

Tratamiento de Señales Visuales

Tema 2: Operadores LSI

Álvaro García Martín
alvaro.garcia@uam.es



Escuela Politécnica Superior



Universidad Autónoma de Madrid
E28049 Madrid (SPAIN)



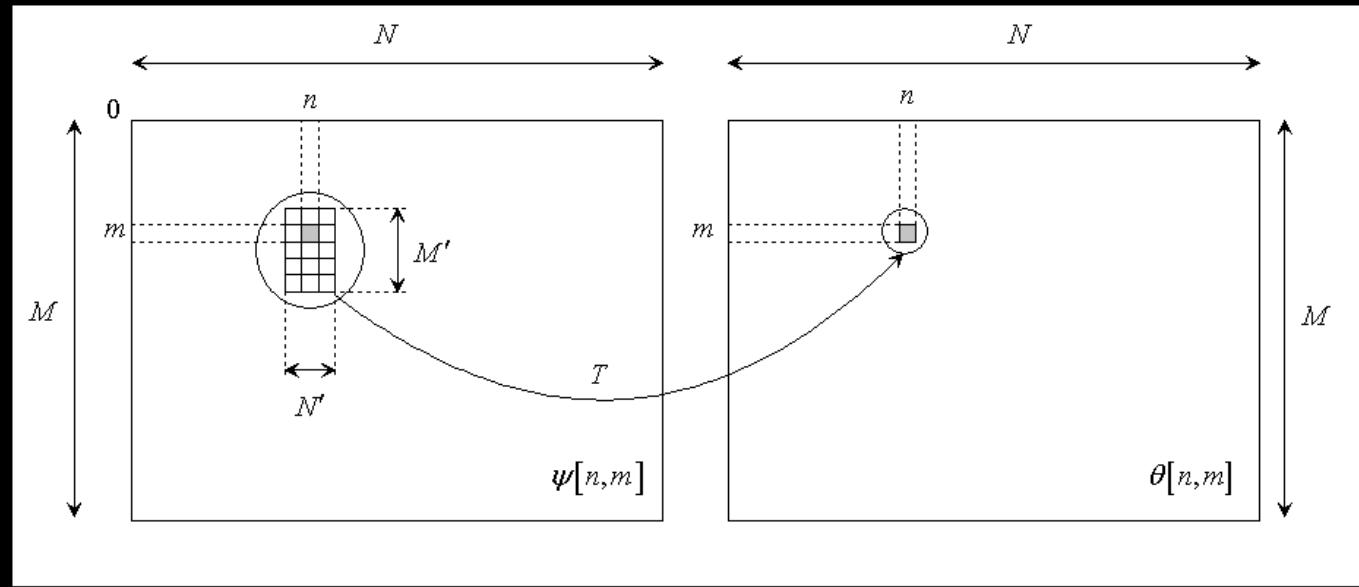
Video Processing
and Understanding
Lab

Grupo de Tratamiento e
Interpretación de Vídeo

1. Introducción
2. Aspectos operativos
3. Filtrado espacial
4. Ajustes geométricos
5. Restauración lineal

- ➡ 1. Introducción
- 2. Aspectos operativos
- 3. Filtrado espacial
- 4. Ajustes geométricos
- 5. Restauración lineal

- Los operadores locales efectúan una transformación (*spatial filtering*):



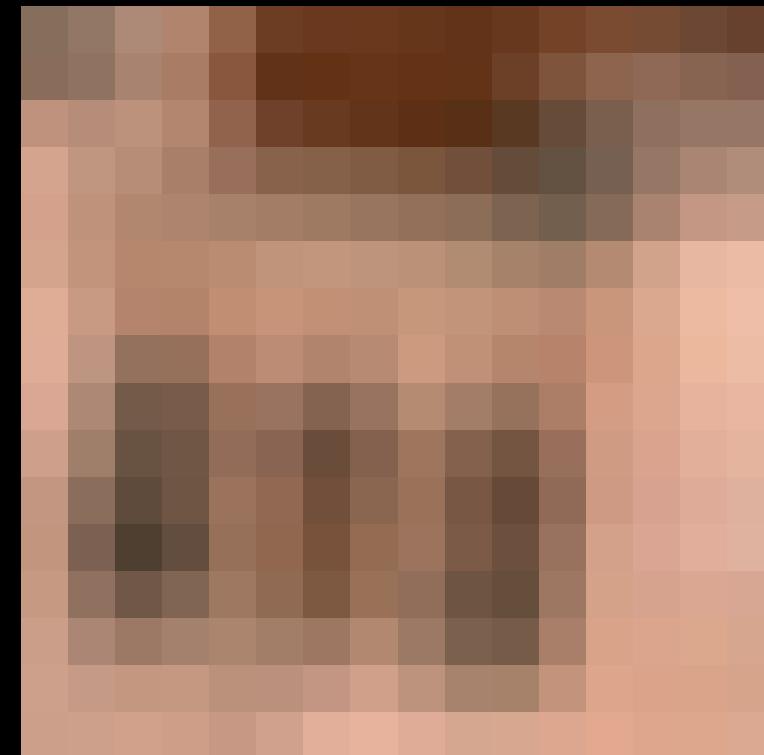
$$\psi[n, m] \xrightarrow{T} \theta[n, m] / \quad \theta[n, m] = T(\text{entorno de } \psi[n, m]), \\ n \in [0, N - 1], m \in [0, M - 1]$$

- Tipos de operadores:
 - Filtrado espacial
 - Ajustes geométricos
 - Operadores morfológicos
 - ...

