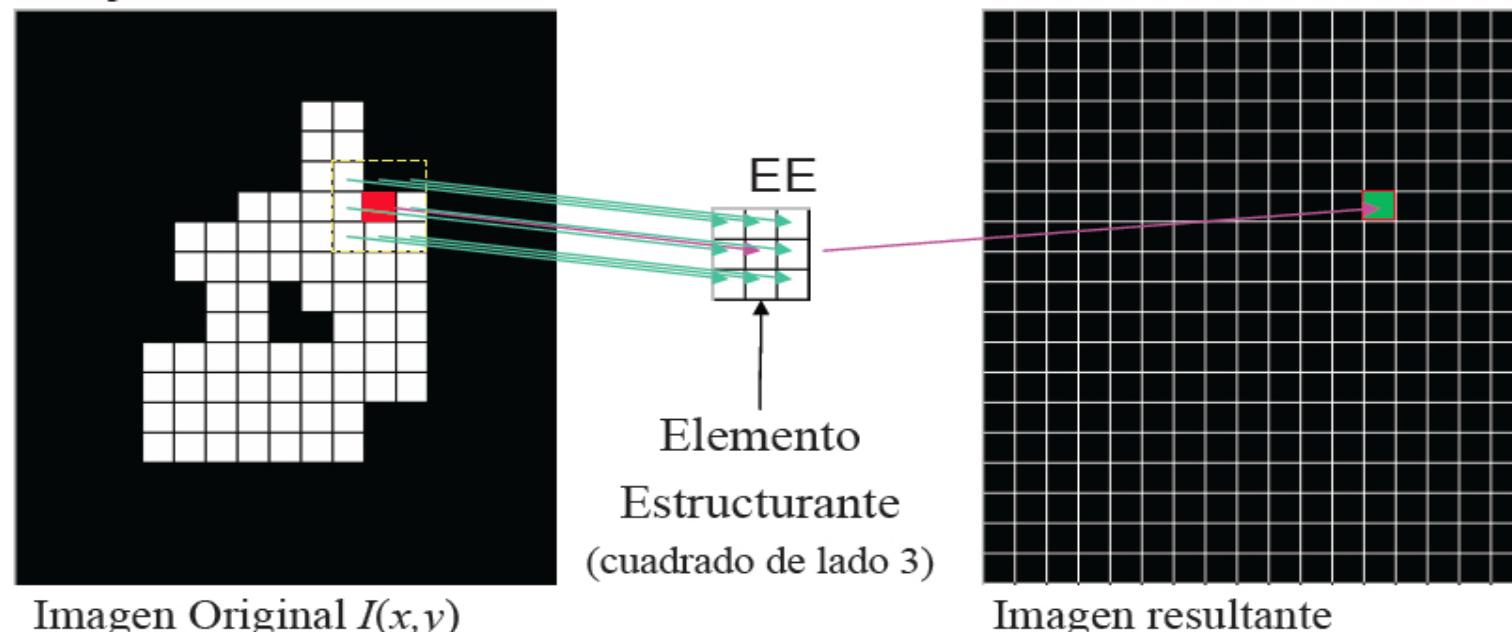
- Idea:
 - Examinar la estructura geométrica de una imagen usando como sonda un patrón de ajuste que se denomina elemento estructurante (E.E.).
 - La forma y tamaño del EE caracteriza las formas de los objetos presentes en una imagen y permiten determinar la estructura de un conjunto X.
 - El EE se maneja de modo análogo a una máscara de convolución, aunque ahora trabajamos con operaciones sobre conjuntos y la convolución se basa en operaciones aritméticas.



Elemento Estructurante

- **Ajuste:**

- El EE puede tener cualquier tamaño y forma (horizontal, vertical, cuadrado, ...).
- Su centro se sitúa en cada píxel de la imagen original, aplicando la operación morfológica sobre los puntos situados bajo el E.E.



Operadores Morfológicos: **DILATACIÓN**



Dilatación de la imagen A por el EE B:

$$A \oplus B = \{p \in \mathcal{E}^2 : p = \underset{\wedge}{a} + b \mid a \in A, b \in B\}$$

$$A \oplus B = \{p \mid (B)_p \cap A \neq \emptyset\}$$

Reflexión de B con respecto a su origen y un desplazamiento p

- Suponiendo que el EE contiene al origen y es simétrico (en este caso el EE y su reflexión son iguales). La salida de la dilatación es el conjunto de puntos barridos por el centro del EE mientras algún punto de B tocaba a algún punto de A.

Operadores Morfológicos: **DILATACIÓN**



Propiedades:

- La operación de dilatación hace que los objetos se expandan.
 - Añade todos los puntos del fondo que tocan el borde de un objeto. La dilatación es extensiva
- La cantidad y la forma en que crezcan depende del elemento estructurante.
 - Un elemento estructurante con forma simétrica hará que la expansión sea igual en todas las direcciones.
- Rellena entrantes en los que no quepa el EE (pequeños agujeros y bahías)



Dilatación

$$A \oplus B = \{p \in \mathcal{E}^2 : p = a + b \mid a \in A, b \in B\}$$

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

1	1
---	---

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

$$A = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$$

$$B = \{(0,0), (1,0)\}$$

$$A \oplus B = \{(1,0), (1,1), (1,2), (0,3), (0,4), (2,0), (2,1), (2,2), (3,2), (1,3), (1,4)\}$$



Dilatación

$$A \oplus B = \{p \mid (\hat{B})_p \cap A \neq \emptyset\}$$

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

1	1
---	---

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

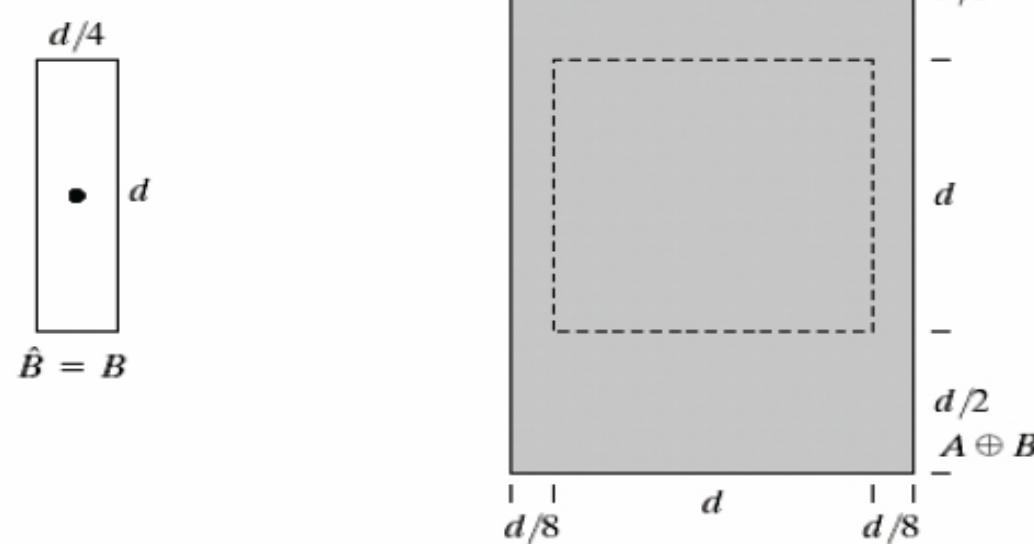
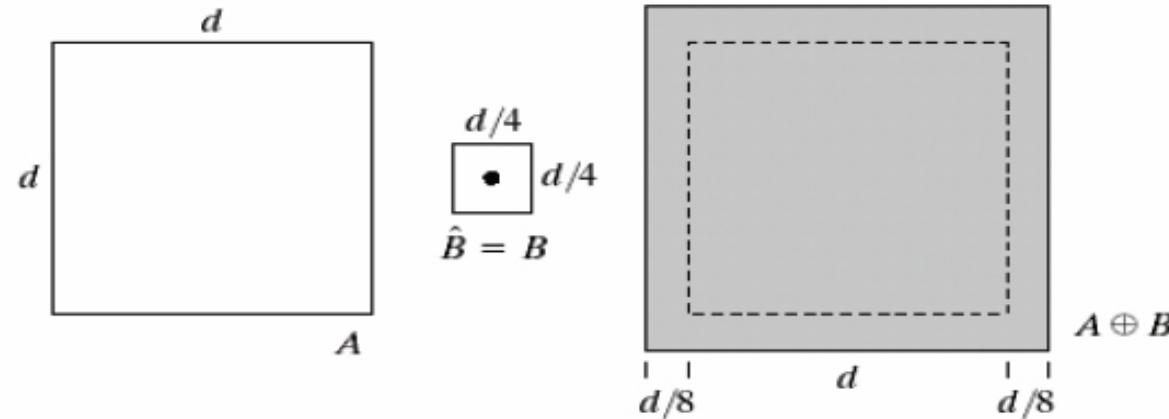
$$A = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$$

$$B = \{(0,0), (1,0)\} \quad \hat{B} = \{(0,0), (-1,0)\}$$

$$A \oplus B = \{(1,0), (1,1), (1,2), (0,3), (0,4), (2,0), (2,1), (2,2), (3,2), (1,3), (1,4)\}$$



Dilatación





Dilatación

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



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

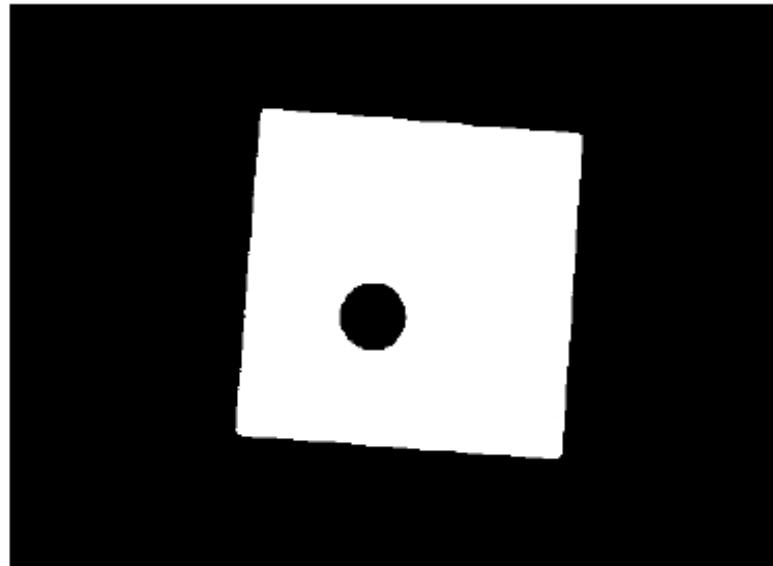


EE
cuadrado

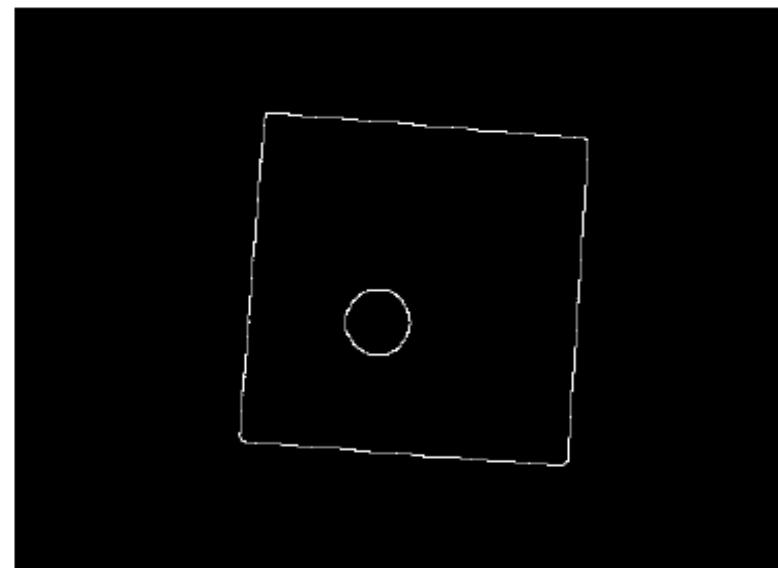




Dilatación



Dilatación y substracción





Operadores Morfológicos: **EROSIÓN**

Erosión de la imagen A por el EE B:

$$A \ominus B = \{p \in \mathcal{E}^2 : p + b \in A, \forall b \in B\}$$

$$A \ominus B = \{p \mid (B)_p \subseteq A\}$$

Los puntos p, tales que B trasladado p, están contenidos en A.

- Suponiendo que el EE contiene al origen y es simétrico: la salida de la erosión es el conjunto de puntos barridos por el centro del EE mientras se cumpla que todos los puntos de B estaban contenidos en A.

Operadores Morfológicos: ***EROSIÓN***



Propiedades:

- La operación de dilatación hace que los objetos se contrayan. Reduce el tamaño del Objeto.
 - Elimina grupos de píxeles donde el EE no cabe.
- Elimina elementos en los que no quepa el EE (pequeñas islas y protuberancias)
- ¿Es la erosión inversa de la dilatación?
 - No, pero están relacionadas
$$(A \ominus B)^c = (A^c \oplus B)$$



Erosión

$$A \ominus B = \{p \in \mathcal{E}^2 : p + b \in A, \forall b \in B\}$$

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

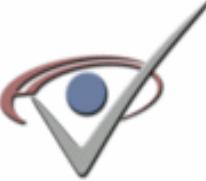
1	1
---	---

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

$$A = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$$

$$B = \{(0,0), (1,0)\}$$

$$A \ominus B = \{(1,2)\}$$



Erosión

$$A \ominus B = \{p \mid (B)_p \subseteq A\}$$

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

1	1
---	---

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

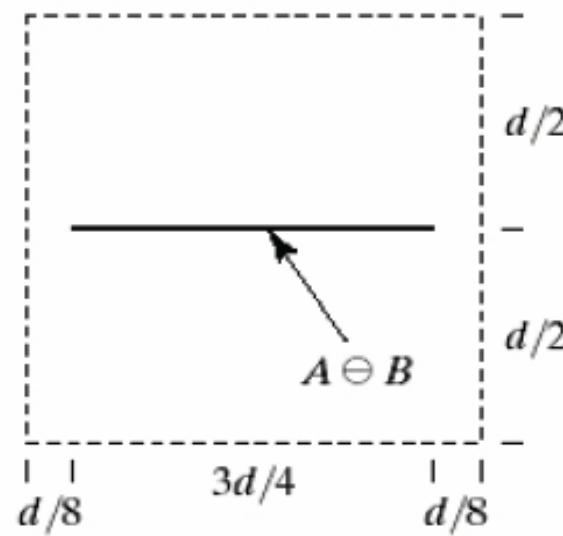
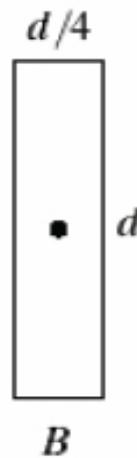
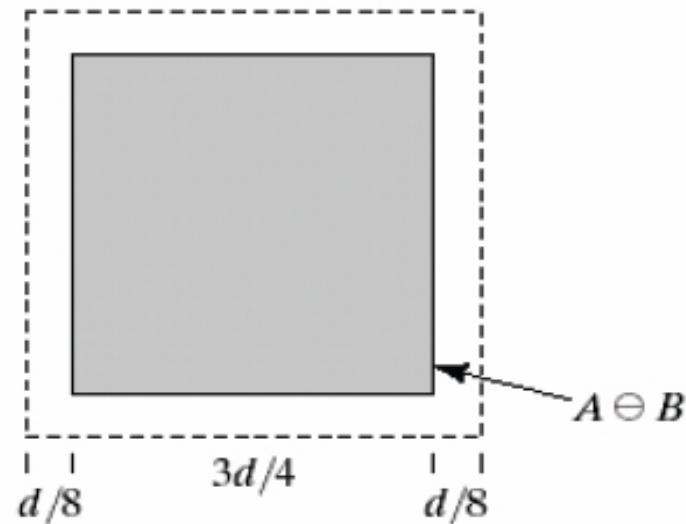
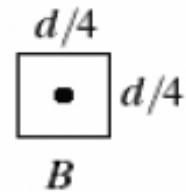
$$A = \{(1,0), (1,1), (1,2), (2,2), (0,3), (0,4)\}$$

$$B = \{(0,0), (1,0)\}$$

$$A \ominus B = \{(1,2)\}$$

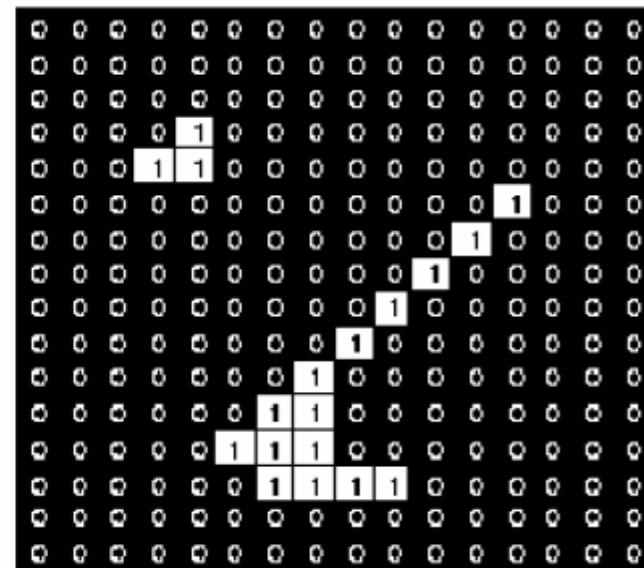
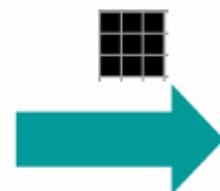
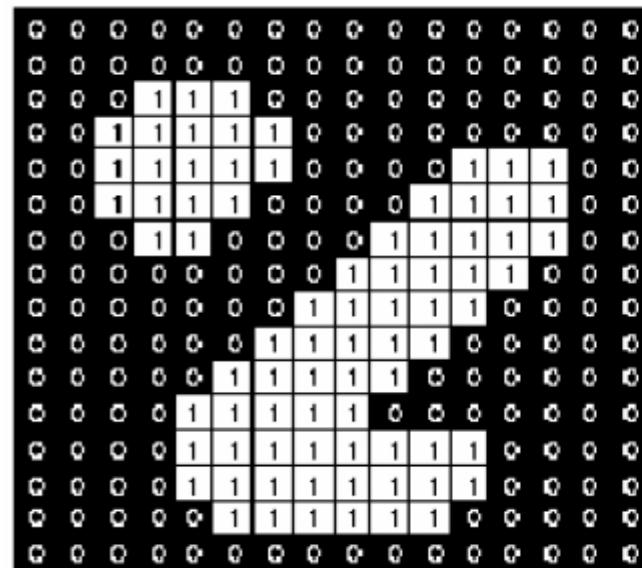


Erosión





Erosión



EE: cuadrado

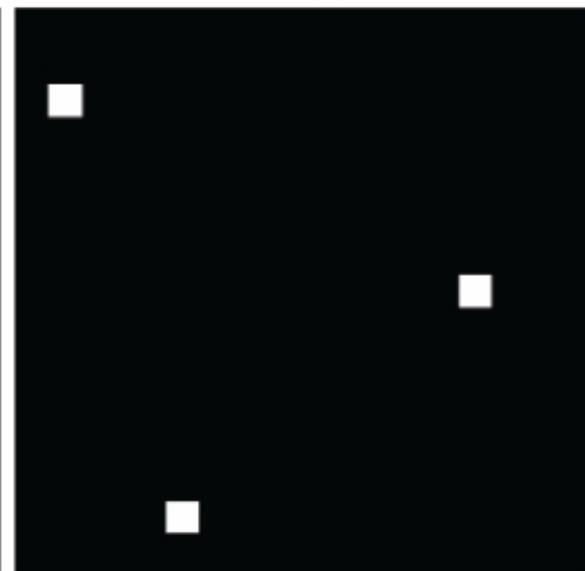
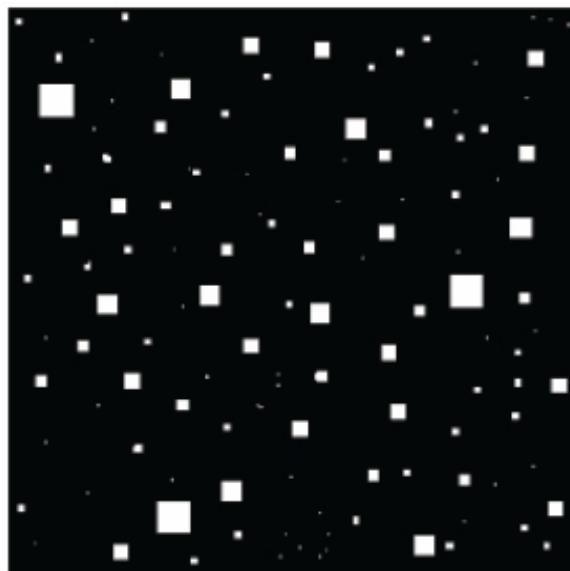


Erosión

Una de las aplicaciones más típicas de la erosión es la eliminación de detalles irrelevantes (desde el punto de vista de tamaño)

erosión

dilatación





Ejercicio

- Aplicar dilatación en esta imagen con el elemento estructurante indicado

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

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



Ejercicio

- Aplicar erosión en esta imagen con el elemento estructurante indicado

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

1	1
---	---



Filtros Morfológicos: **APERTURA**

Apertura de la imagen A por el EE B:

- Erosión de la imagen A con el elemento estructurante B seguida de una dilatación con el mismo elemento estructurante

$$A \circ B = (A \ominus B) \oplus B$$

- Se obtiene desplazando el EE B por el interior del conjunto y eliminando las zonas por las que B no pueda “pasar”.