- Operadores locales: ventana deslizante

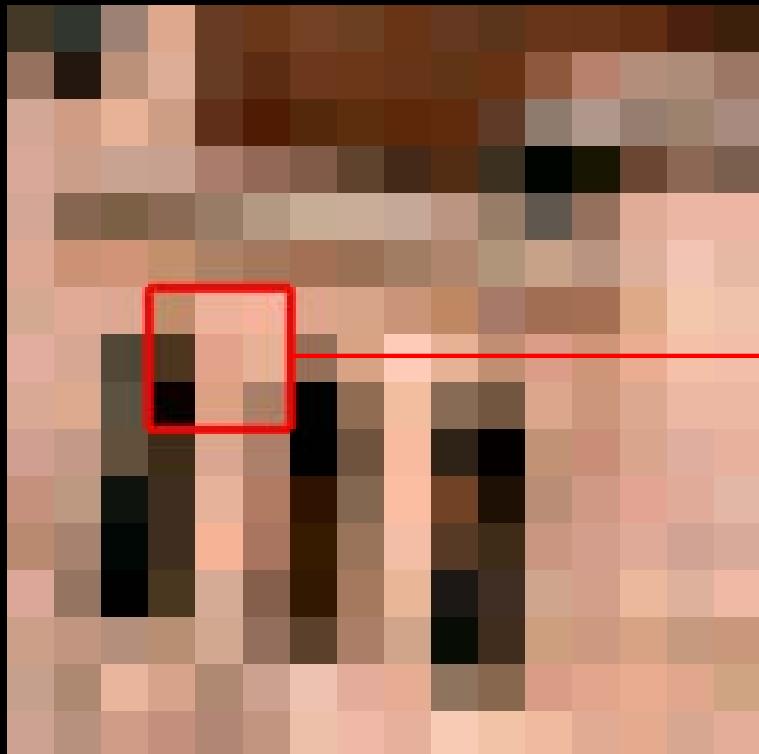


original

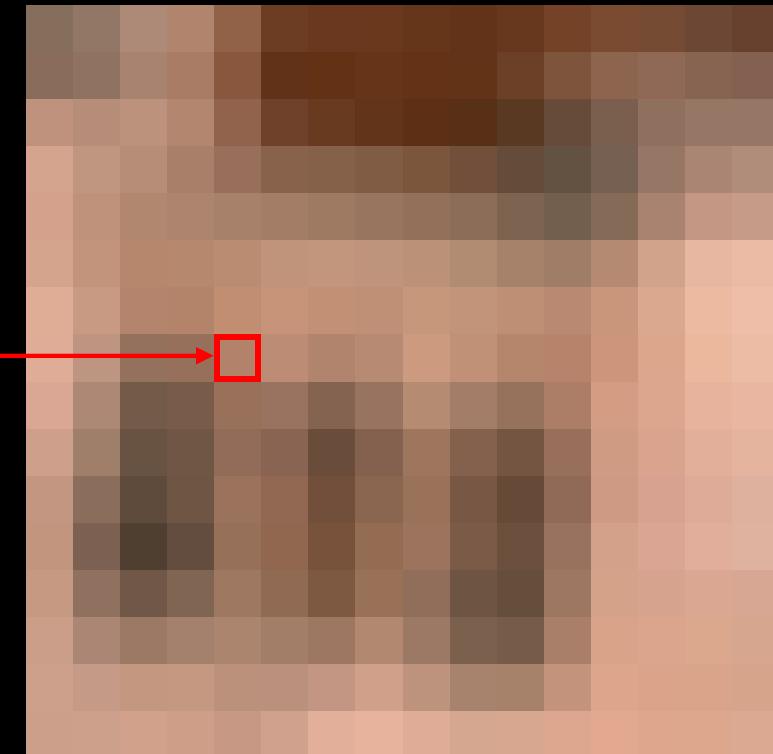


3×3 media

- Operadores locales: ventana deslizante

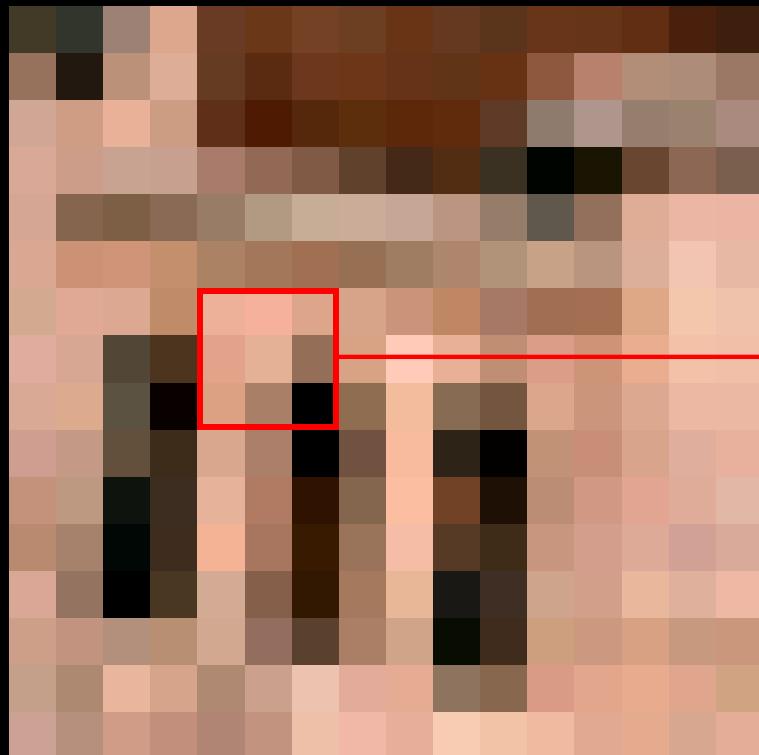


original

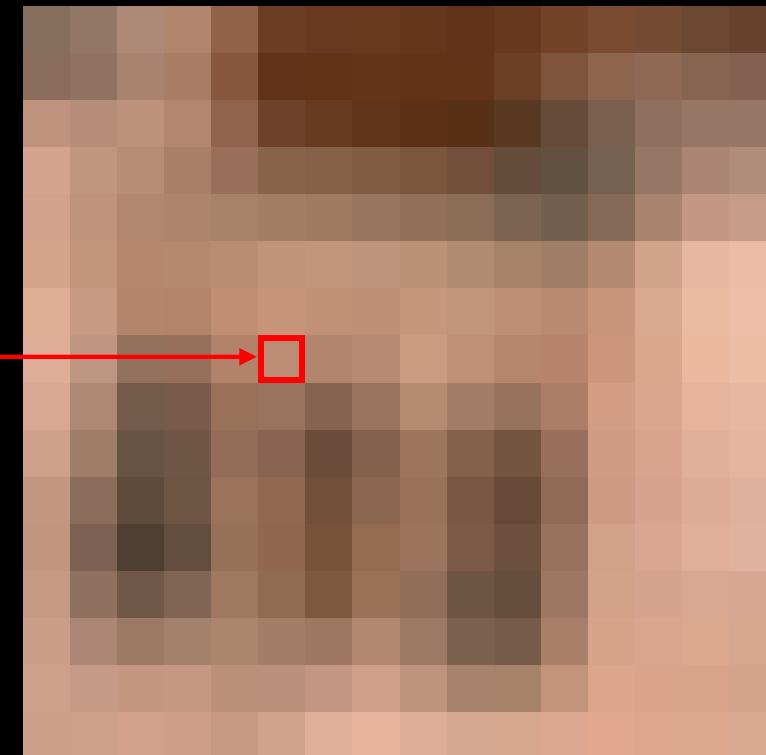


3×3 media

- Operadores locales: ventana deslizante

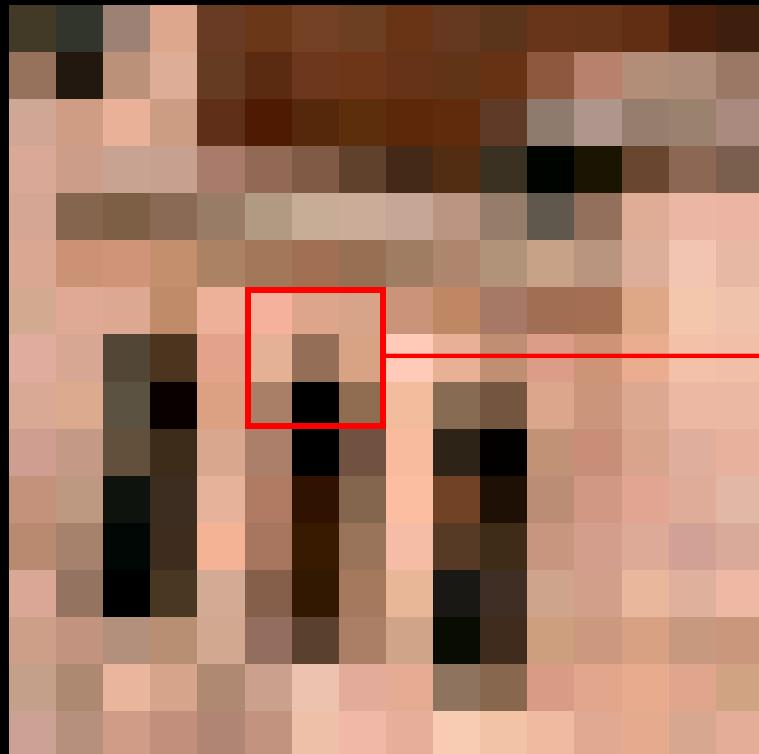


original

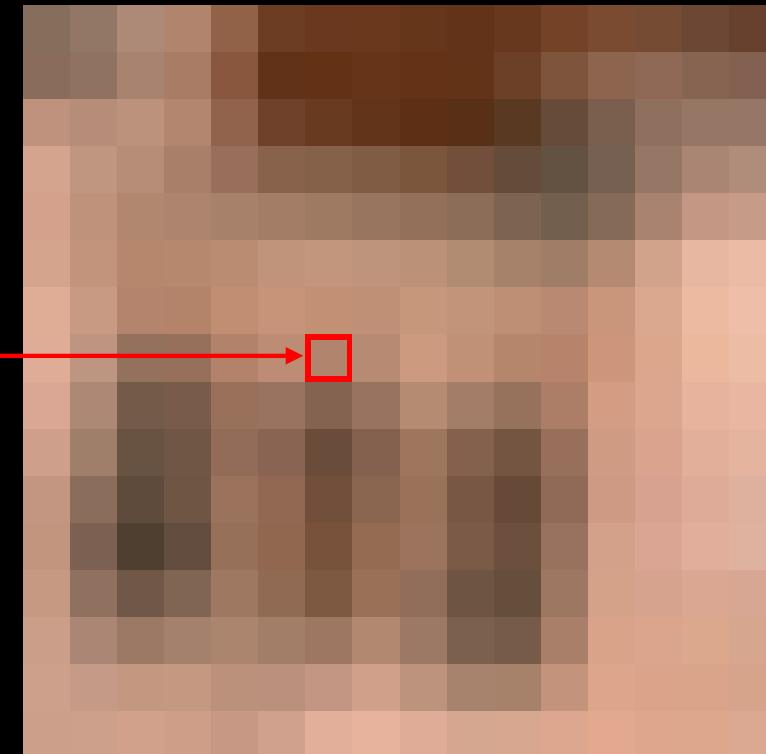


3×3 media

- Operadores locales: ventana deslizante

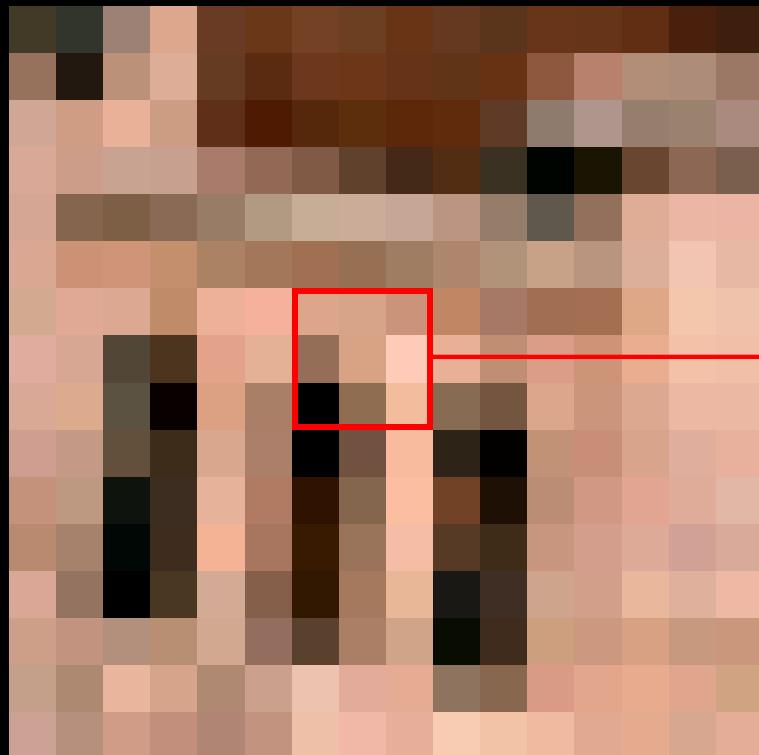


original

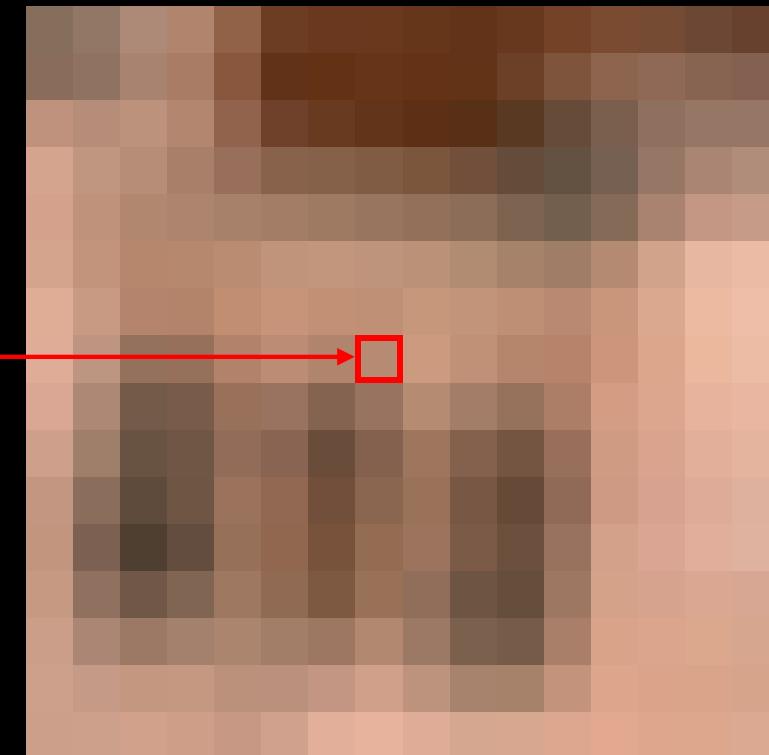


3x3 media

- Operadores locales: ventana deslizante

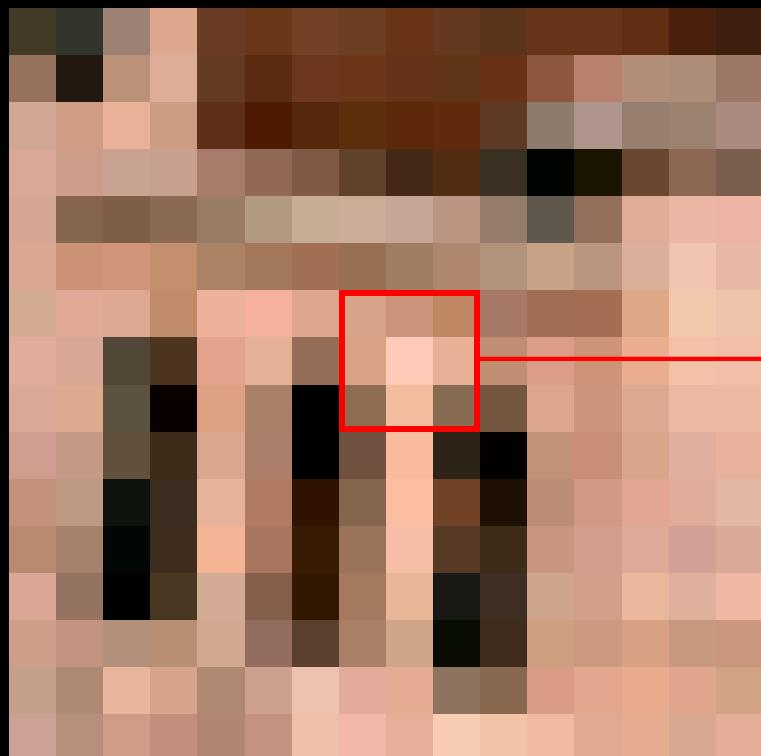


original

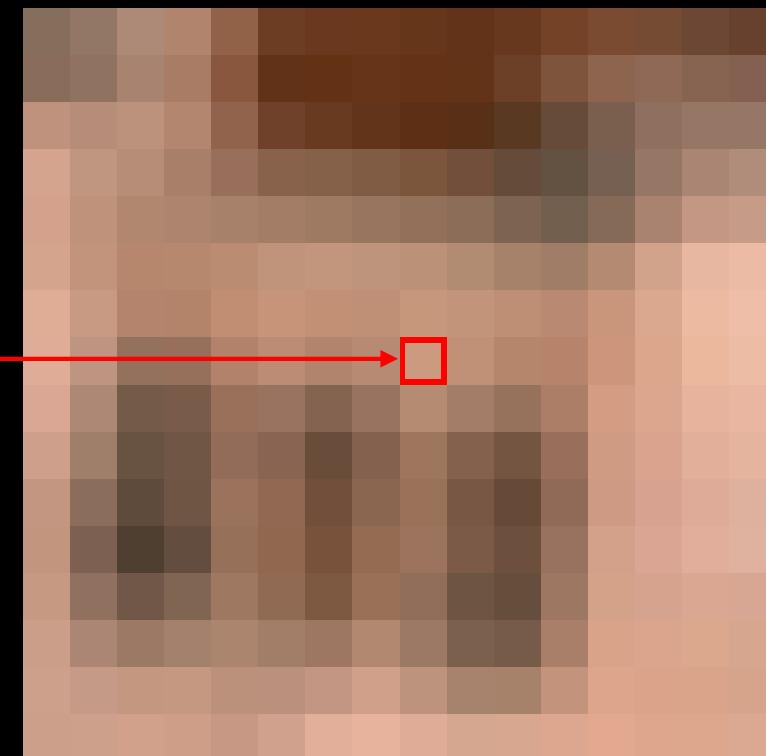


3×3 media

- Operadores locales: ventana deslizante

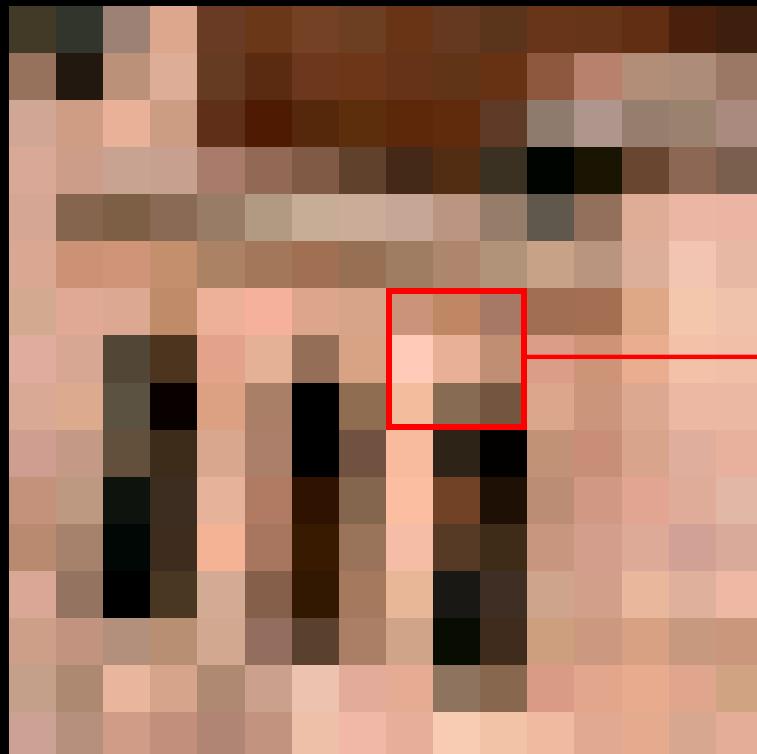


original

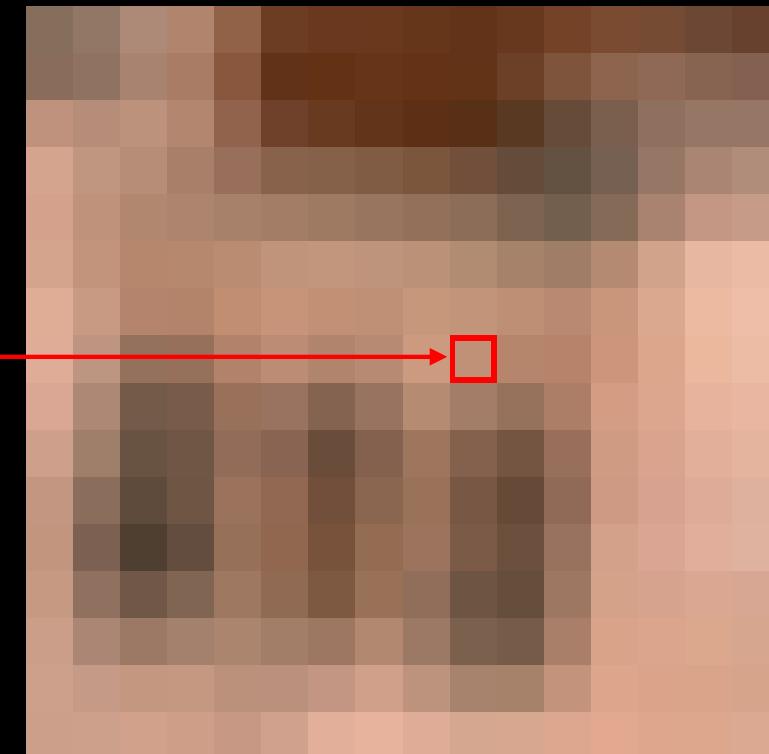


3×3 media

- Operadores locales: ventana deslizante



original



3x3 media

1. Introducción
2. Aspectos operativos
 - Respuesta al impulso
 - Tipos de filtros
 - Aplicación en los bordes
 - Separabilidad
3. Filtrado espacial
4. Ajustes geométricos
5. Restauración lineal

- Si la transformación es LSI (Lineal e Invariante con el tiempo), su respuesta al impulso unidad vendrá dada por:

$$\delta[n, m] \xrightarrow{T} h[n, m] = T(\delta[n, m])$$

- La respuesta a una imagen cualquiera resulta (desde el punto de vista de un sistema):

$$\psi[n, m] \xrightarrow{T} \theta[n, m] = \psi[n, m] * h[n, m]$$

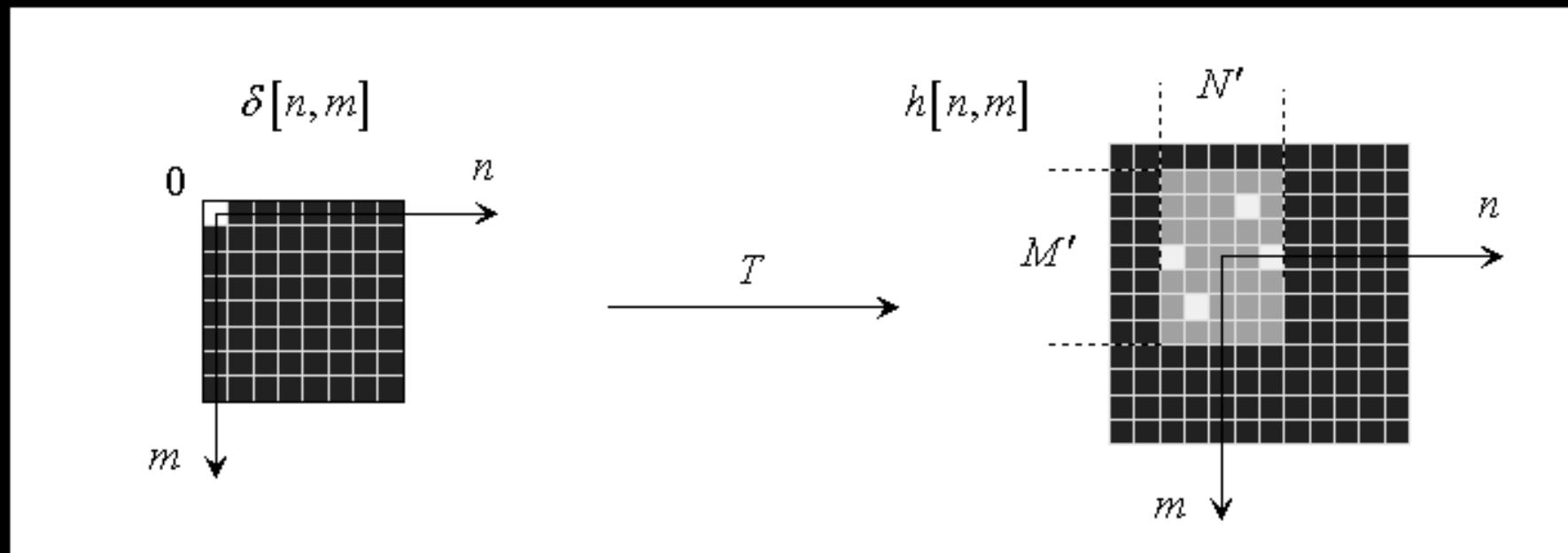


Convolución

→ Operadores locales también se conocen como operadores LSI

- Suponiendo que $h[n, m]$ es real, rectangular ($M' \times N'$) y simétrica respecto al origen:

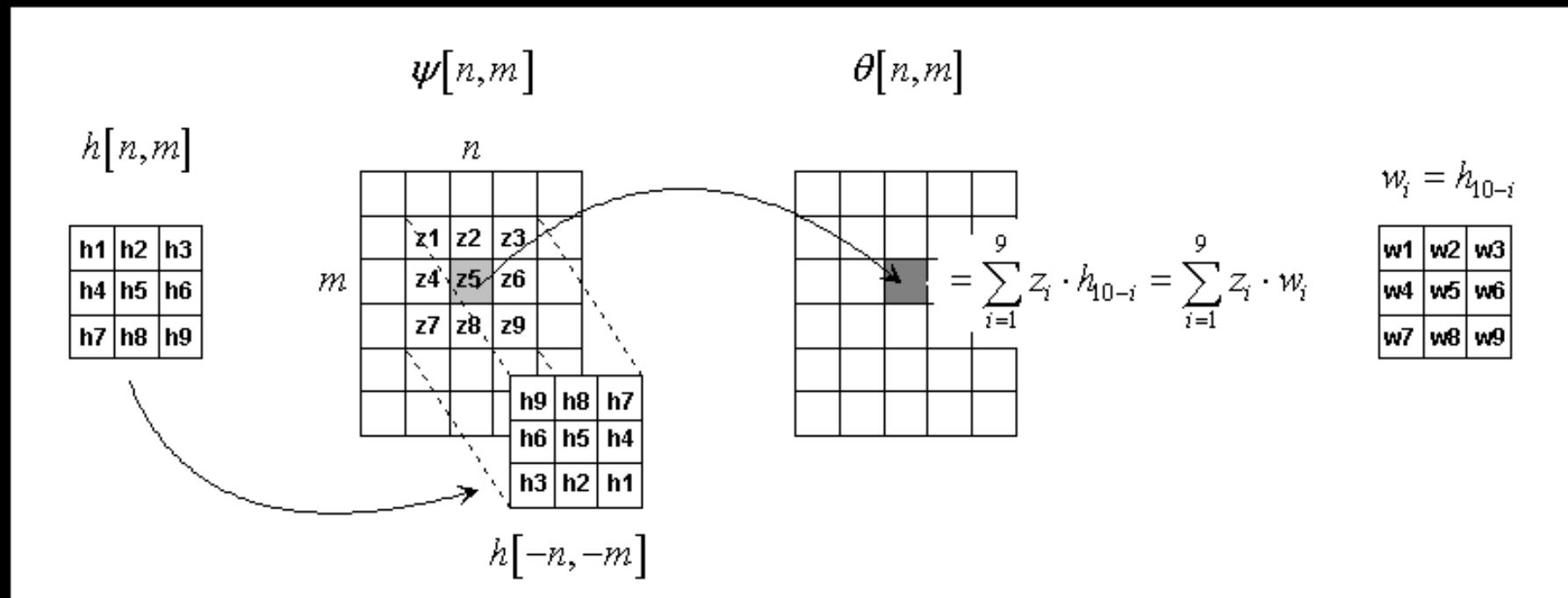
$h[n, m] = 0 \forall n, m / |n| > b, |m| > a, \text{ de modo que } N' = 2b + 1, M' = 2a + 1$



- La operación de convolución puede expresarse como:

$$\theta[n, m] = \psi[n, m] * h[n, m] = \sum_{k=-a}^a \sum_{l=-b}^b \psi[k, l] \cdot h[n - k, m - l]$$

- $\theta[n, m]$ se obtiene centrándolo $h[-n, -m]$ sobre cada píxel $\psi[n, m]$ y sumando los productos de cada coeficiente de h por cada píxel homólogo de la vecindad



- A la ‘imagen’ $h[-n, -m] = w[n, m]$ se le denomina por ello máscara, filtro o *kernel* de la transformación. Se puede convolucionar con h o correlacionar con w

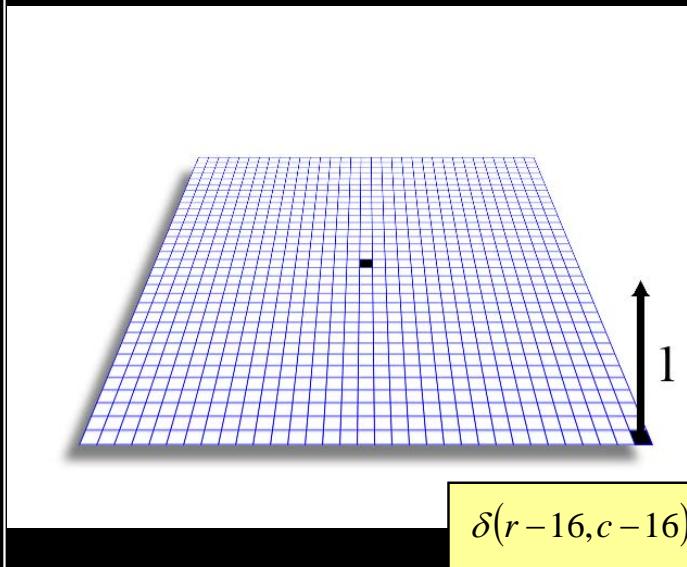
$$\theta[n, m] = \sum_{k=-a}^a \sum_{l=-b}^b \psi[k, l] \cdot w[k, l]$$

- La operación de filtrado puede llevarse a cabo bien el dominio ‘natural’ (o del pixel) o bien en el dominio frecuencial, aplicando la propiedad de convolución:

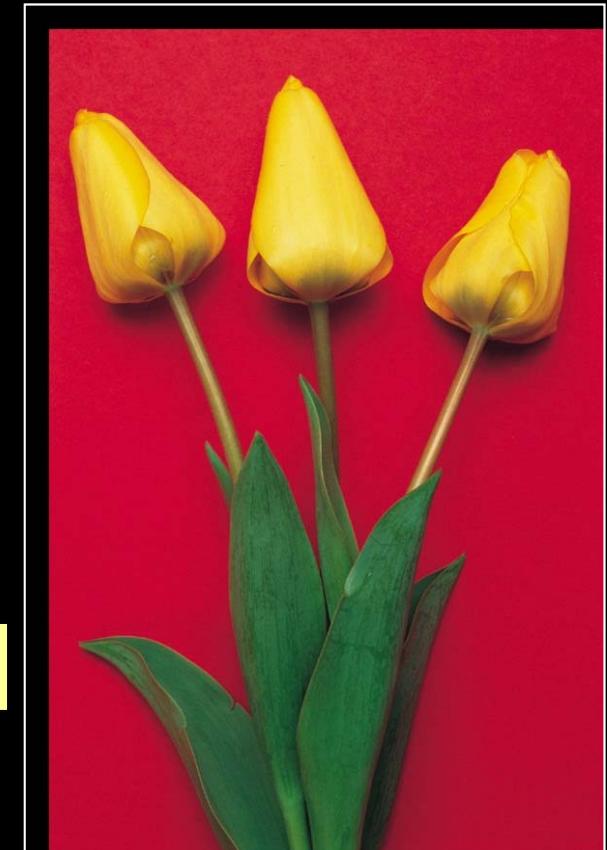
$$\theta[n, m] = \psi[n, m] * h[n, m] \xrightarrow{DSFT} O[u, v] = \Psi[u, v] \cdot H[u, v]$$

- Las máscaras son versiones truncadas de las respuestas al impulso correspondientes a los filtros frecuenciales.

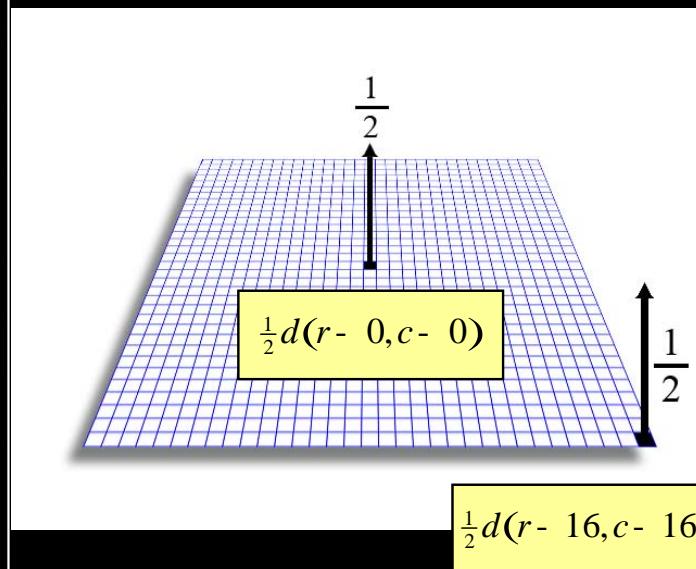
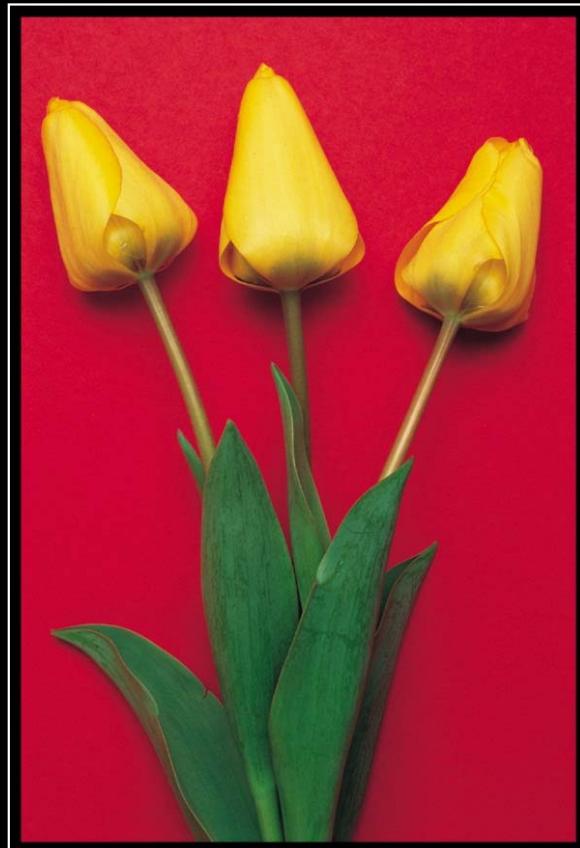
- Convolución con un impulso: desplazamiento



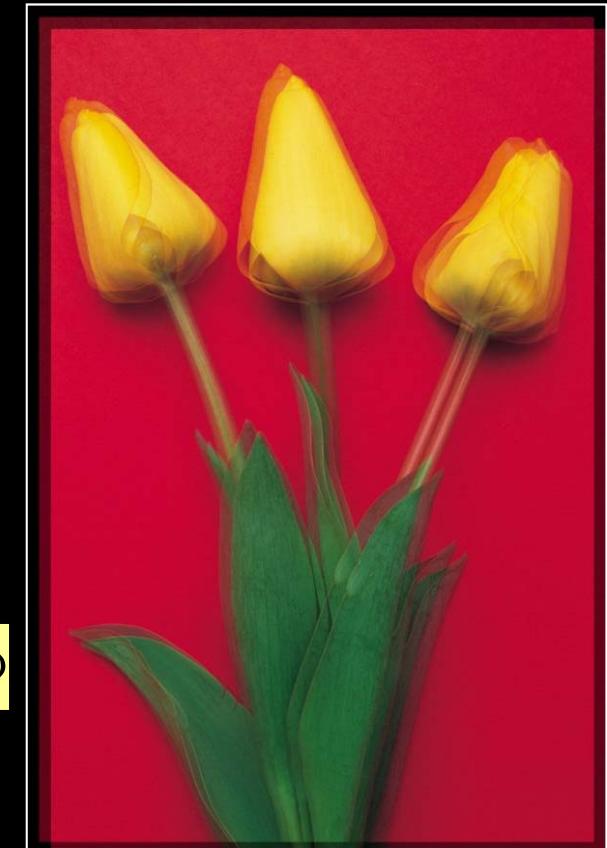
Desplazamiento abajo-derecha por 16 pixels.



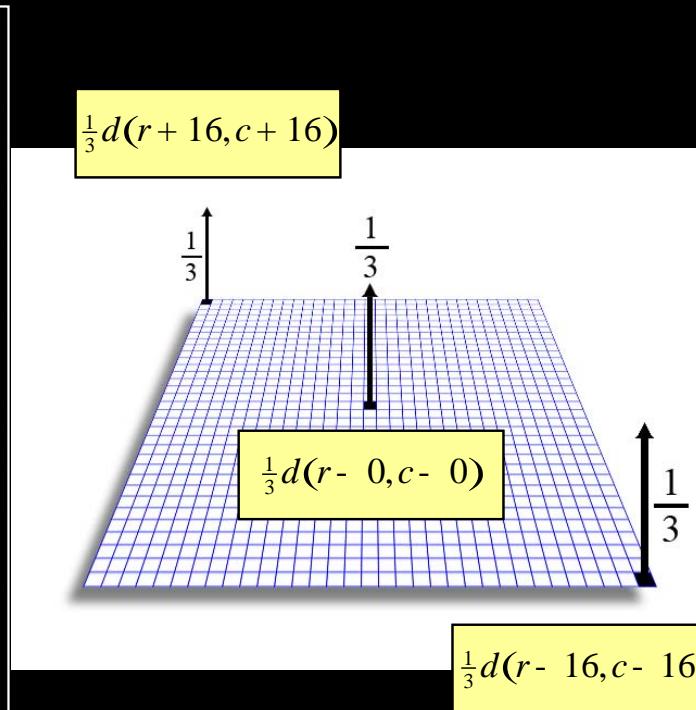
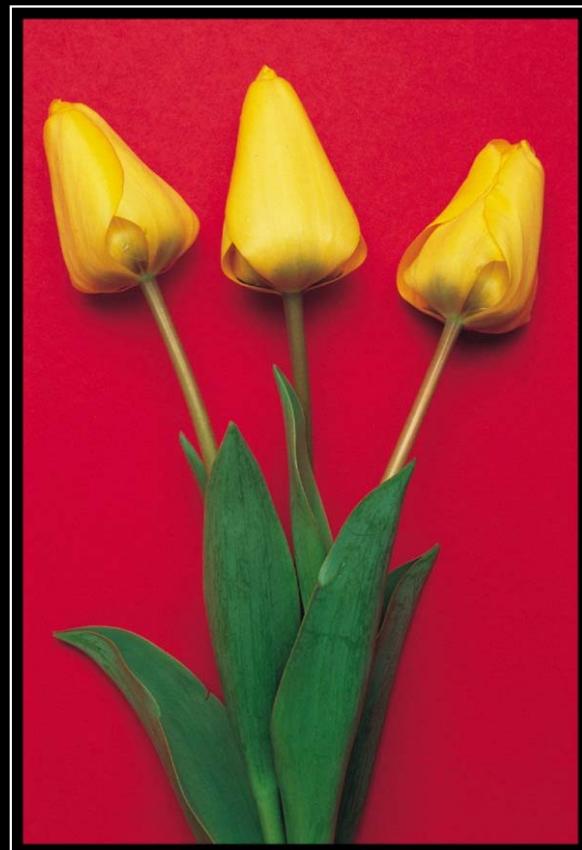
- Convolución con dos impulsos: desplazamientos + superposición



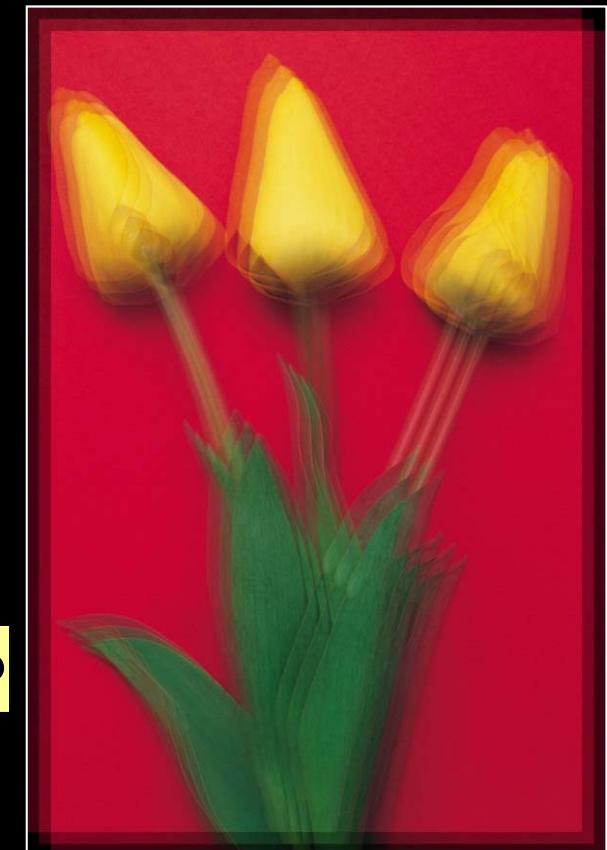
Dos copias, una
desplazada y otra sin
desplazar, promediadas.