APERTURA

Original



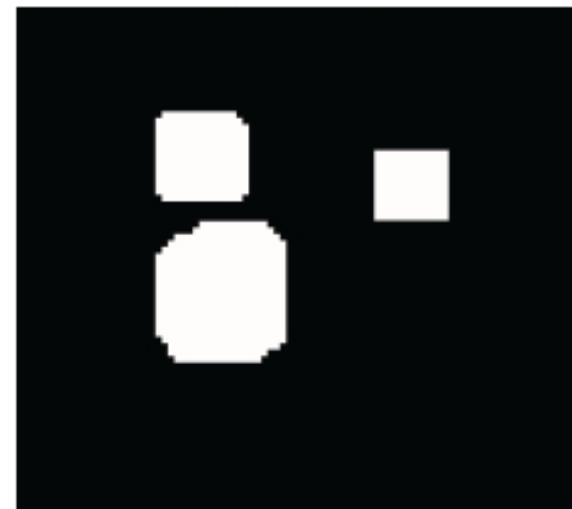
Círculo 7



Vertical 11



Cuadrado 11



Filtros Morfológicos: ***APERTURA***

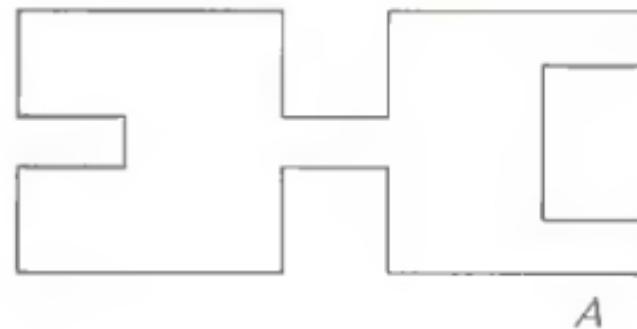


Propiedades:

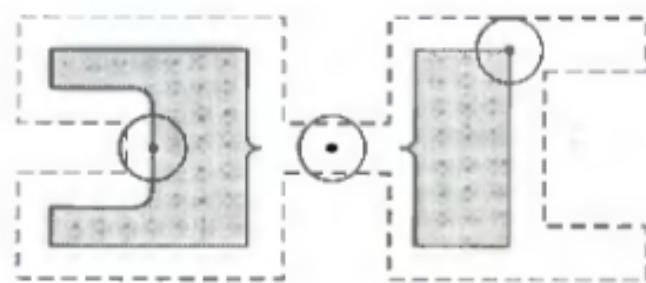
- Alisa contornos (redondear las esquinas donde no quepa el EE)
- Eliminar las protuberancias donde no quepa el EE
- Rompe istmos entre objetos. Separa objetos en puntos estrechos
- Elimina salientes



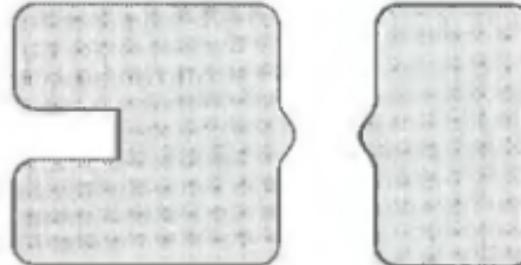
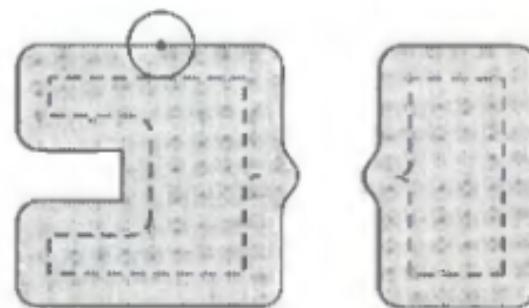
Apertura



A



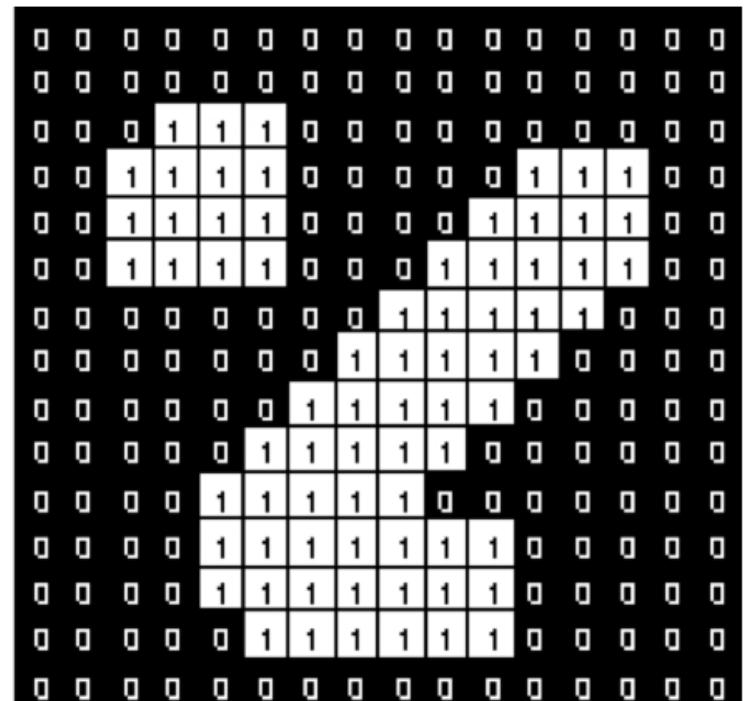
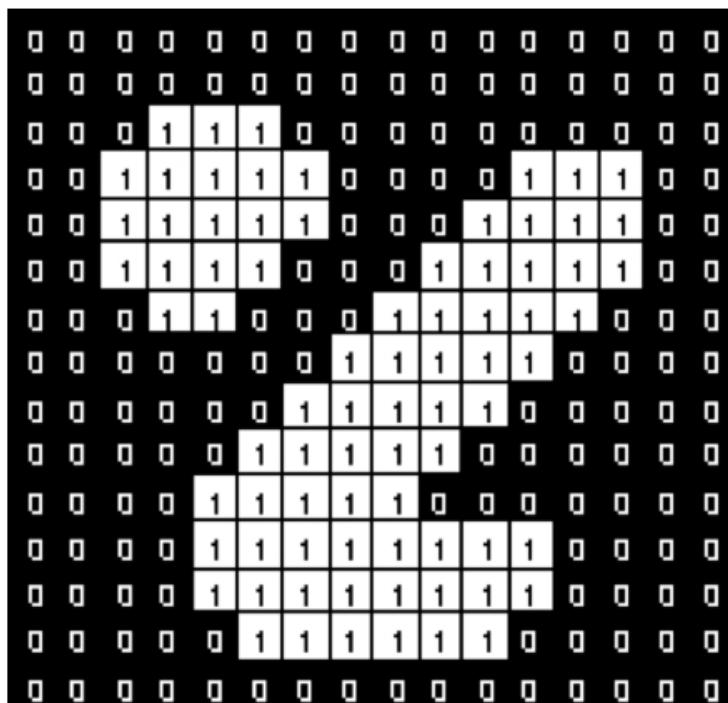
$A \ominus B$



$$A \circ B = (A \ominus B) \oplus B$$



Apertura

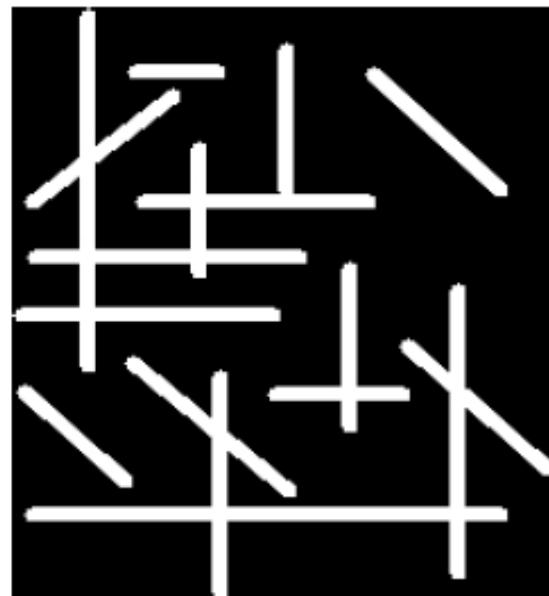
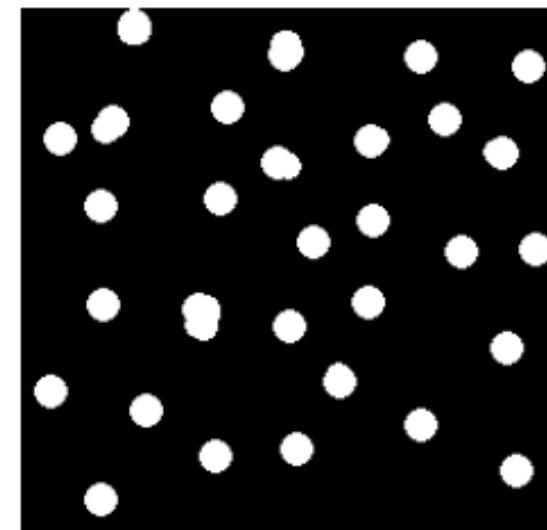




Apertura



Diámetro = 11



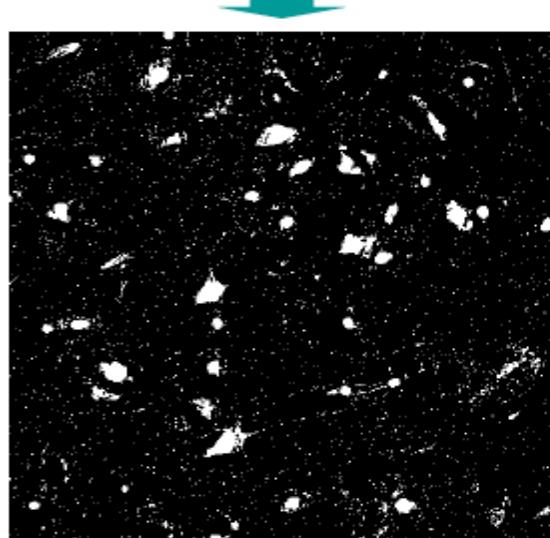
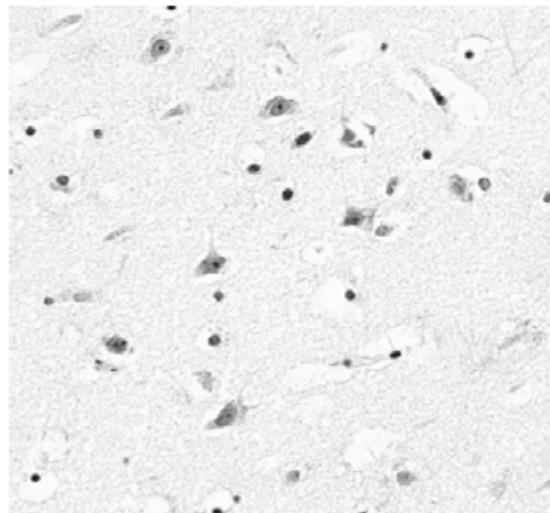
3x9



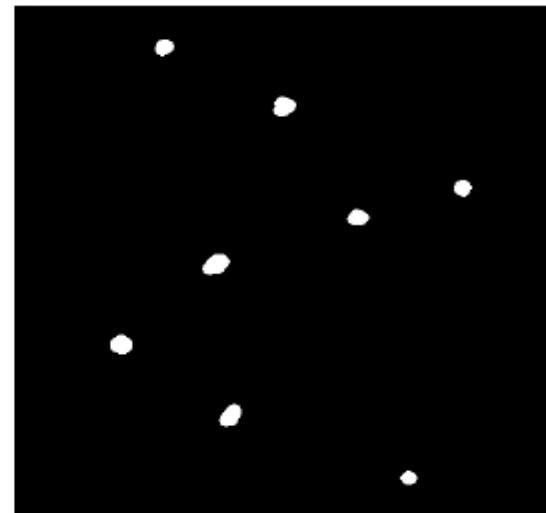


Apertura

Apertura como filtro de tamaño



D = 11



D = 7





Filtros Morfológicos: **CIERRE**

Cierre de la imagen A por el EE B:

- Dilatación de la imagen A con el elemento estructurante B seguida de una erosión con el mismo elemento estructurante

$$A \cdot B = (A \oplus B) \ominus B$$

- Se obtiene desplazando el EE B por el exterior del conjunto y anexionando las zonas por las que B no pueda “pasar”.