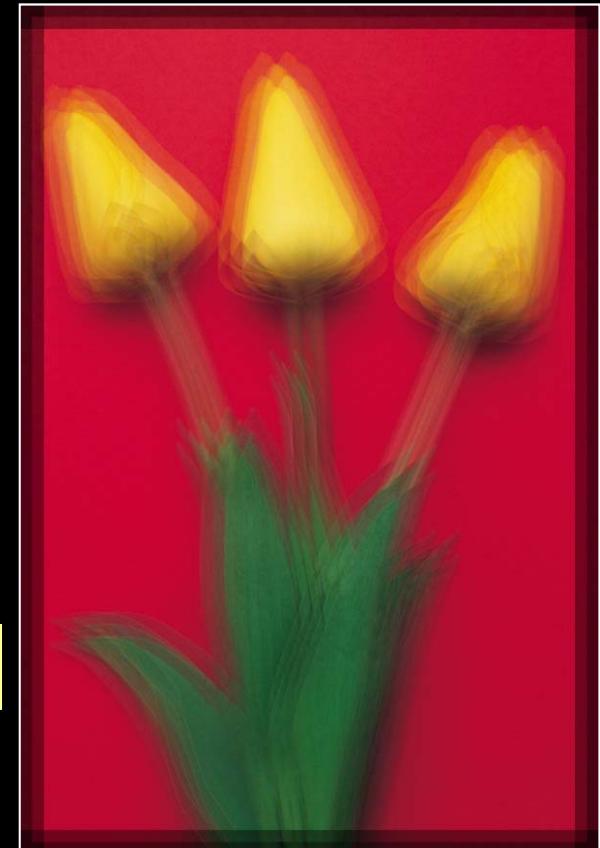
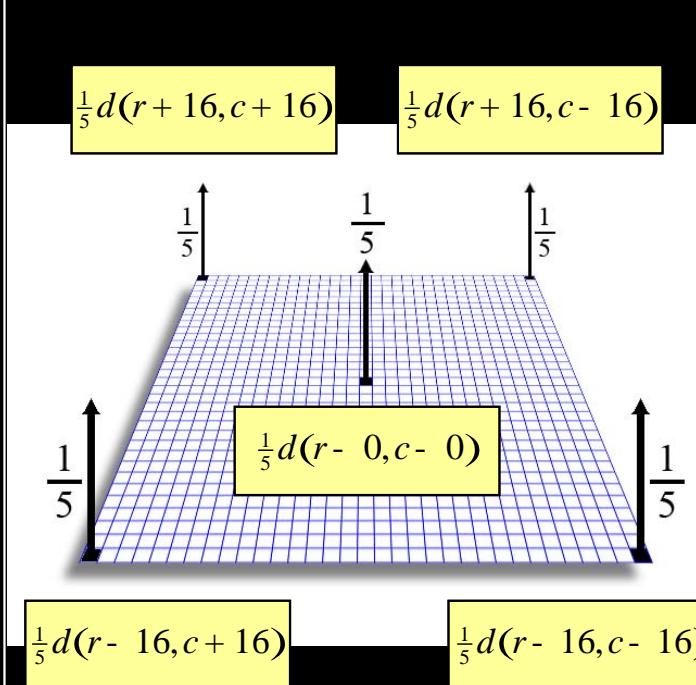
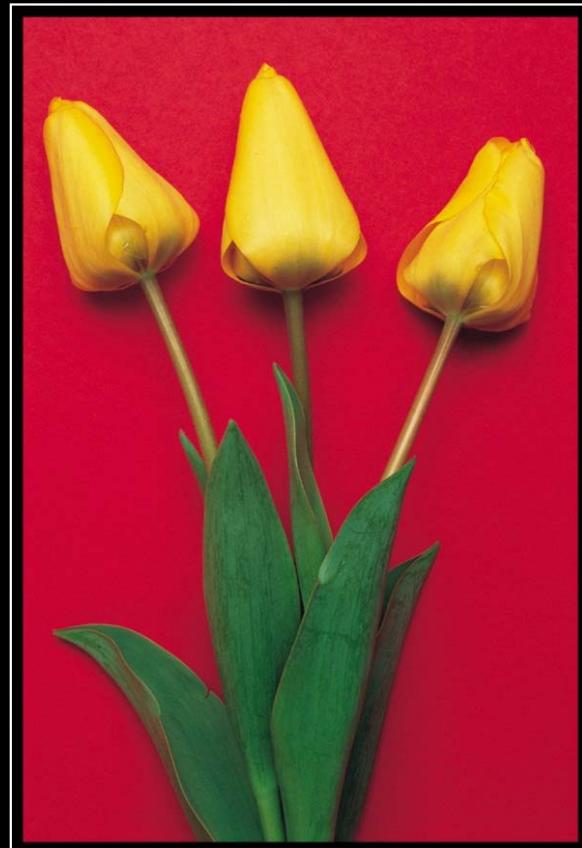
- Convolución con tres impulsos: desplazamientos + superposición



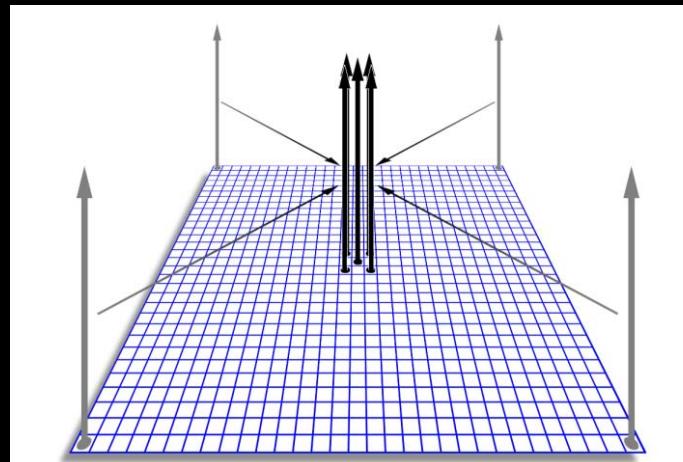
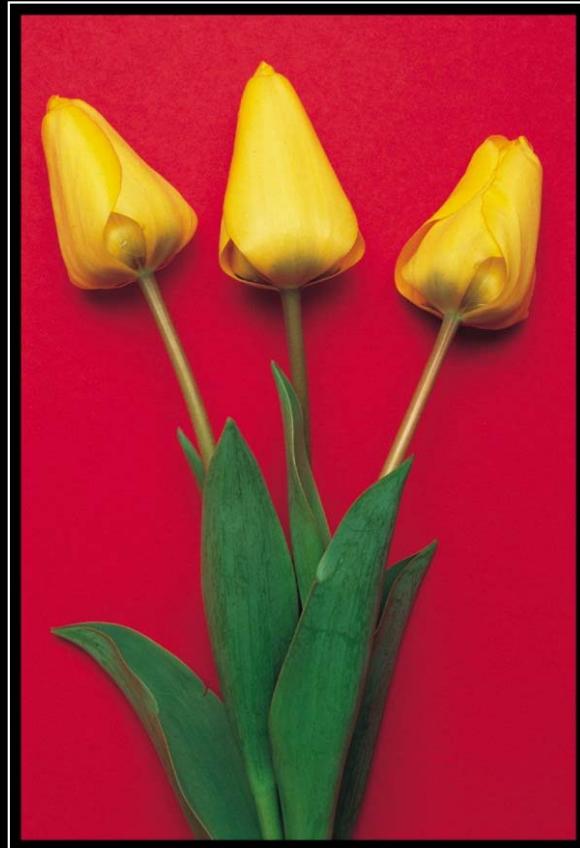
Tres copias, dos
desplazadas y otra sin
desplazar, promediadas



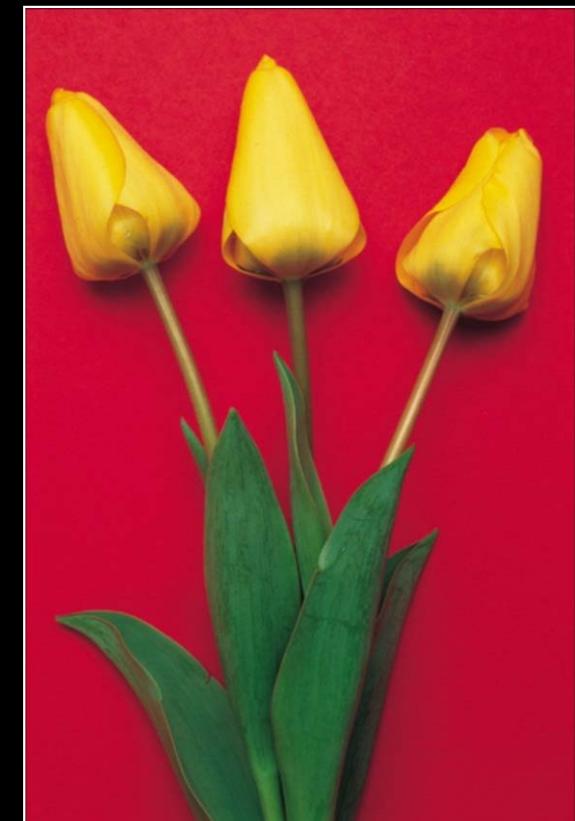
- Convolución con cinco impulsos: desplazamientos + superposición



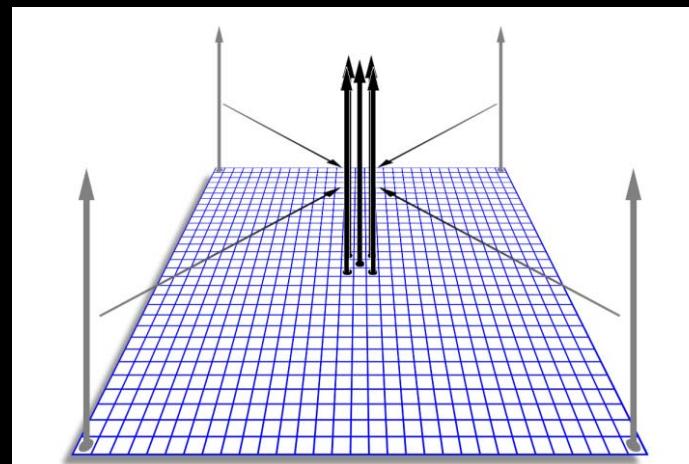
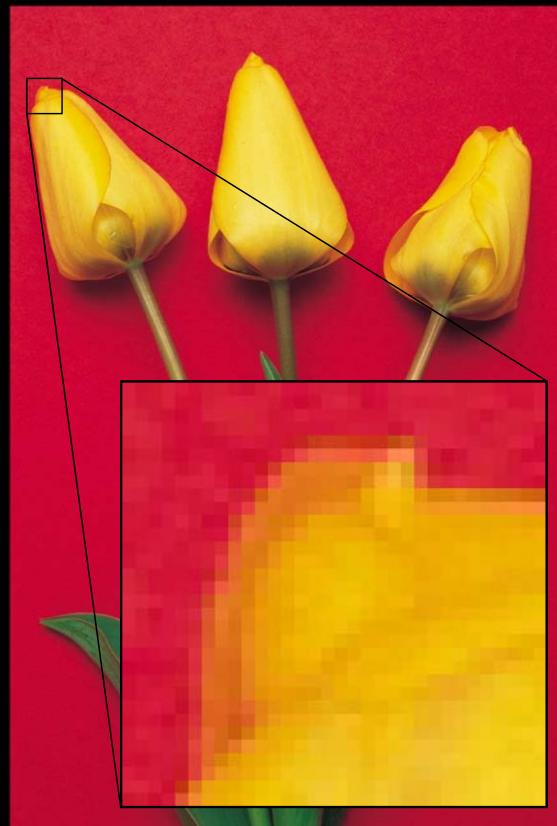
- Convolución con cinco impulsos: desplazamientos + superposición
 - Aplicación para suavizado



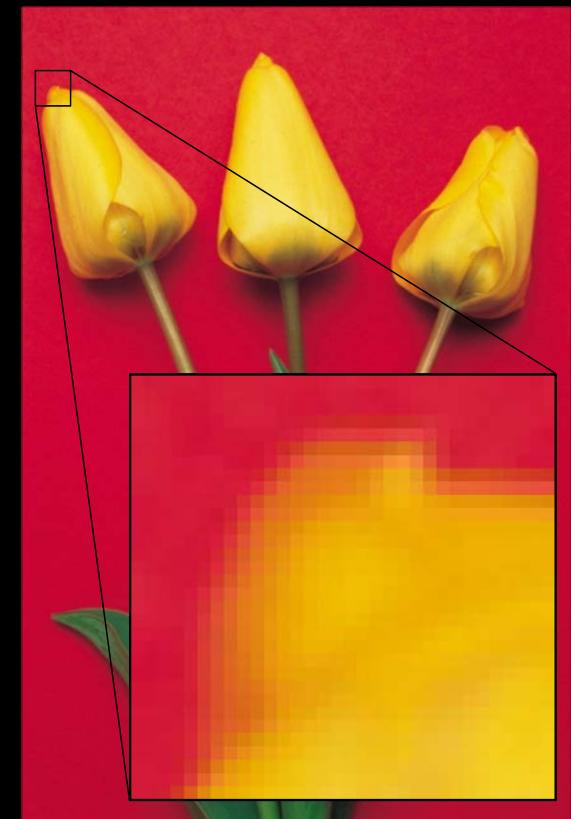
Si colocamos los impulsos de manera adyacente , la convolucion se convierte en un filtro de emborronado.