CIERRE



Original



Vertical 11



Cuadrado 11



Círculo 7





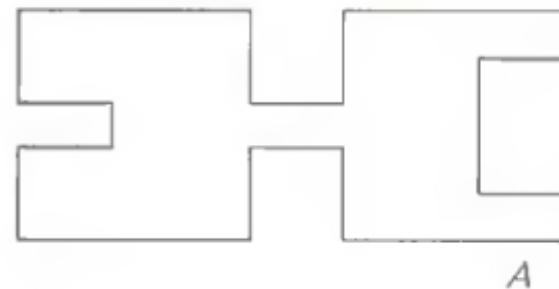
Filtros Morfológicos: **CIERRE**

Propiedades:

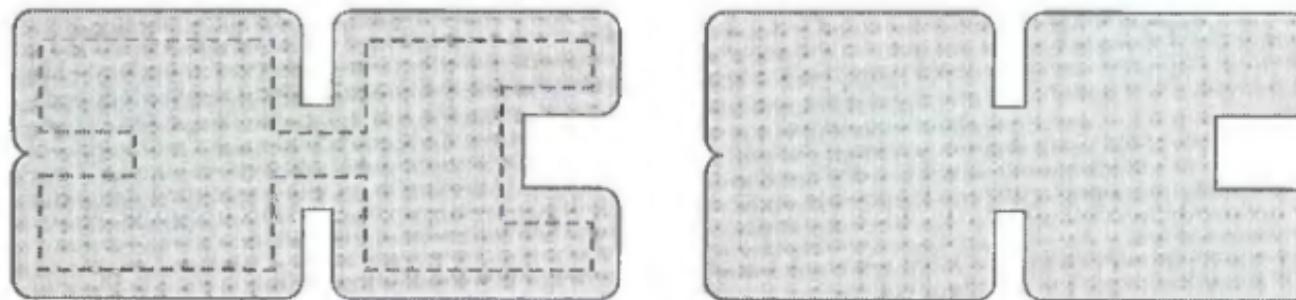
- Tiende a alisar porciones del contorno, fusiona estrechas grietas, y rellena vacíos en el contorno
- Rellena agujeros pequeños
- Eliminar los entrantes pequeños
- Conecta objetos vecinos



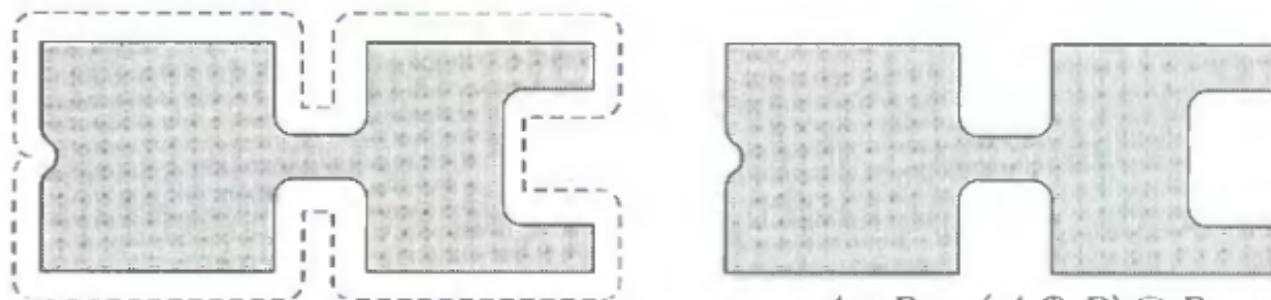
Cierre



A



A ⊕ B



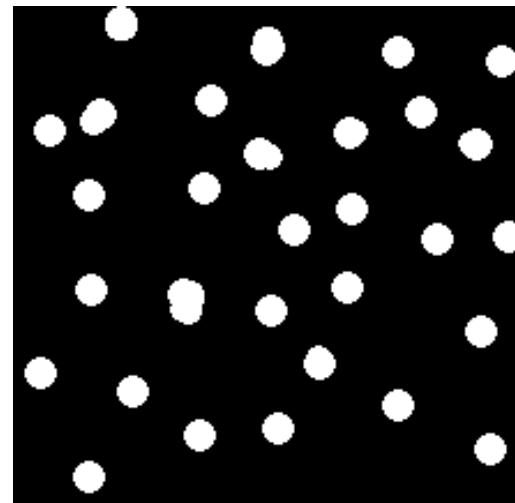
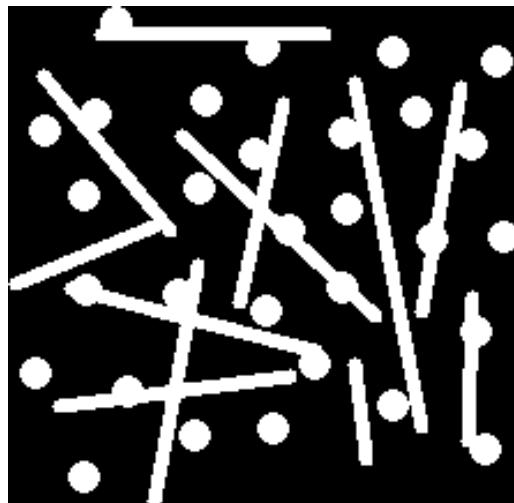
A · B = (A ⊕ B) ⊖ B

Cierre

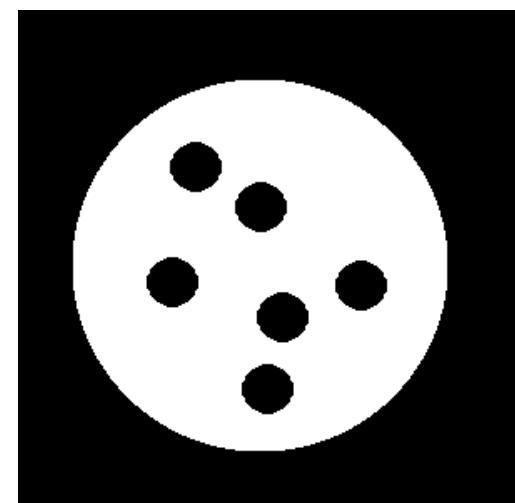
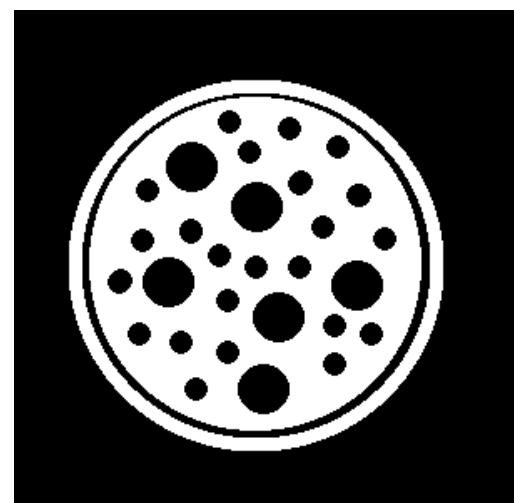




Apertura y cierre



Apertura con un elemento estructurante circular de tamaño 11x11

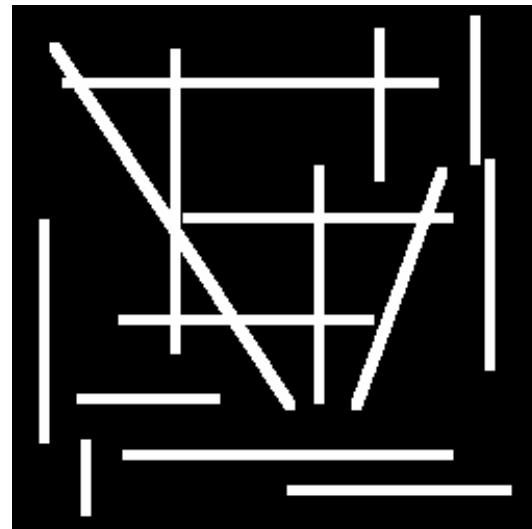


Cierre con un elemento estructurante circular de tamaño 22x22



Ejercicios

- Determinar el operador morfológico y el elemento estructurante necesario para
 - Eliminar las líneas verticales
 - Mantener únicamente los cruces de líneas



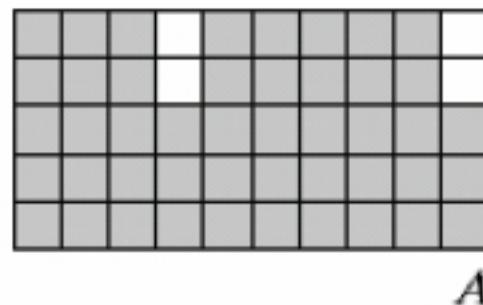
Detección de Contornos



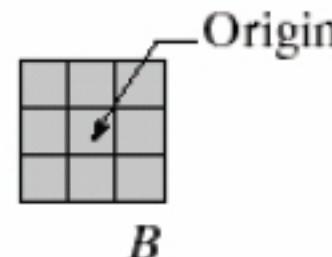
Extracción de un contorno:

$$\text{Cont}(A) = A - (A \ominus B)$$

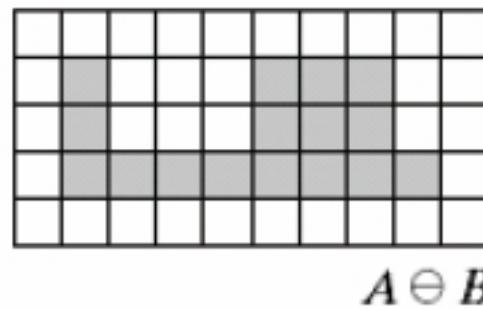
B es un EE adecuado para la aplicación



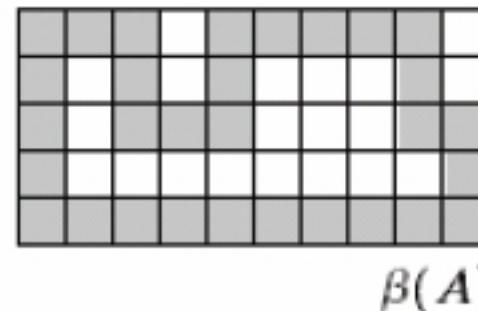
A



B

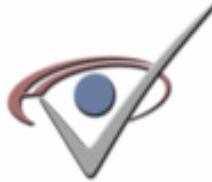


A \ominus *B*



$\beta(A)$

Detección de Contornos





LLENADO de REGIONES

- se parte de un conjunto A de puntos 8-conectados que encierran una región que se desea llenar
- se asigna el valor 1 a un punto p interior al contorno A ($X_0 = p$)
- se realiza la dilatación condicional dada por la fórmula siguiente con un elemento estructurante 3x3 de conectividad-4 (cruz):

$$X_k = (X_{k-1} \oplus B) \cap A^c$$

- el algoritmo termina cuando $X_k = X_{k+1}$
- el conjunto final es $X_k \cup A$