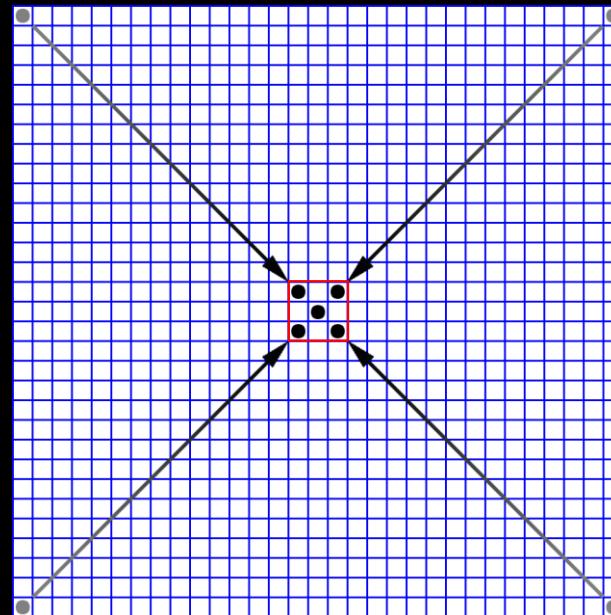
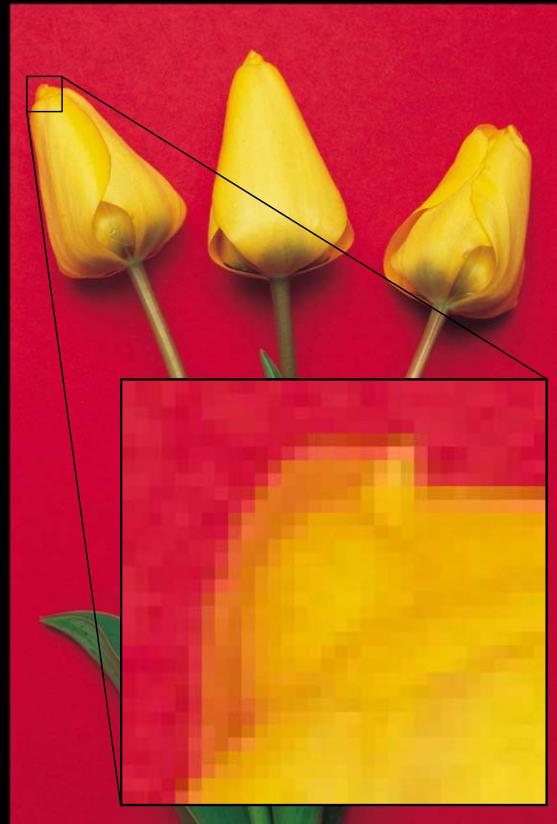
- Convolución con cinco impulsos: desplazamientos + superposición
 - Aplicación para suavizado



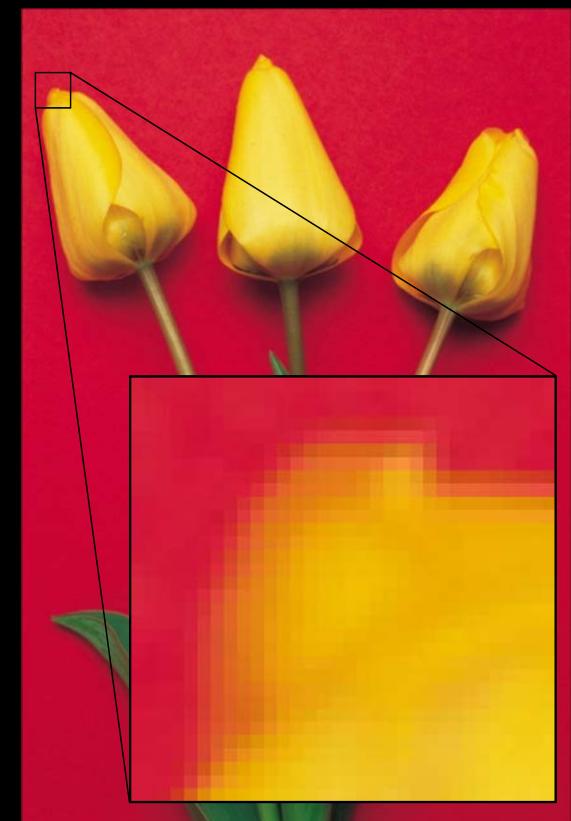
Los impulsos se convierten en valores (selección) de datos en una vecindad de 3×3



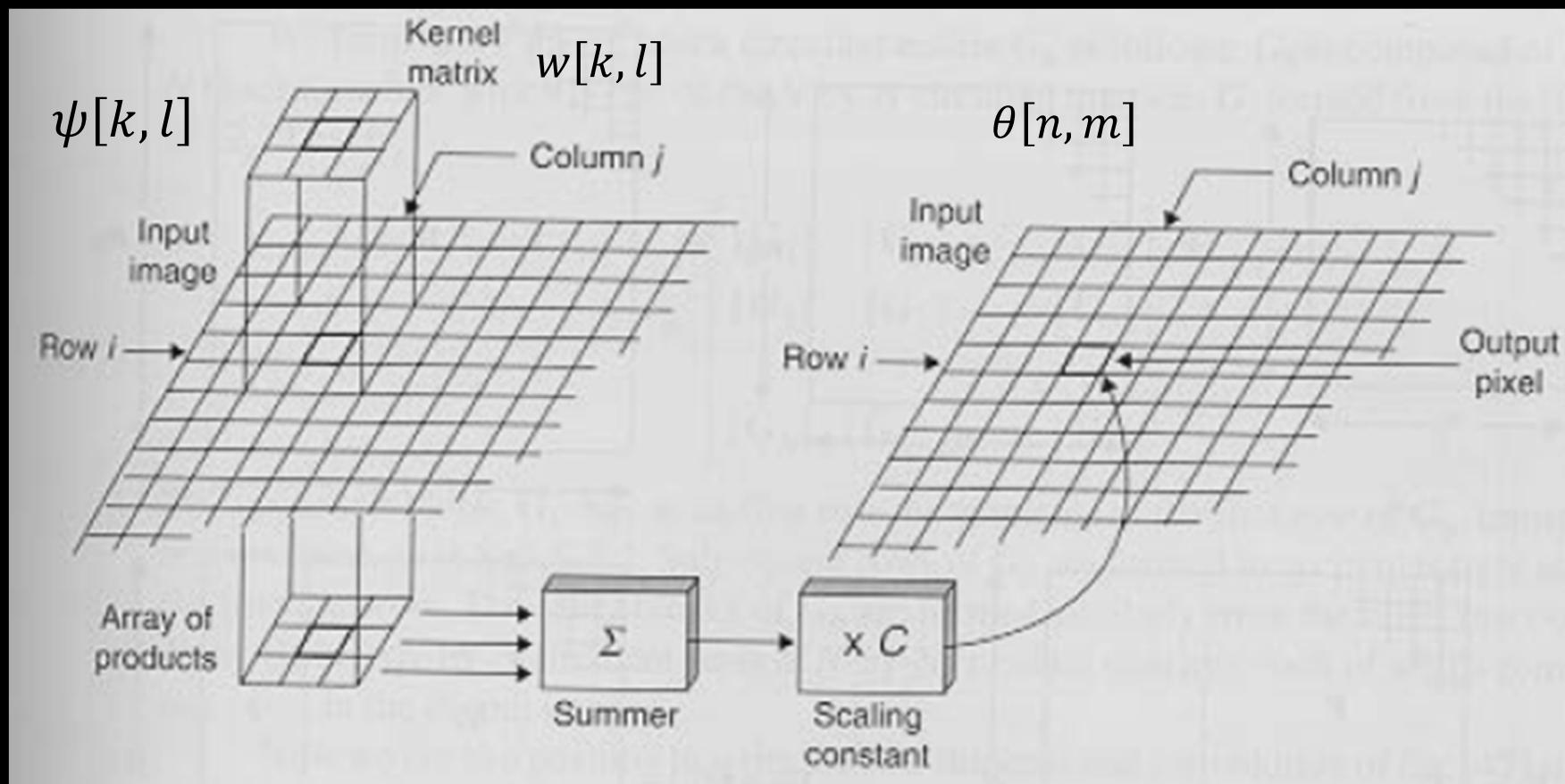
- Convolución con cinco impulsos: desplazamientos + superposición
 - Aplicación para suavizado



La máscara de convolución tiene cinco elementos de valor $1/5$ y cuatro de valor 0.



- Correlación (utilizando la máscara $w[k, l]$)



$$\theta[n, m] = \sum_{k=-a}^a \sum_{l=-b}^b \psi[k, l] \cdot w[k, l]$$

- Correlación (utilizando la máscara $w[k, l]$)



Utilizada con frecuencia en aplicaciones donde se necesita medir la similitud entre imágenes o partes de ellas (e.g., reconocimiento de patrones).



- Lineal: la salida es una suma ponderada de los pixeles de entrada (linealidad)

$$\theta[n, m] = \sum_{k=-a}^a \sum_{l=-b}^b \psi[k, l] \cdot w[k, l]$$

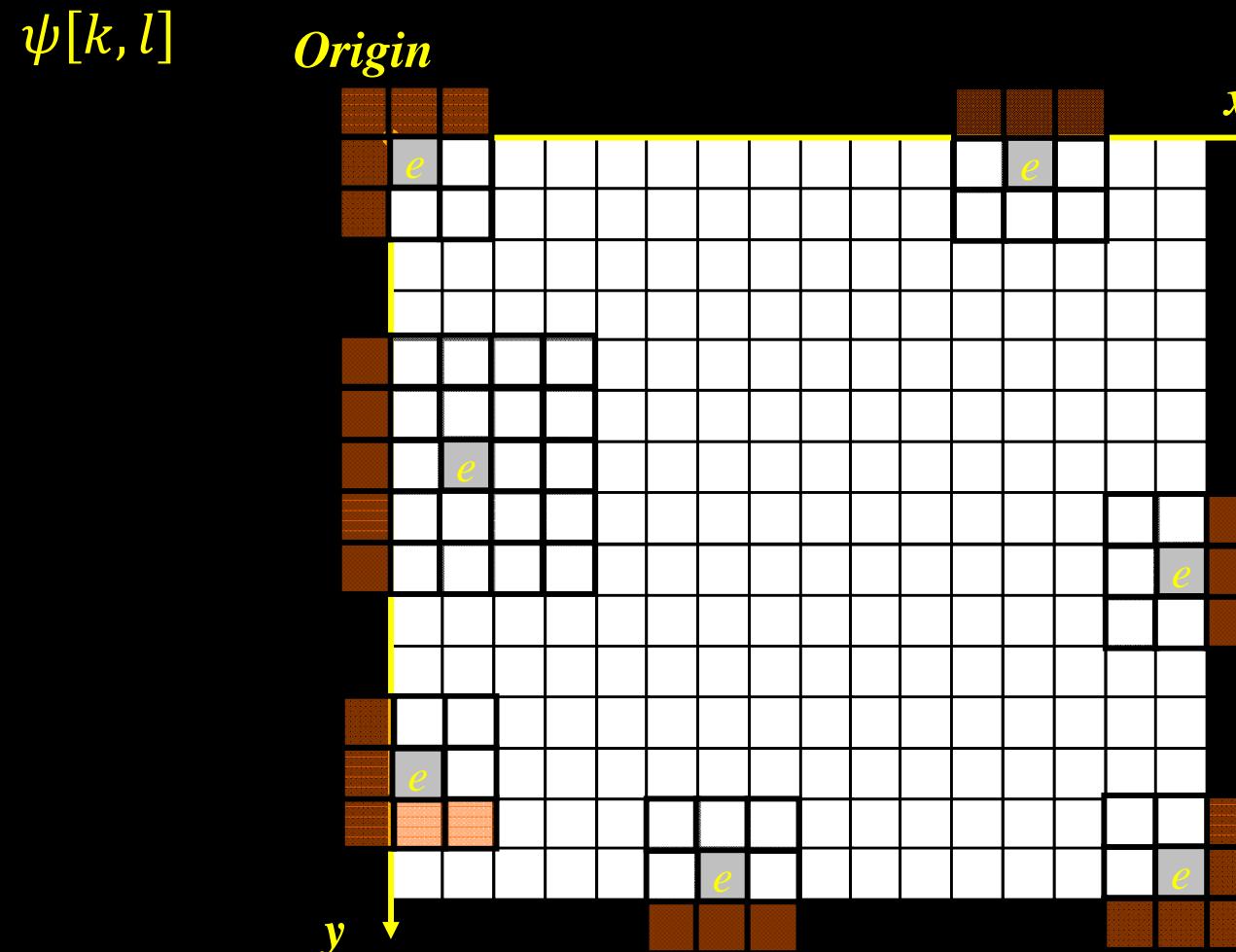
→ Permite descomposición en filas/columnas

- No-lineal: si no satisface la condición de linealidad

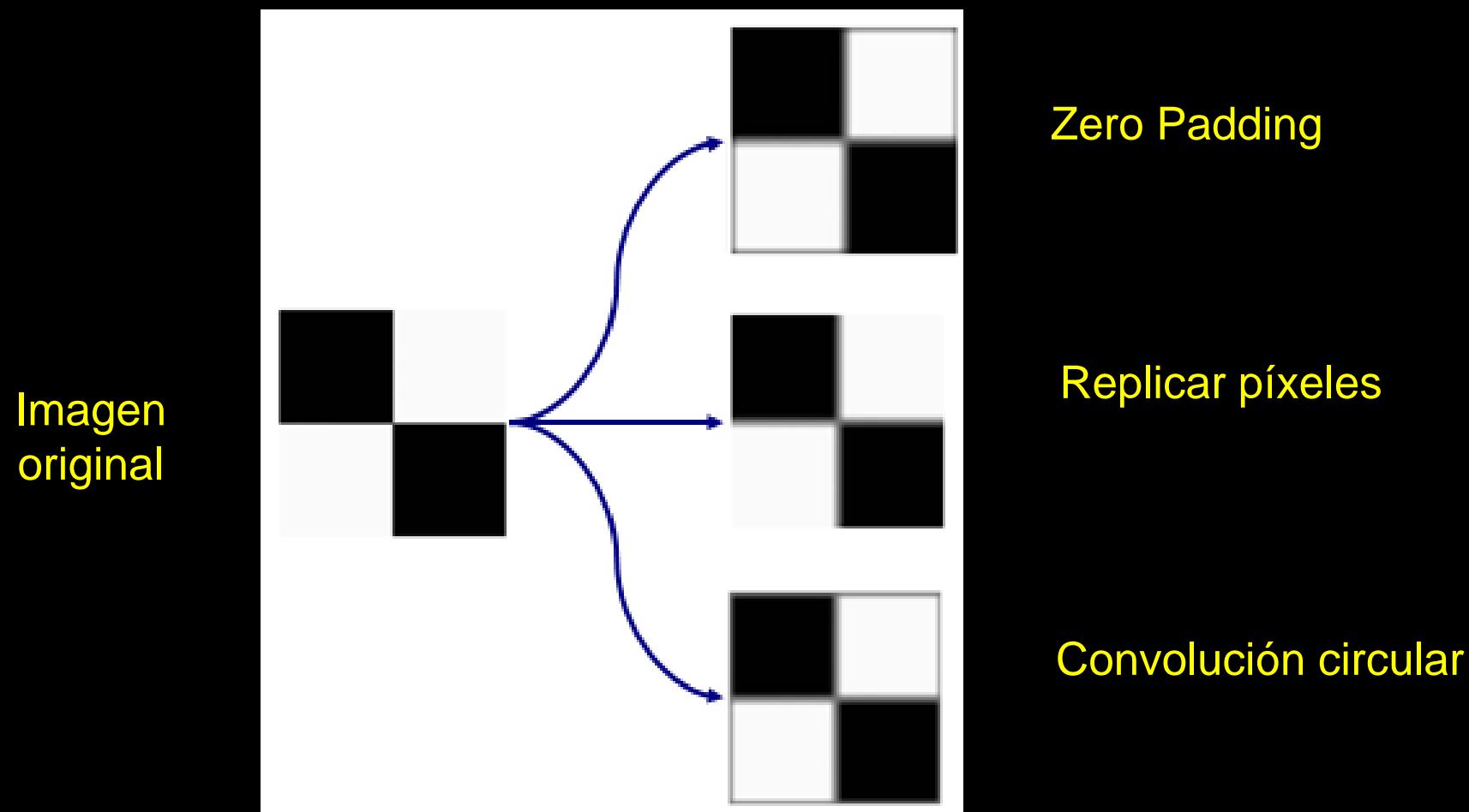
$$\theta[n, m] = \max(\psi[k, l], w[k = -a, \dots, a, l = -b, \dots, b])$$