LLENADO de REGIONES

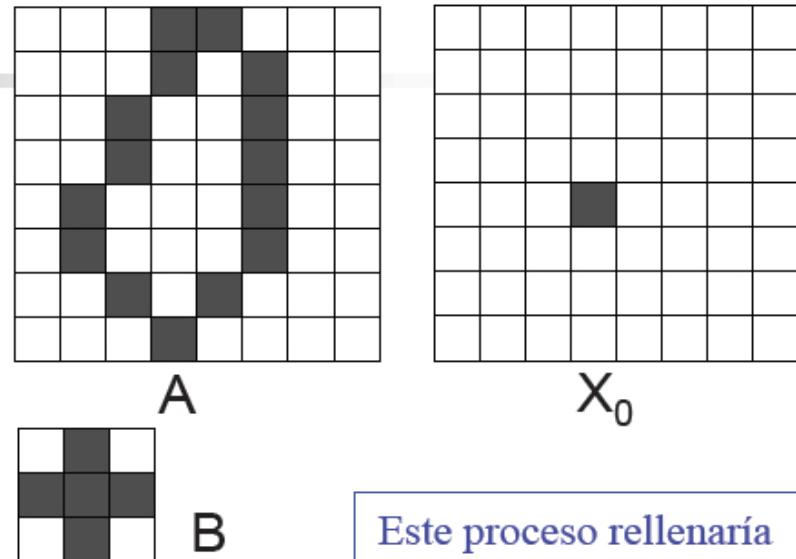


Ejemplo:

- Dilatación condicional

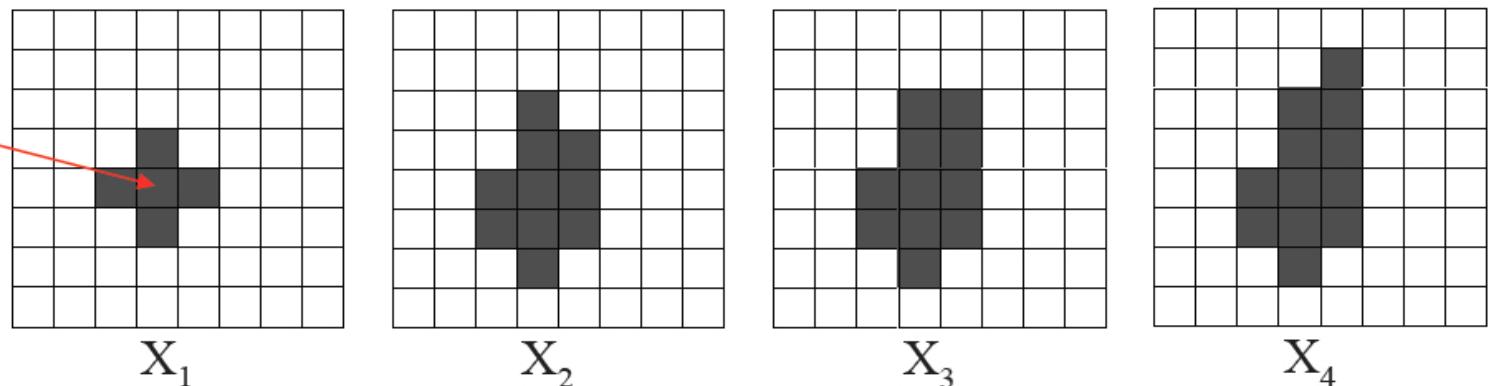
- $X_k = (X_{k-1} \oplus B) \cap A^c$
 $k=1,2,\dots$

- Si $X_n = X_{n-1} \rightarrow Y = X_n \cup A$
(y fin del proceso)



Este proceso rellenaría toda el área, por lo que se aplica la condición de intersección con A^c

Se empieza a partir de un punto dentro de la región





Transformada HIT-or-MISS (ganancia o pérdida)

Permite buscar determinados patrones de fondo y objeto(X).

- Sea $B=(J,K)$ donde J es el conjunto formado por los píxeles negros de B y K el conjunto formado por los píxeles negros de B^c .

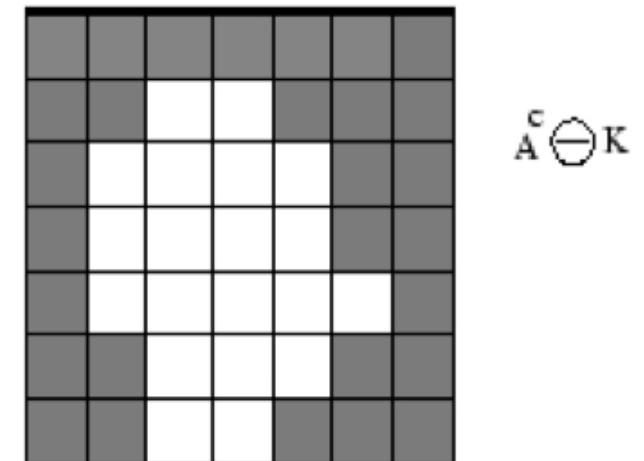
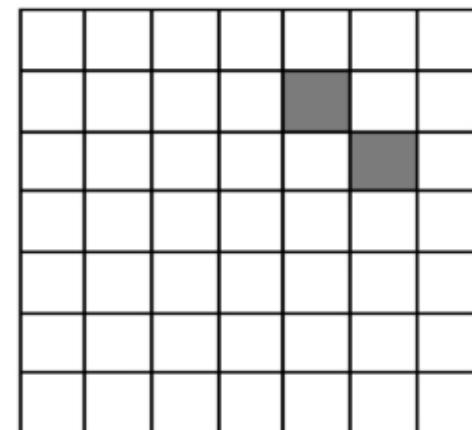
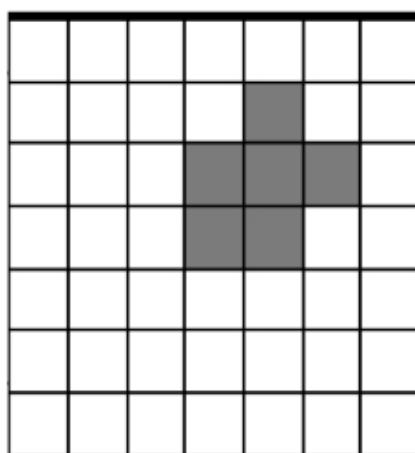
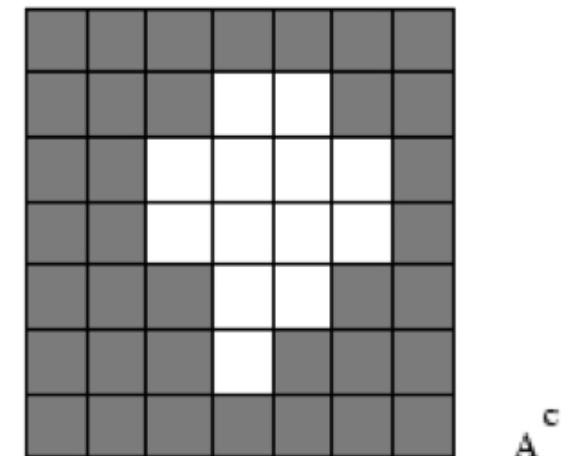
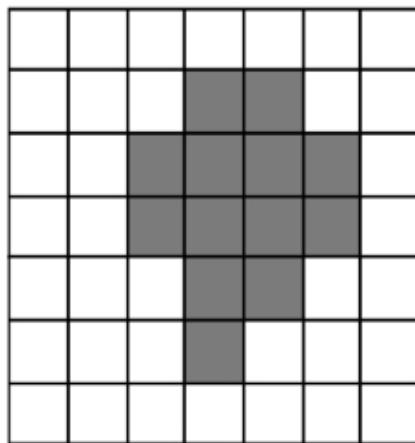
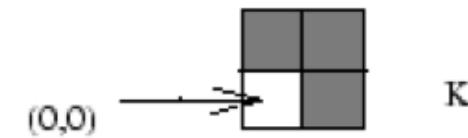
$$A \circledast B = (A \ominus J) \cap (A^c \ominus K) \quad A \circledast B = (A \ominus J) - (A \oplus \hat{K})$$

El primer conjunto (Hit) se queda con los objetos que incluyen a X (mayores que X)

El segundo conjunto (Miss) se queda con los incluidos en X (menores que X)



Transformada Hit-or-Miss





Transformada Hit-or-Miss

	1	
0	1	1
0	0	

	1	
1	1	0
	0	0

	0	0
1	1	0
	1	

0	0	
0	1	1
	1	

A 20x20 binary image where the 11th column contains a vertical sequence of 1's from row 4 to row 14. All other cells are 0's.



The same 20x20 binary image as the input, but with a vertical column of 1's from row 4 to row 14 highlighted in white, indicating the result of the hit-or-miss transform.



Transformada Hit-or-Miss

1)

0	0	0
0	1	0
0	0	0

2)

0	1	0
0	0	0

3a)

	1	
1		1
1		1

3b)

1		
1		1
1		1

Cuatro elementos estructurantes

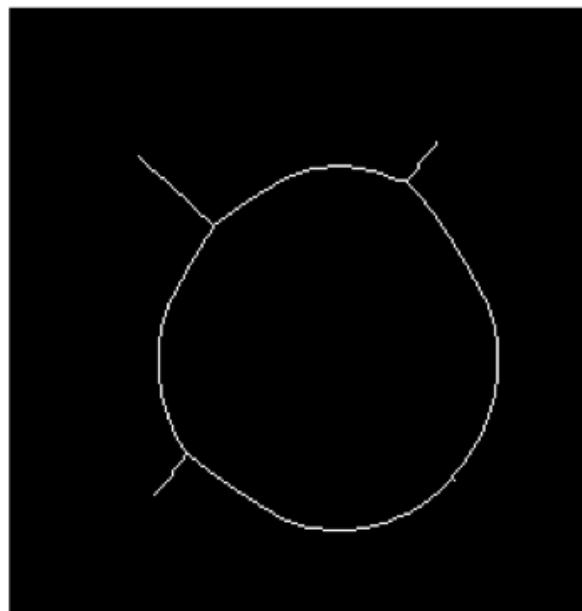
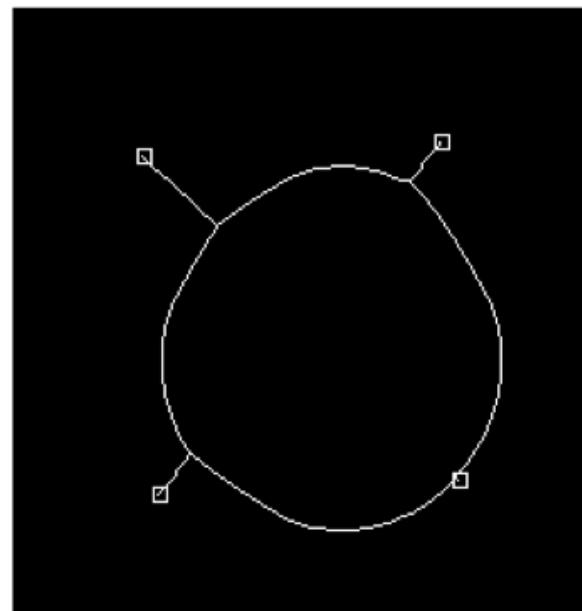
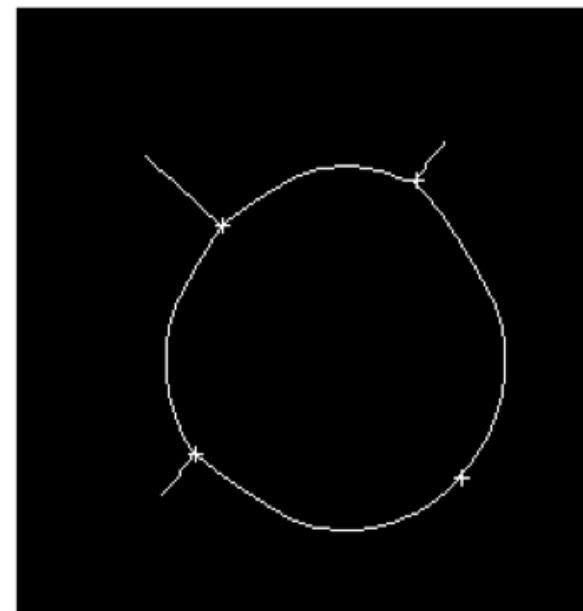