→ Tratamiento más complejo

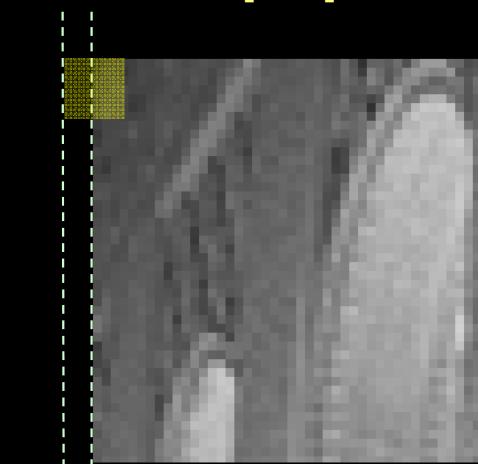
- Situación en los bordes: La aplicación de un filtro de $N' \times N'$ da lugar a una imagen con $(N' - 1)/2$ píxeles menos por cada lado.



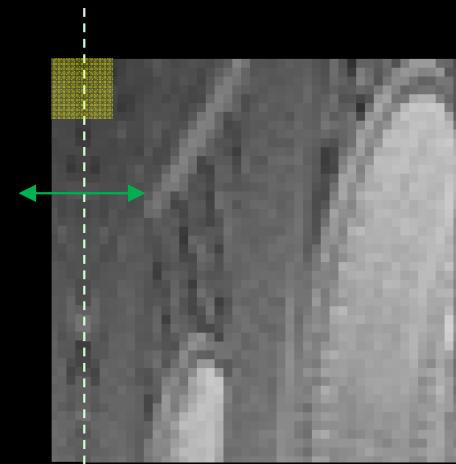
- Si se desea que la imagen resultante tenga igual dimensión hay varias opciones: filtrado parcial, convolución circular o “padding”.



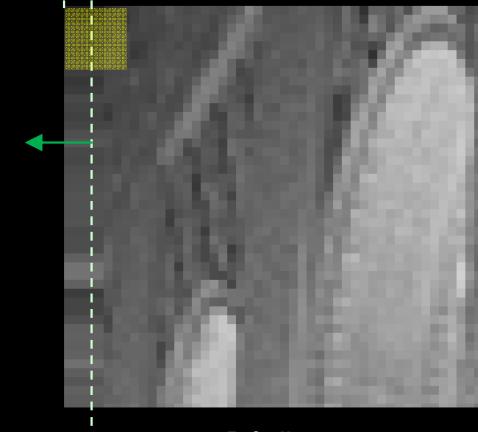
$\psi[n, m]$



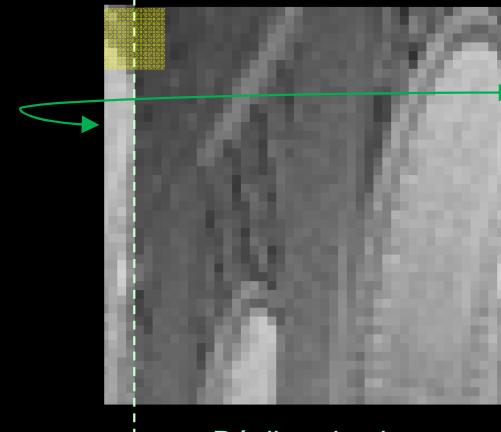
Ceros u otro valor



Réplica simétrica



Réplica



Réplica circular

- Separabilidad: permite reducir notablemente el número de operaciones (\otimes representa el producto externo, tensorial o de Kronecker).

$$\mathbf{w} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1N'} \\ w_{21} & w_{21} & \dots & w_{1N'} \\ \dots & \dots & \dots & \dots \\ w_{M'1} & w_{M'1} & \dots & w_{M'N'} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{M'} \end{bmatrix} \otimes [x_1 \quad x_2 \quad \dots \quad x_{N'}] = \mathbf{x} \otimes \mathbf{y}$$

$$\Rightarrow w_{ij} = y_i \cdot x_j$$

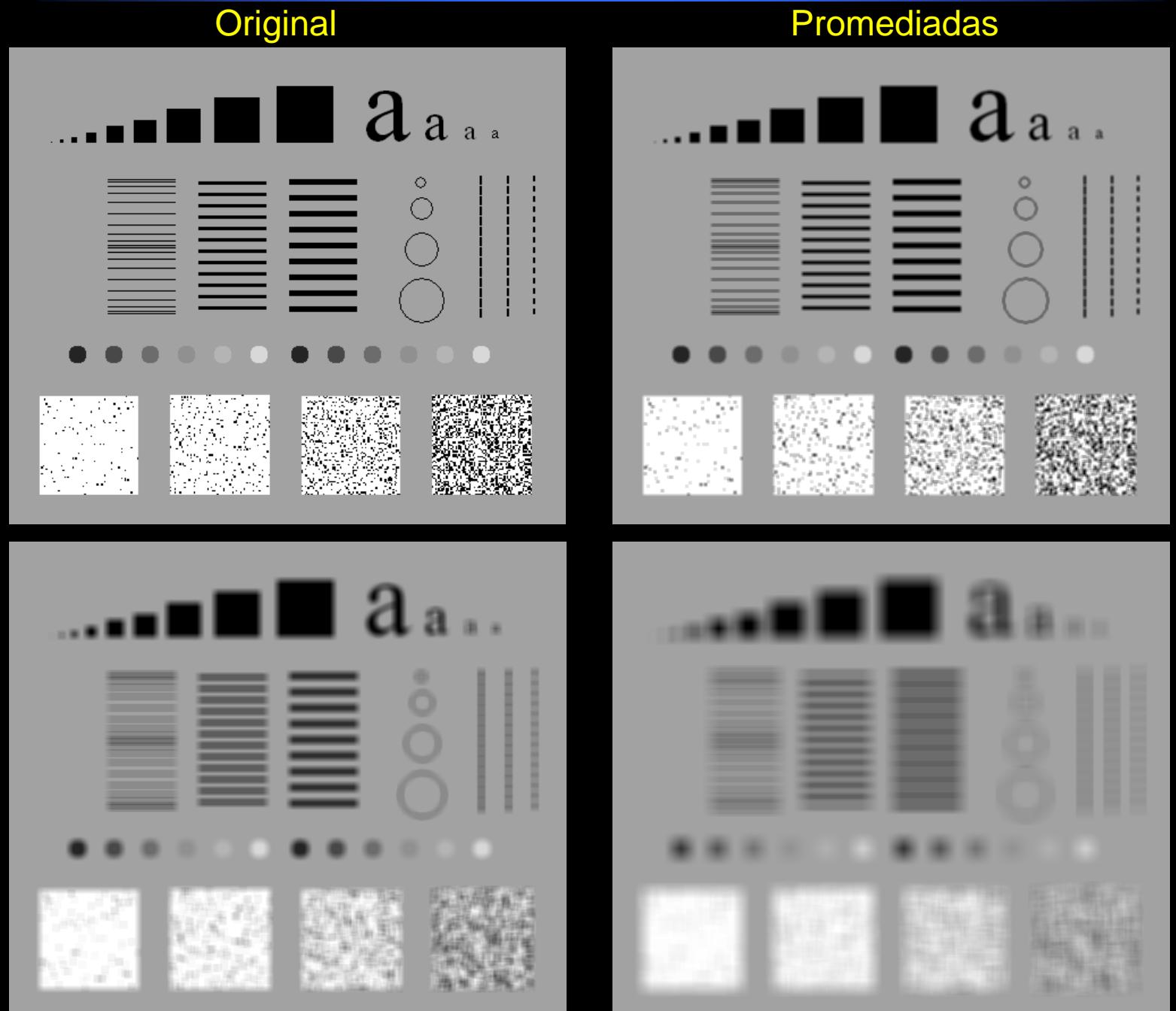
→ Solamente aplicable en filtros lineales

1. Introducción
2. Aspectos operativos
3. Filtrado espacial
 - Suavizado
 - Realce de contornos
 - Detección y localización de bordes
4. Ajustes geométricos
5. Restauración lineal

- Suavizado (smoothing)
 - Filtros utilizados para suavizar, emborronar (*blurring*) o eliminar ruido
 - Los elementos de la máscara han de ser positivos
 - La suma de los elementos de la máscara es 1.
- Filtrado en el dominio espacial
 - Diseño intuitivo de máscaras: promediado (*average*) y promedio ponderado (*weighted average*)

$$\frac{1}{9} \cdot \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \frac{1}{9} \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \cdot [1 \quad 1 \quad 1]$$

$$\frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} = \frac{1}{16} \cdot \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \cdot [1 \quad 2 \quad 1]$$



- La máscara de filtrado promediado ponderada es un caso particular de filtro binomial, familia de filtros separables resultante de la aplicación sucesiva y en ambas dimensiones de la máscara:

$$B_x = B_y^T = \frac{1}{2} \cdot [1 \quad 1]$$

- Si se aplica ‘n’ veces, equivale a aplicar en cada dimensión las máscaras

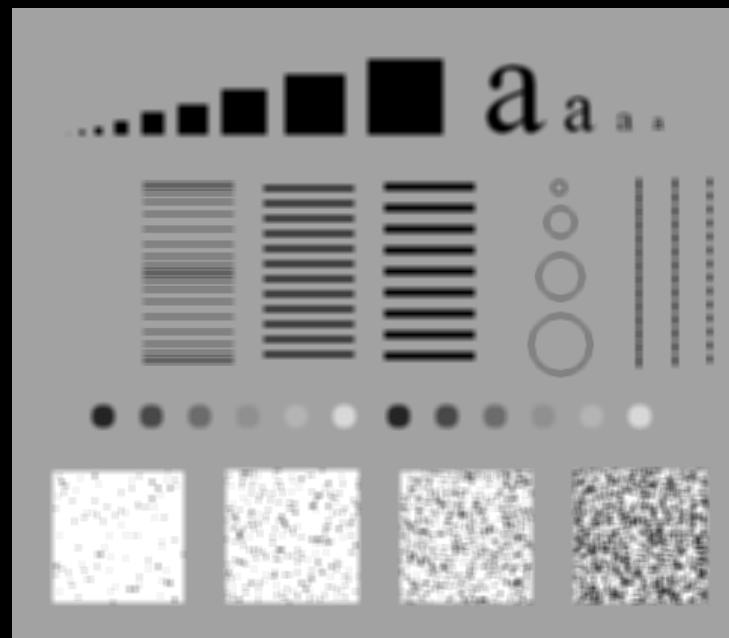
$$B_x^2 = \frac{1}{2^2} \cdot [1 \quad 2 \quad 1], B_x^3 = \frac{1}{2^3} \cdot [1 \quad 3 \quad 3 \quad 1], B_x^4 = \frac{1}{2^4} \cdot [1 \quad 4 \quad 6 \quad 4 \quad 1], \text{etc.}$$

- El resultado en 2D son máscaras o filtros de la forma:

$$B^n = B_y^n \cdot B_x^n, B^2 = \frac{1}{2^2} \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \cdot \frac{1}{2^2} [1 \quad 2 \quad 1] = \frac{1}{16} \cdot \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

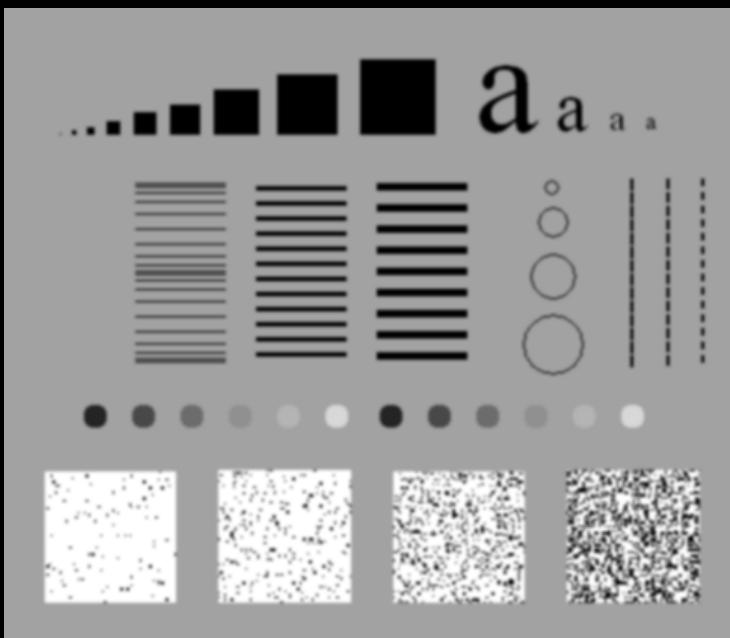
Promediado

5x5

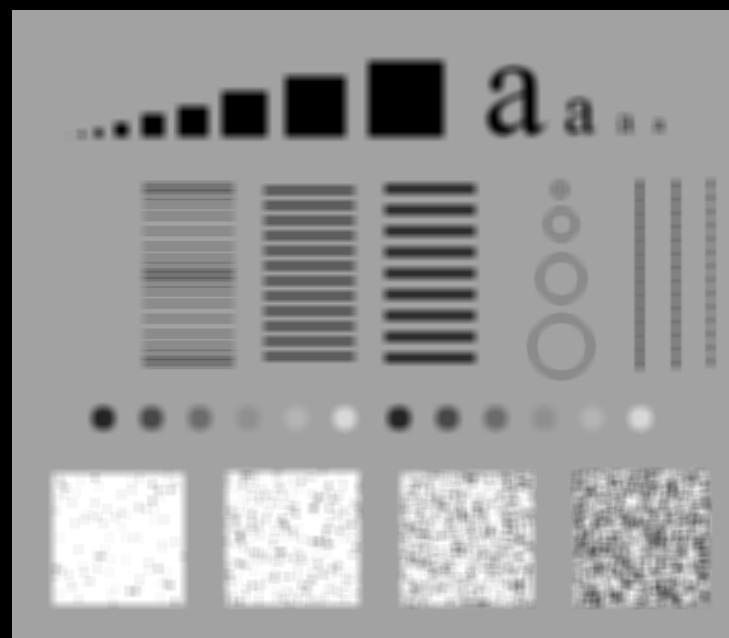


Binomial

5x5



7x7



7x7

Original

Esto es un ejemplo de texto escrito con caracteres no suavizados, para observar el efecto que sobre ellos tiene la aplicación de un filtro de suavizado.

Promediado
3x3

Esto es un ejemplo de texto escrito con caracteres no suavizados, para observar el efecto que sobre ellos tiene la aplicación de un filtro de suavizado.

Binomial
3x3

Esto es un ejemplo de texto escrito con caracteres no suavizados, para observar el efecto que sobre ellos tiene la aplicación de un filtro de suavizado.

Esto es
escrito

Esto es
escrito

7x7

Esto es
escrito

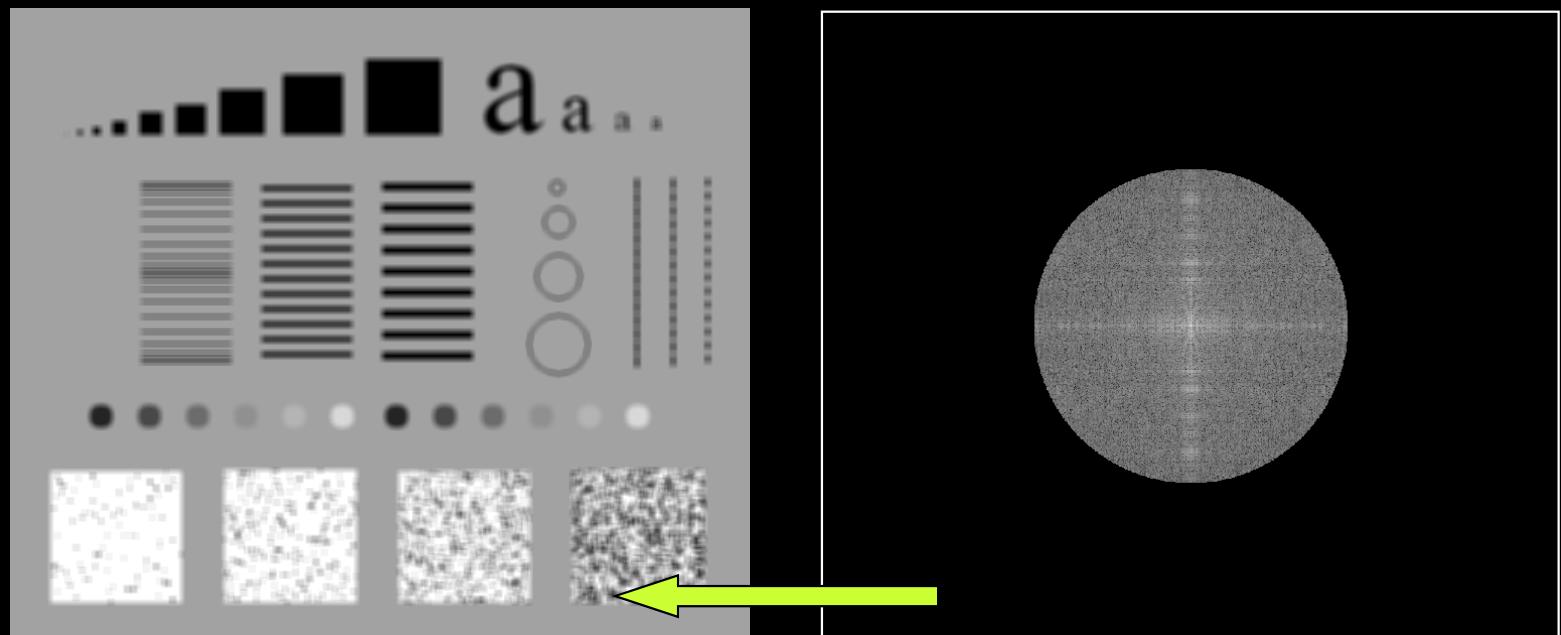
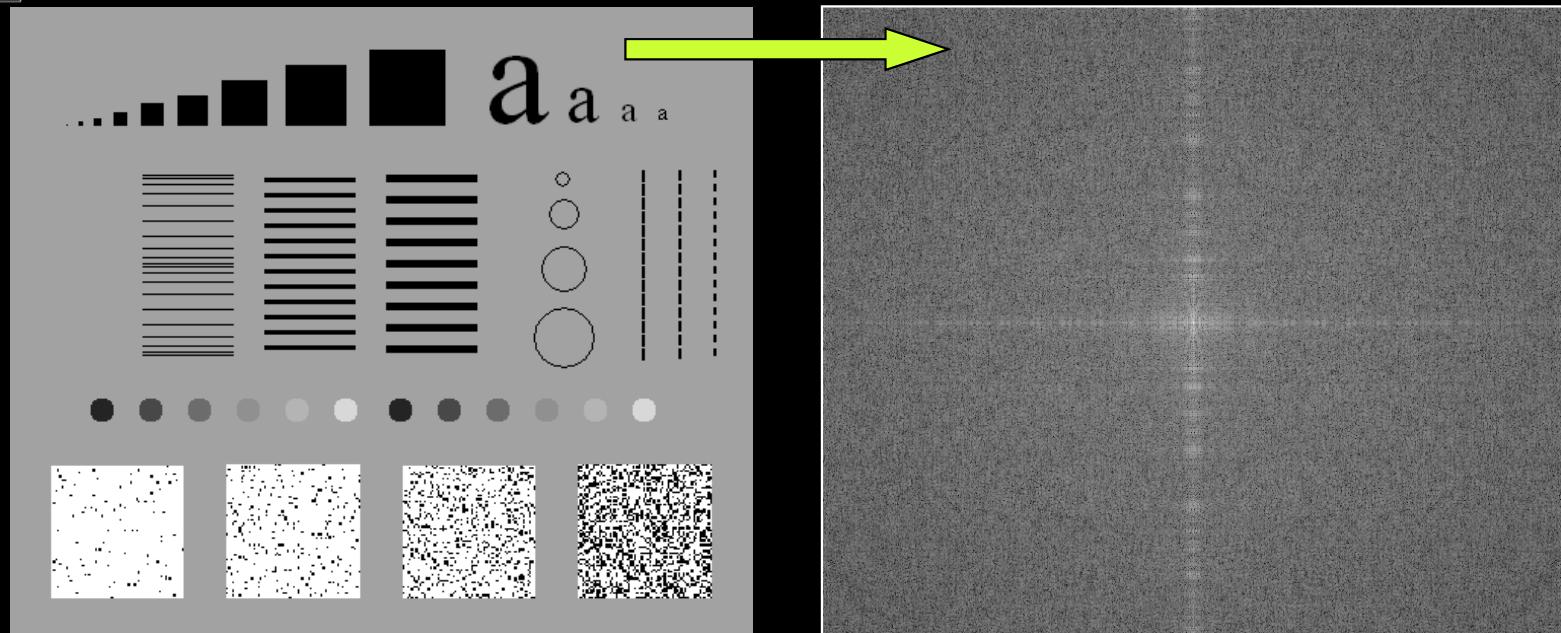
7x7

- Diseño de filtros motivación frecuencial

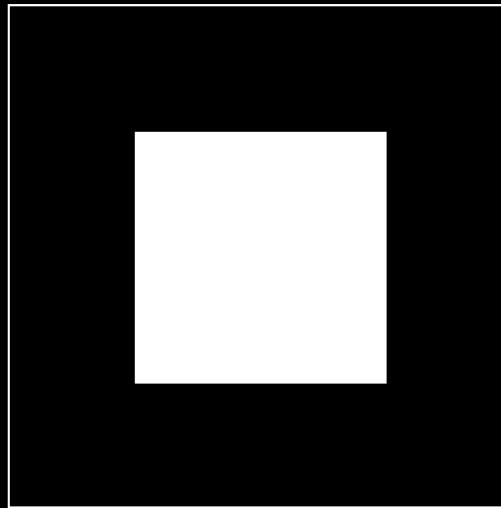
- La operación de filtrado puede llevarse a cabo bien en el dominio “natural” o del píxel o bien en el dominio frecuencial, aplicando la propiedad de convolución:

$$\theta[n, m] = \psi[n, m] * h[n, m] \xrightarrow{DSFT} O[u, v] = \Psi[u, v] \cdot H[u, v]$$

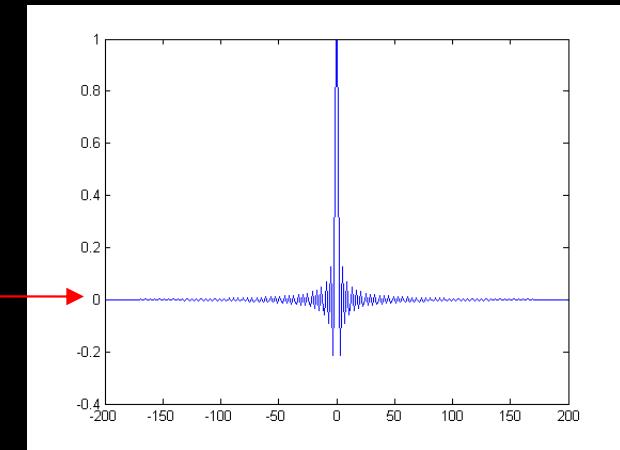
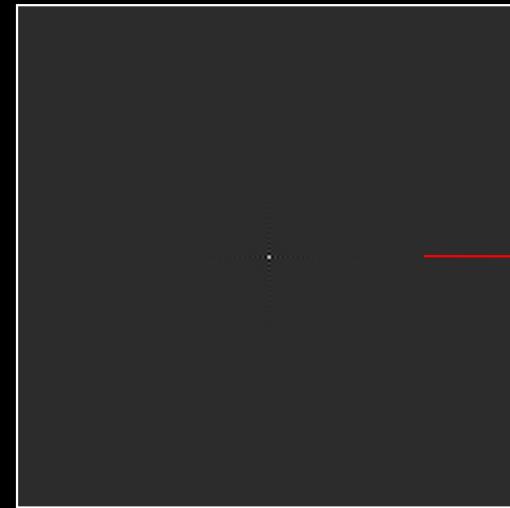
- Las máscaras son versiones truncadas de las respuestas al impulso correspondientes a los filtros frecuenciales.
 - Ejemplo: filtros paso bajo



$H[u, v]$



$h[n, m]$

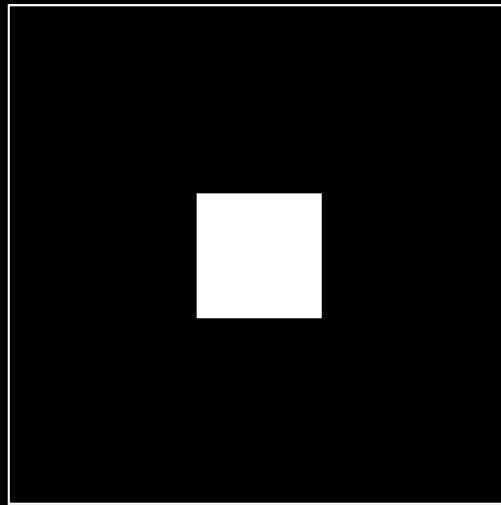


$$f_c = 1/4$$

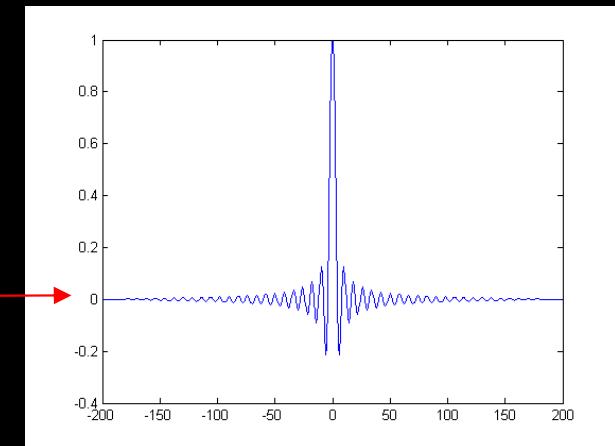
$$h_{3 \times 3}[n, m] = \frac{1}{625} \begin{bmatrix} 49 & 77 & 49 \\ 77 & 121 & 77 \\ 49 & 77 & 49 \end{bmatrix}$$

$$h_{7 \times 7}[n, m] = \frac{1}{305} \begin{bmatrix} 4 & 0 & -12 & -19 & -12 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -12 & 0 & 36 & 57 & 36 & 0 & -12 \\ -19 & 0 & 57 & 89 & 57 & 0 & -19 \\ -12 & 0 & 36 & 57 & 36 & 0 & -12 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & -12 & -19 & -12 & 0 & 4 \end{bmatrix}$$