Imagen original



Puntos detectados
usando 2



Puntos detectados
usando 3



THINNING

- Esta operación permite eliminar píxeles del objeto.
- Su objetivo es ir estrechando los objetos hasta dejarlos como un camino de píxeles conexos.
- El ***thinning*** se aplica reiteradas veces de manera que se vaya limando la figura del objeto.
- Su principal utilidad es reducir el grosor de los bordes que se obtienen después de aplicar un detector.
- Los elementos estructurantes están formados por unos y ceros.
- Se puede expresar el thinning como:

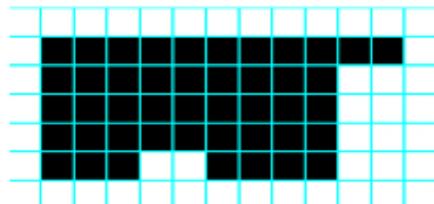
$$\text{thin}(I, J) = I - \text{hit-and-miss}(I, J)$$

donde la resta se define como:

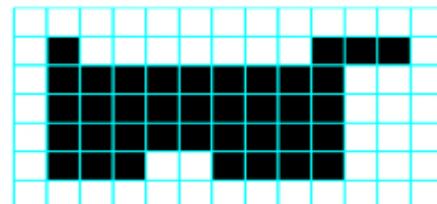
$$X - Y = X \cap \text{NOT } Y$$



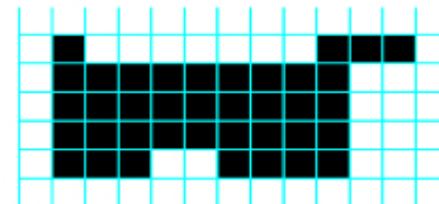
Thinning



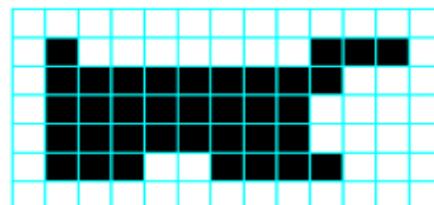
A



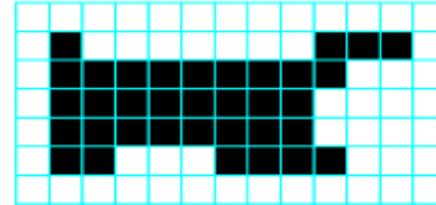
$A \otimes B_1$



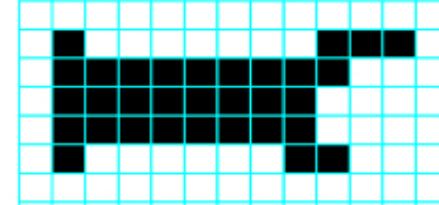
$A \otimes B_{1,2}$



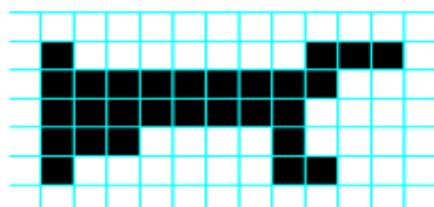
$A \otimes B_{1,3}$



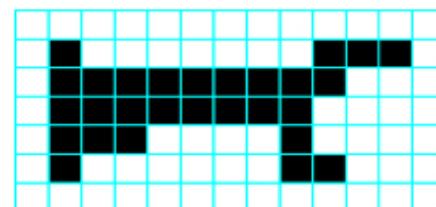
$A \otimes B_{1,3,4}$



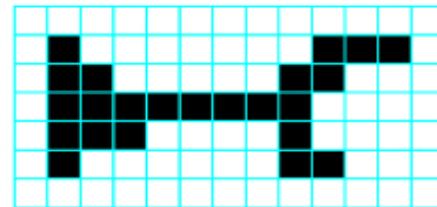
$A \otimes B_{1,3,4,5}$



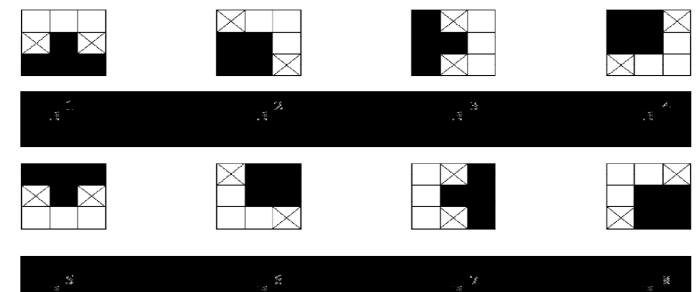
$A \otimes B_{1,3,4,5,6}$



$A \otimes B_{1,3,4,5,6,7,8,1}$



$A \otimes B_{1,3,4,5,6,7,8,1}$





THICKENING

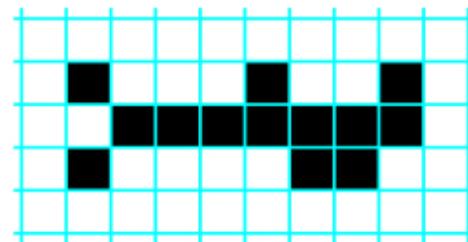
- Operación dual de thinning

$$AoB = A \cup (A \otimes B)$$

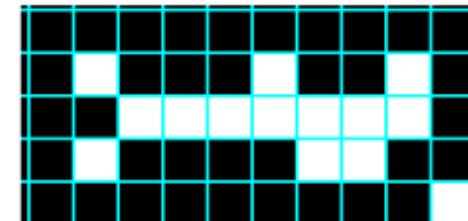
- En este caso se usan los mismos elementos estructurantes que en el Thinning pero cambiando los ceros por los unos.
- Esta idea se suele reemplazar por adelgazar el fondo y luego calcular su complementario.



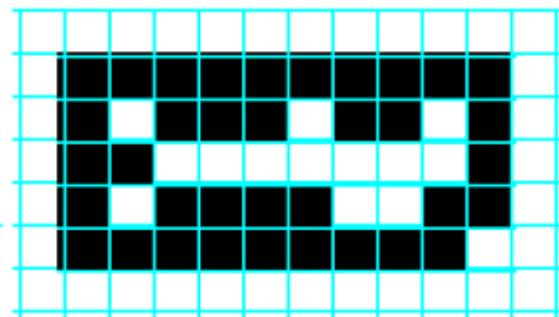
THICKENING



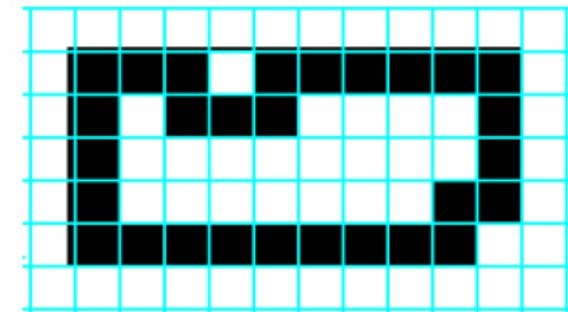
A



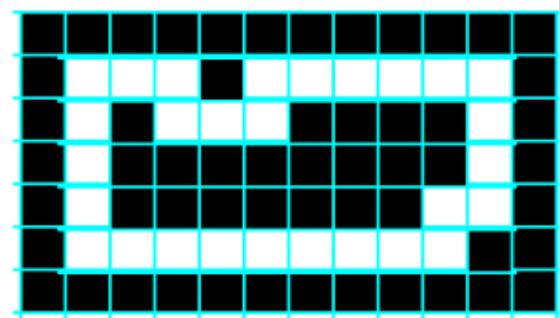
A^C



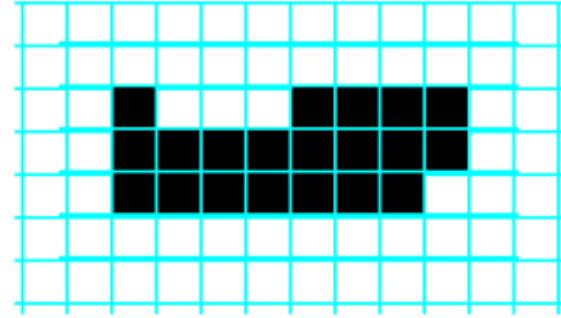
A^C



adelg (A^C)



adelg (A^C)^C



engros(A)

Imágenes en escala de grises



- Extensión de las operaciones morfológicas en imágenes binarias
- Uso de operaciones mínimo/máximo sobre un vecindario
- Elemento estructurante es una función de dos variables que especifica el nivel de gris deseado
 - El valor del elemento estructurante es añadido/restado al máximo/mínimo del vecindario



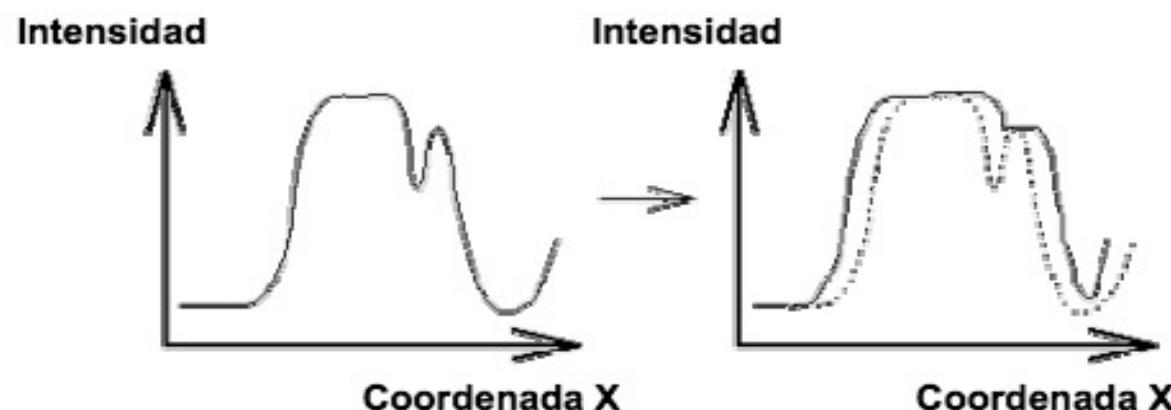
Dilatación en escala de grises

- Al aplicar la dilatación a las imágenes de niveles de gris se aplica el elemento estructurante (que también es de niveles de gris) a cada píxel para definir una vecindad y se escoge el valor máximo de la suma de los correspondientes píxeles.

$$D_G(A, B) = \max_{[j,k] \in B} \{a[m-j, n-k] + b[j, k]\}$$

- Habitualmente se simplifica como:

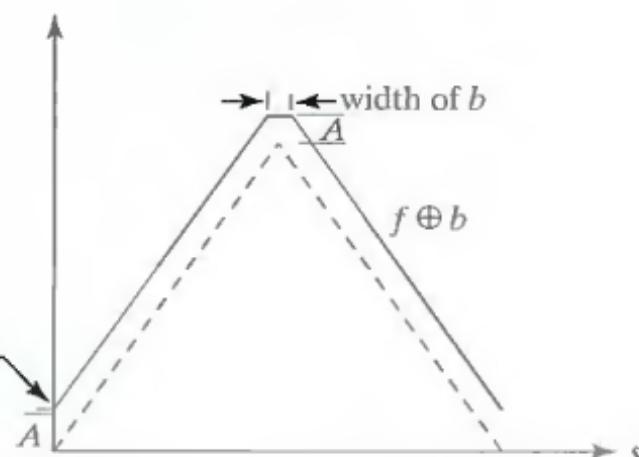
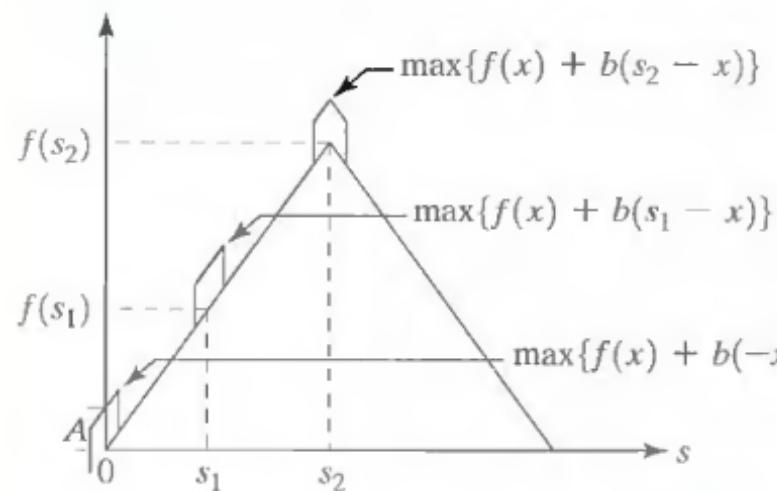
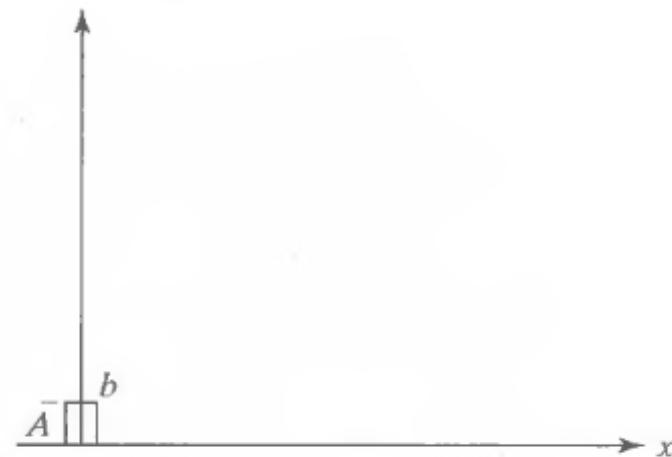
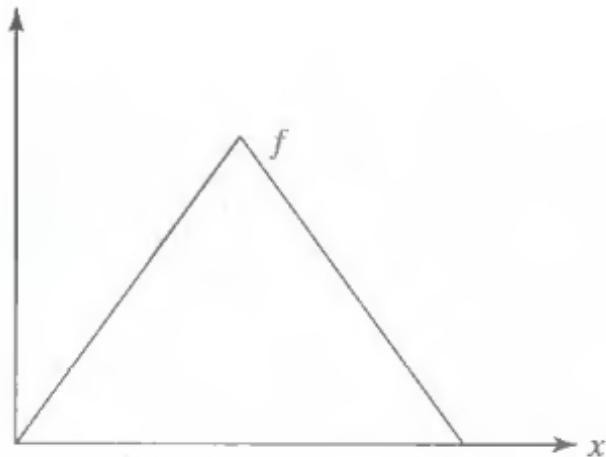
$$D_G(A, B) = \max_{[j,k] \in B} \{a[m-j, n-k]\} = \max_B(A)$$





Dilatación en 1D

$$(f \oplus b)(s) = \max \{ f(s-x) + b(x) \mid (s-x) \in D_f; x \in D_b \}$$



Efecto de la dilatación en escala de grises



- $b(x,y) > 0$
- Imagen de salida es más clara
- Los detalles oscuros se reducen/eliminan dependiendo de la relación entre las forma/tamaño de los detalles de la imagen y el elemento estructurante



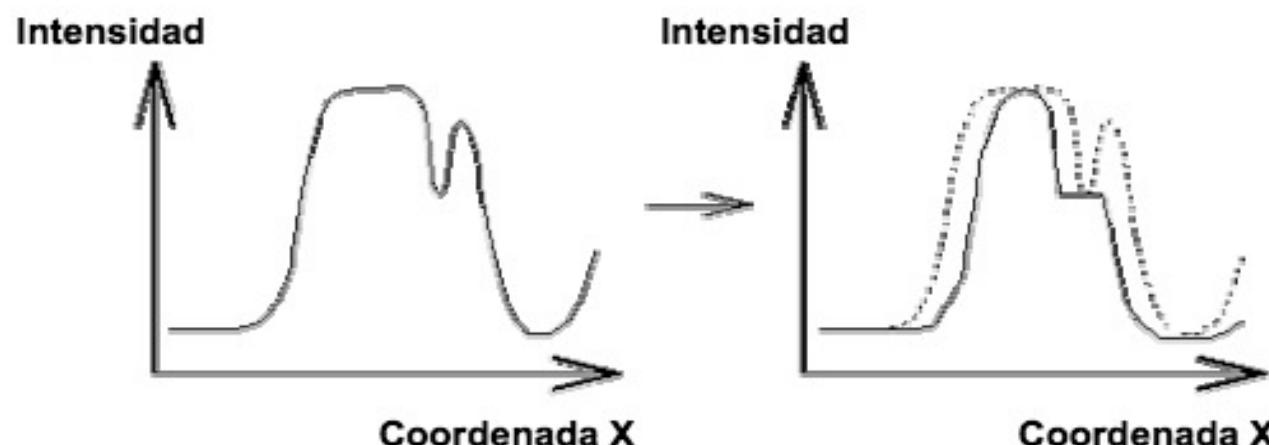
Erosión en escala de grises

- Al aplicar la erosión a las imágenes de niveles de gris se aplica el elemento estructurante (que también es de niveles de gris) a cada píxel para definir una vecindad y se escoge el valor **mínimo** de la suma de los correspondientes píxeles.

$$E_G(A, B) = \min_{[j,k] \in B} \{a[m+j, n+k] - b[j, k]\}$$

- Habitualmente se simplifica como

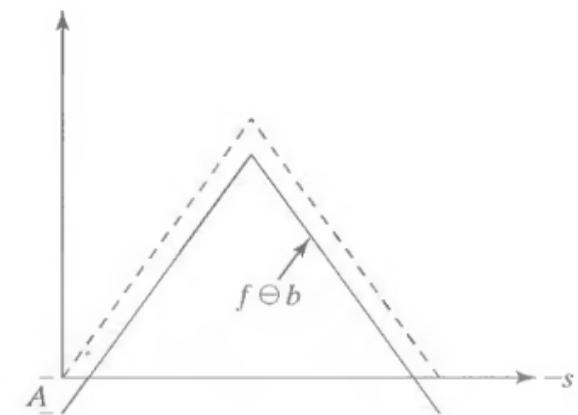
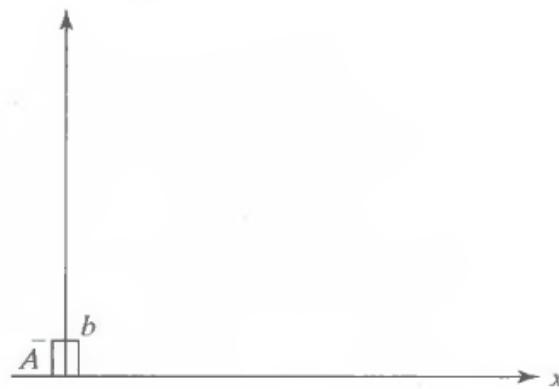
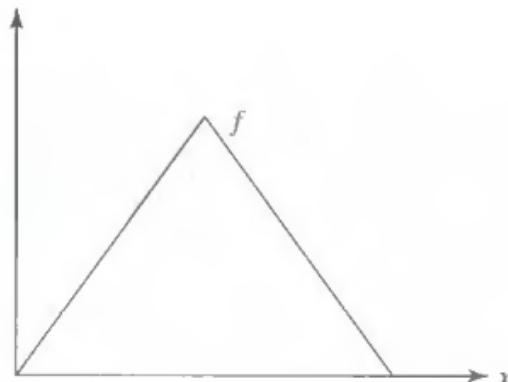
$$E_G(A, B) = \min_{[j,k] \in B} \{a[m-j, n-k]\} = \min_B(A)$$





Erosión en 1D

$$(f \ominus b)(s) = \min \{ f(s-x) - b(x) \mid (s-x) \in D_f; x \in D_b \}$$





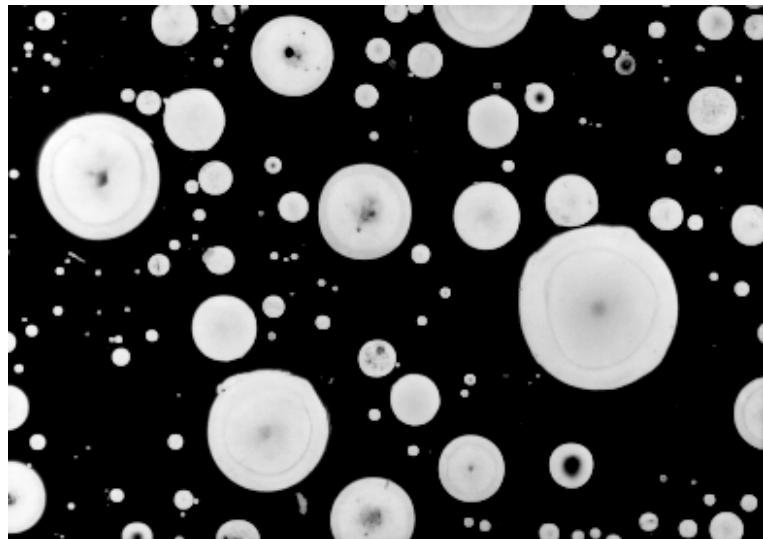
Efecto Dilatación en escala de grises

- $b(x,y) > 0$
- Imagen de salida es más oscuro
- Los detalles claros se reducen/eliminan dependiendo de la relación entre las forma/tamaño de los detalles de la imagen y el elemento estructurante

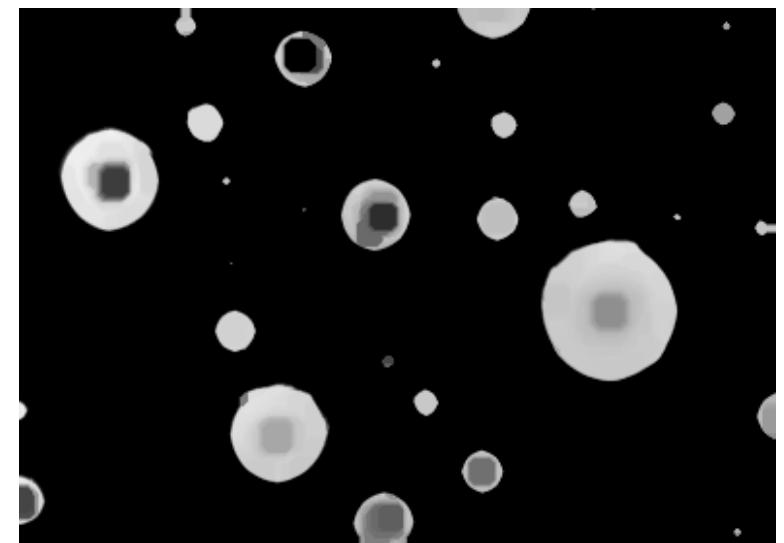


Erosión/dilatación en escala de grises

Dilatación
elemento estructurante
circular 13x13



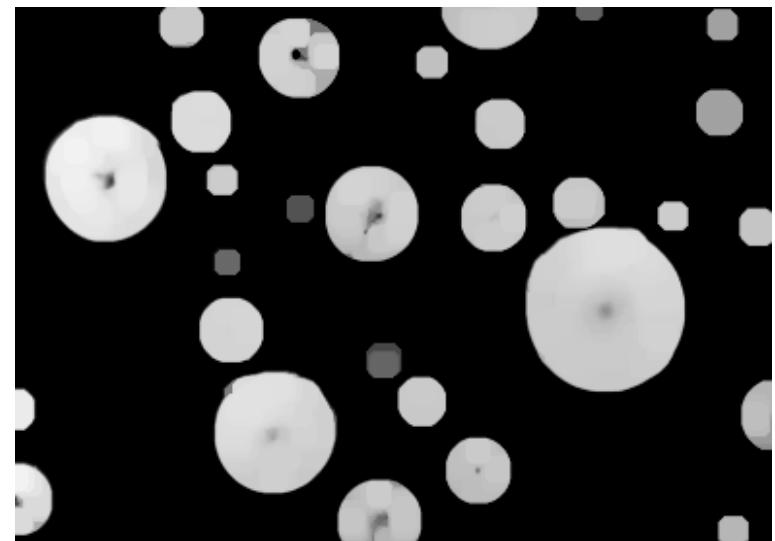
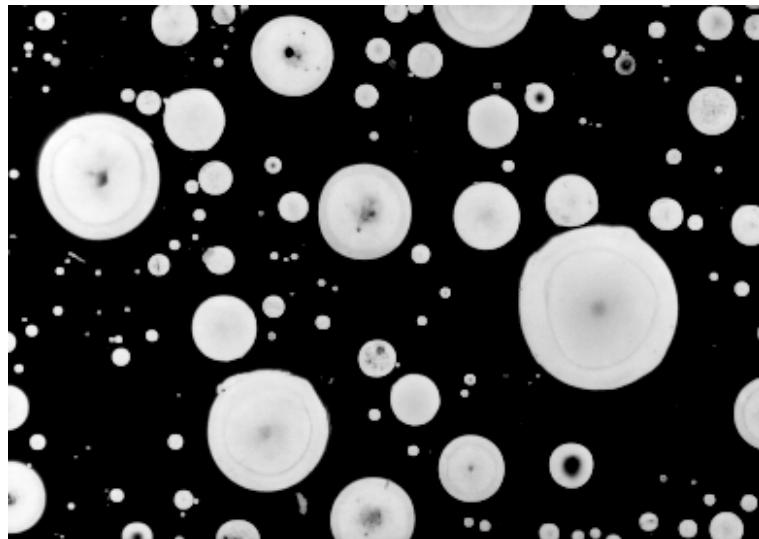
Erosión
Elemento estructurante
circular 13x13



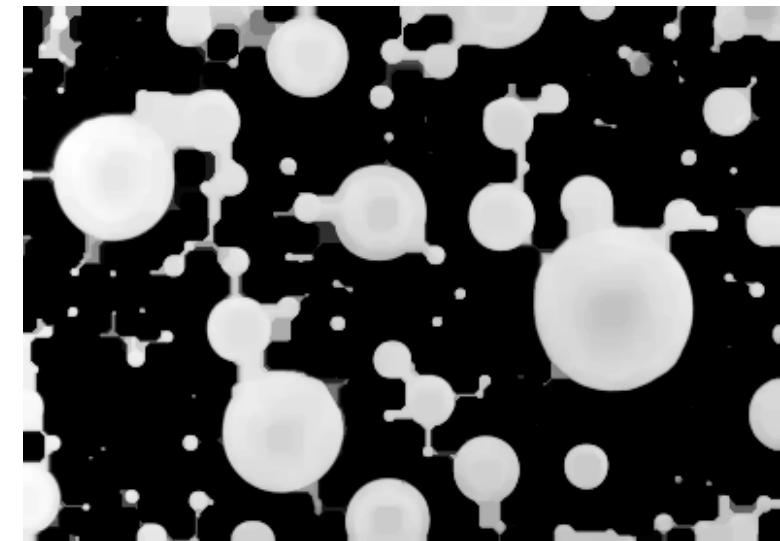


Apertura/cierre en escala de grises

Apertura
elemento estructurante
circular 13x13



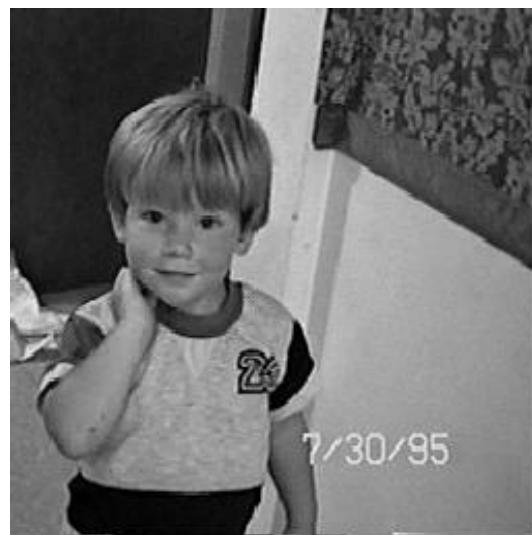
Cierre
Elemento estructurante
circular 13x13





Suavizado morfológico

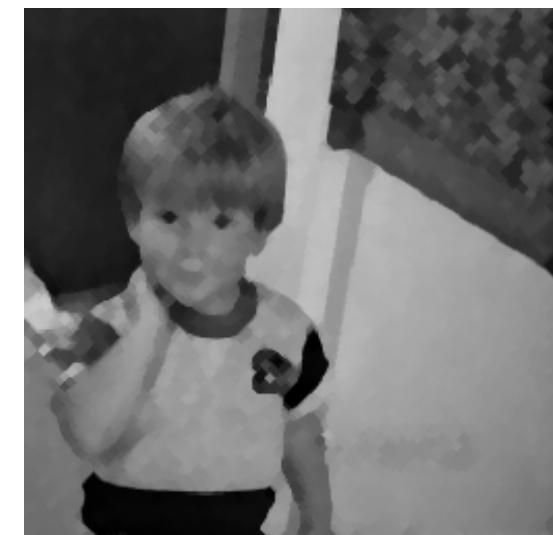
- Apertura seguida de cierre
 - Eliminación o atenuación de artefactos claros y oscuros



Apertura



Apertura, cierre



Elemento estructurante circular 5x5



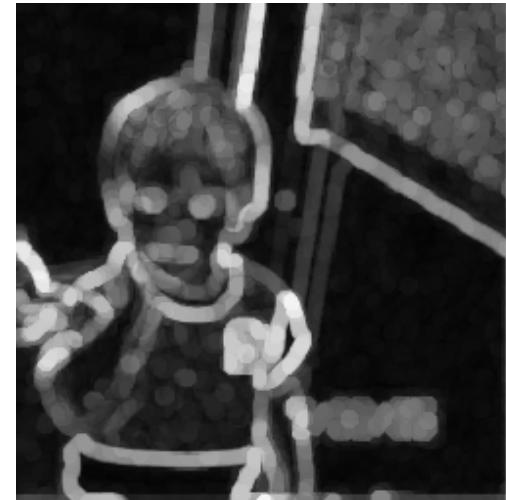
Gradiente morfológico

$$g = (A \oplus B) - (A \ominus B)$$

- Dilatación → imagen más clara con detalles oscuros suavizados
- Erosión → imagen más oscura con detalles claros suavizados
- Sustracción → diferencias entre zonas claras y oscuras



Gradiente morfológico



Dilatación



Erosión

Elemento estructurante circular 9x9



Transformación Top hat

- Segmentación de objetos en imágenes en escala de grises con intensidad distinta al fondo de la imagen

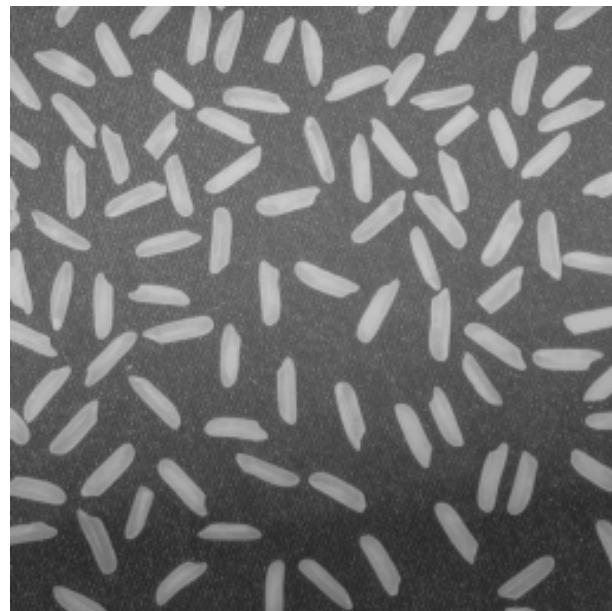
$$h = f - (f \circ b)$$

- Extrae objetos claros en un fondo oscuro no uniforme
- Elimina partes de la imagen que no coinciden con el elemento estructurante



Transformación Top hat

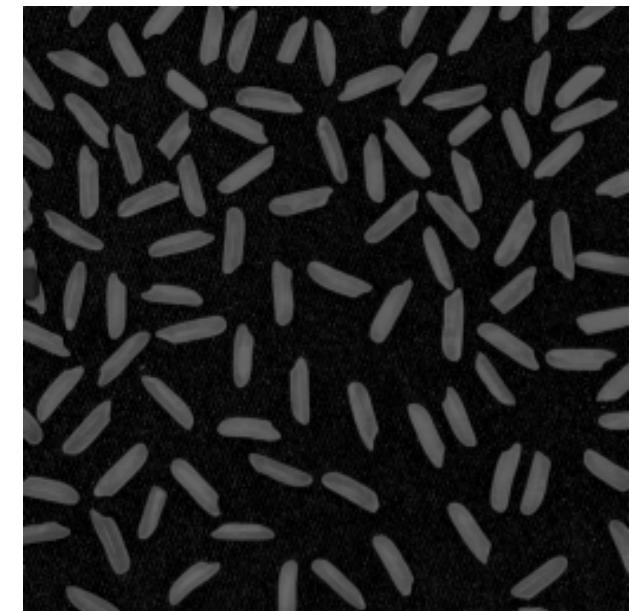
Imagen original



$(f \circ b)$



$f - (f \circ b)$



Elemento estructurante
disco 13x13



Operadores morfológicos

- Se centran en la forma de los objetos
- Multitud de algoritmos desarrollados a partir de dos operaciones básicas
 - Erosión
 - Dilatación