$H[u, v]$



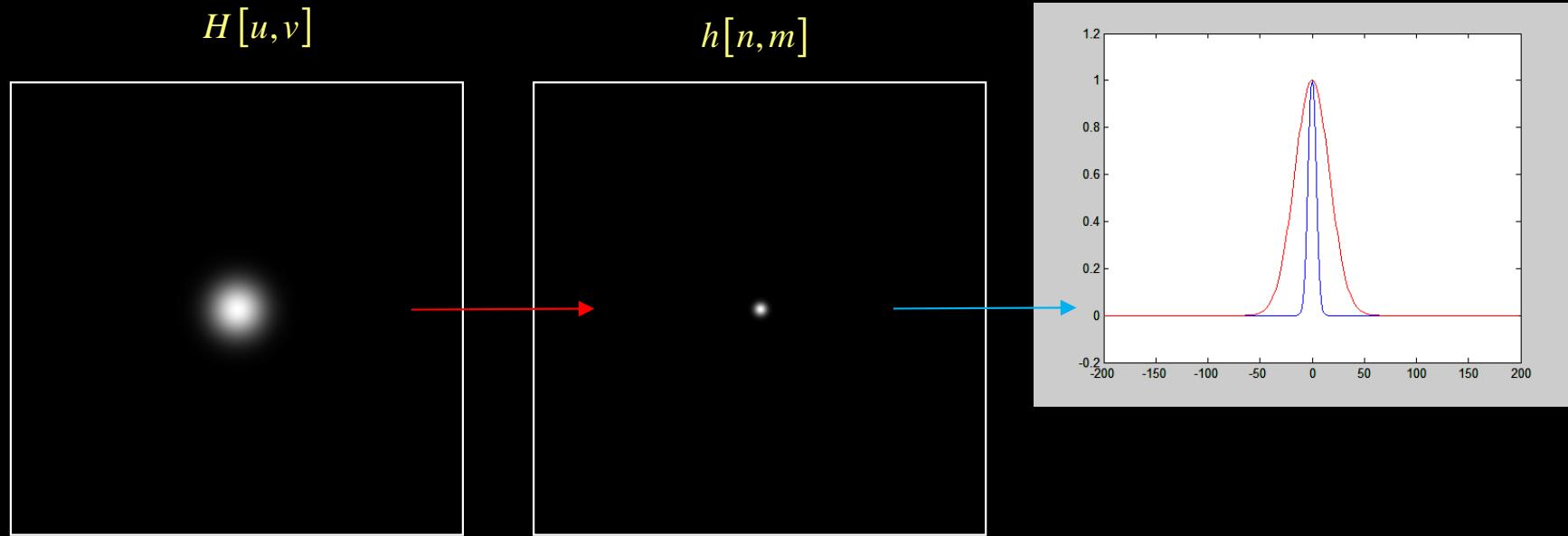
$h[n, m]$



$$f_c = 1/8$$

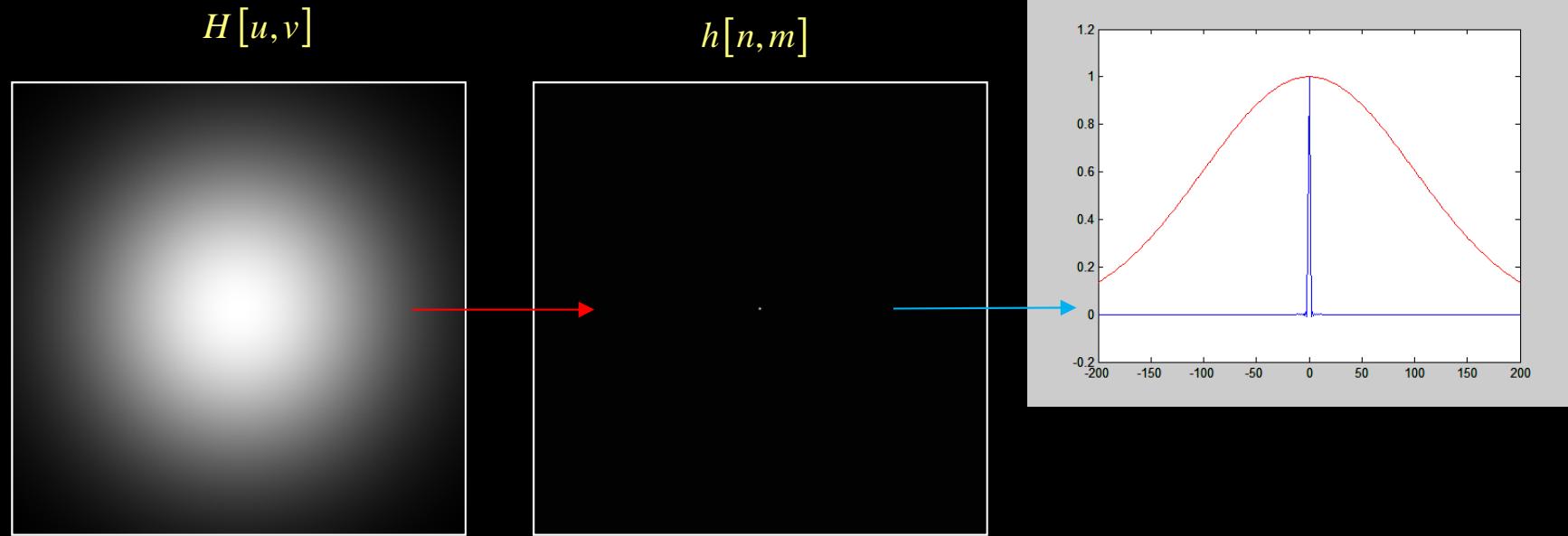
$$h_{3 \times 3}[n, m] = \frac{1}{871} \begin{bmatrix} 90 & 100 & 90 \\ 100 & 111 & 100 \\ 90 & 100 & 90 \end{bmatrix}$$

$$h_{7 \times 7}[n, m] = \frac{1}{1945} \begin{bmatrix} 8 & 17 & 24 & 27 & 24 & 17 & 8 \\ 17 & 36 & 51 & 57 & 51 & 36 & 17 \\ 24 & 51 & 72 & 80 & 72 & 51 & 24 \\ 27 & 57 & 80 & 89 & 80 & 57 & 27 \\ 24 & 51 & 72 & 80 & 72 & 51 & 24 \\ 17 & 36 & 51 & 57 & 51 & 36 & 17 \\ 8 & 17 & 24 & 27 & 24 & 17 & 8 \end{bmatrix}$$



$$f_c = \sigma_f = 1/24 \quad \sigma \cdot \sigma_f = \frac{1}{2\pi} \quad \sigma = 3.82$$

$$h_{3 \times 3}[n, m] = \frac{1}{1077} \begin{bmatrix} 117 & 121 & 117 \\ 121 & 125 & 121 \\ 117 & 121 & 117 \end{bmatrix} \quad h_{2\sigma \times 2\sigma}[n, m] = h_{17 \times 17}[n, m] = \dots$$



$$f_c = \sigma_f = 1/4$$

$$\sigma \cdot \sigma_f = \frac{1}{2\pi}$$

$$\sigma = 0.64$$

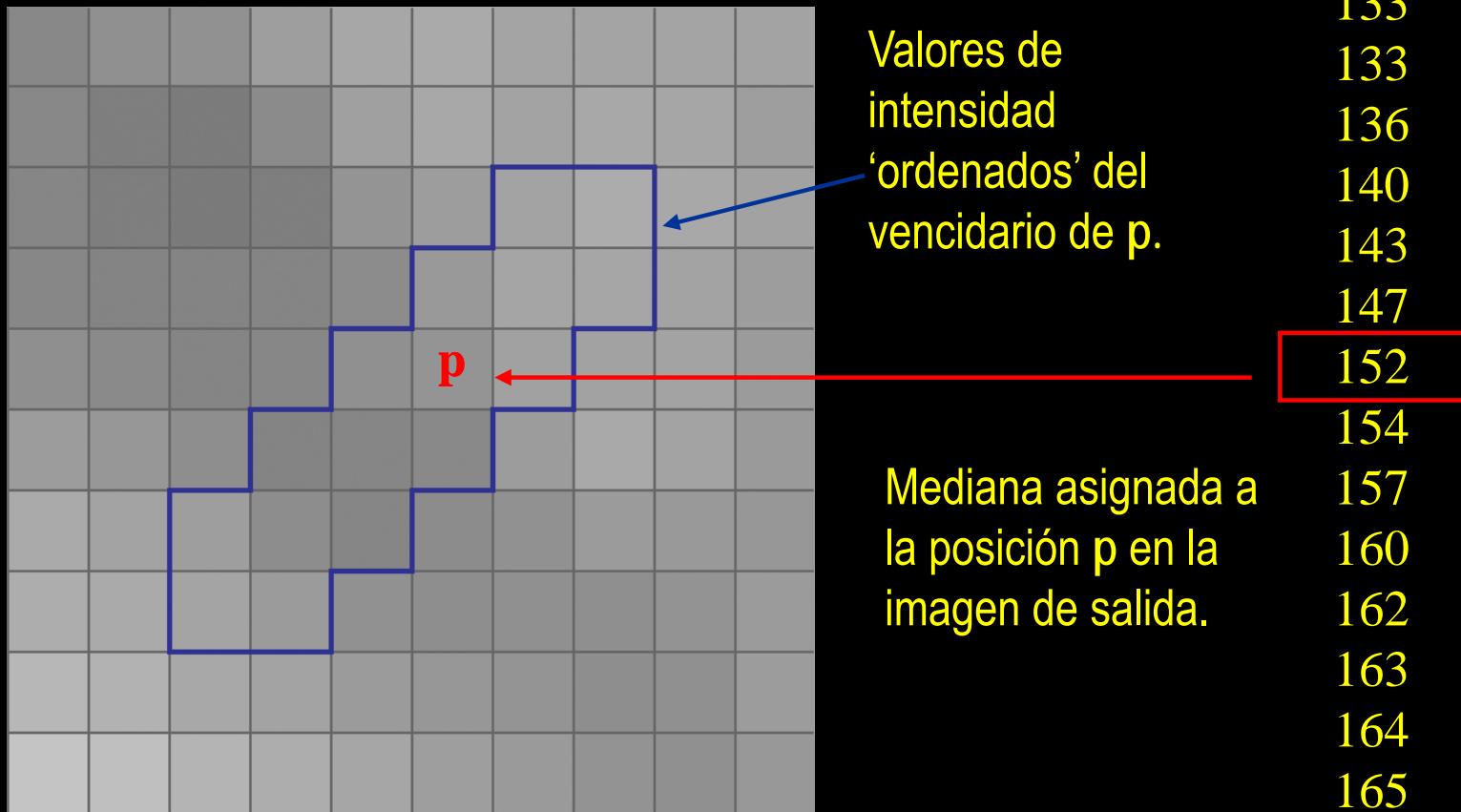
$$h_{2\sigma x 2\sigma} [n,m] = h_{3x3} [n,m] = \frac{1}{315} \begin{bmatrix} 13 & 38 & 13 \\ 38 & 111 & 38 \\ 13 & 38 & 13 \end{bmatrix}$$

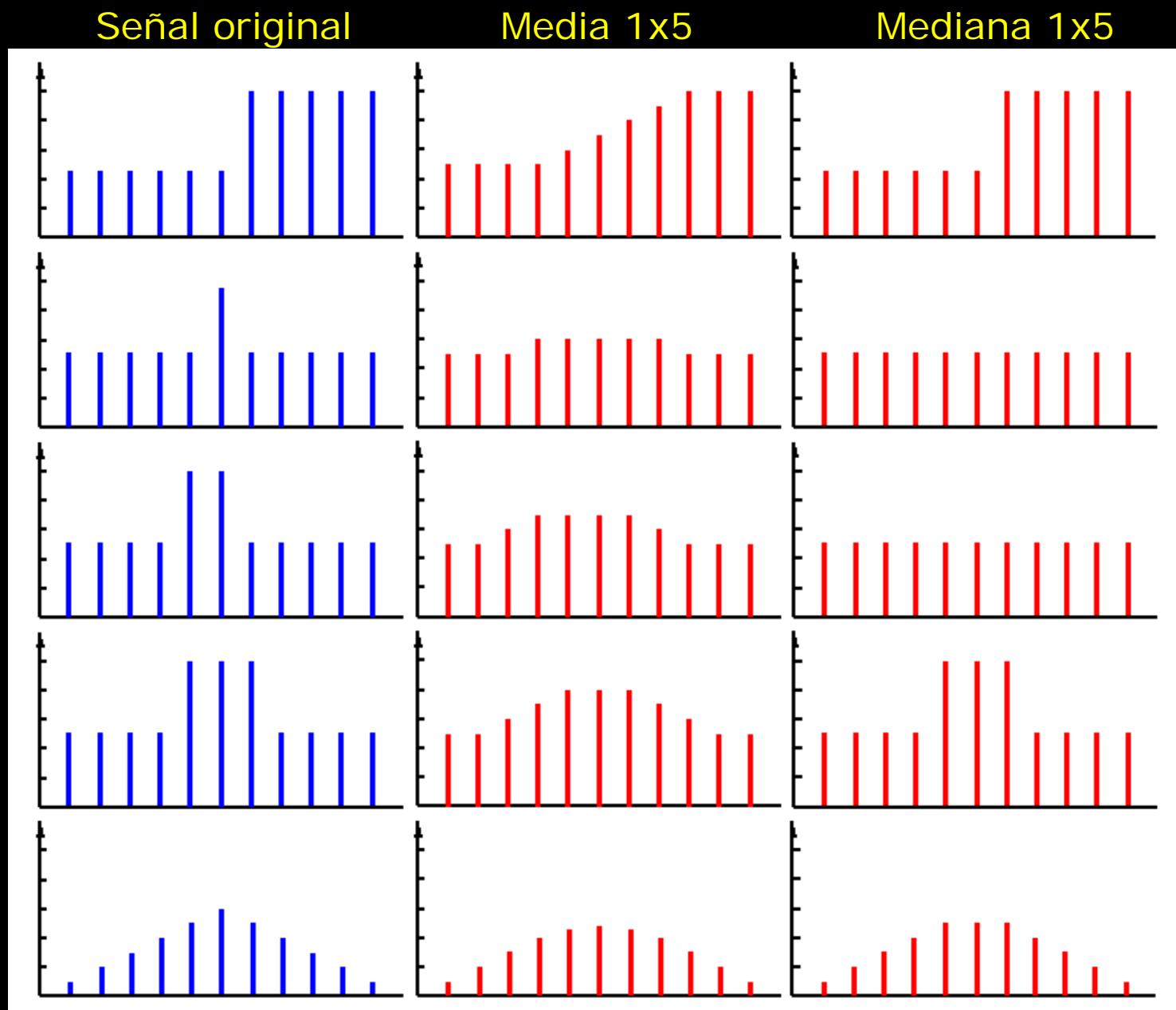
- Filtrado no lineal

- El concepto de máscara y su aplicación es similar, pero las operaciones que se realizan para obtener el resultado no son lineales.
- Preserva bordes.
- Efectivo para eliminar ruido impulsivo y sal-pimienta
- Diseño complejo.
- Es un caso particular de los filtros de estadísticas de orden (*order-statistics filters*): filtros de máximo, filtros de mínimo, filtros de posición, filtros de mediana.

- Filtrado no lineal
 - Filtro mediana: ideado para eliminar ruido impulsivo (o sal&pimienta) respetando los contornos (rampas y discontinuidades).

131
133
133
136
140
143
147
152
154
157
160
162
163
164
165
171



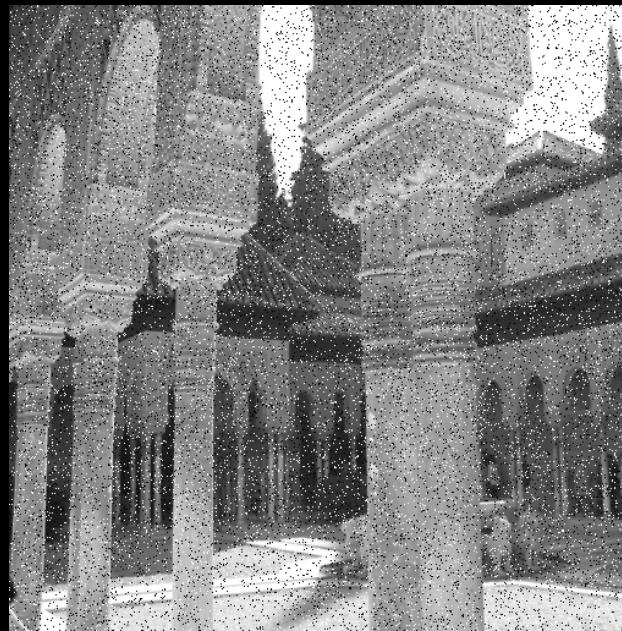


Fuente: http://fourier.eng.hmc.edu/e161/lectures/smooth_sharpen/node3.html

SUAVIZADO

Eliminación de ruido (sal y pimienta)

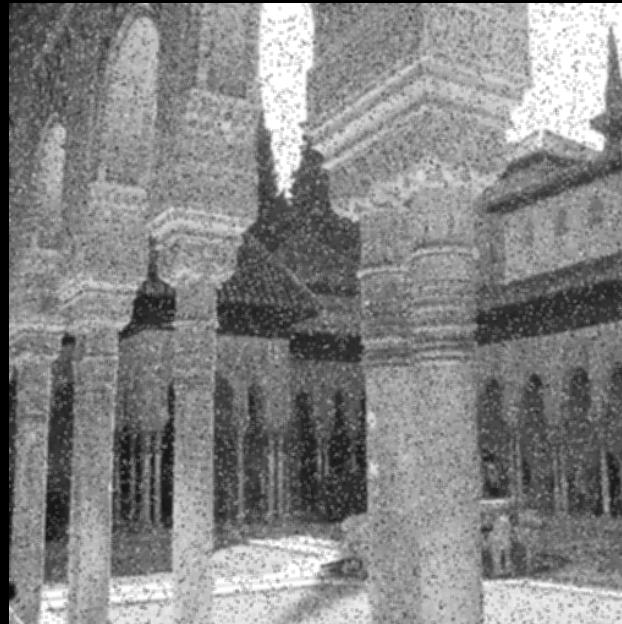
Original



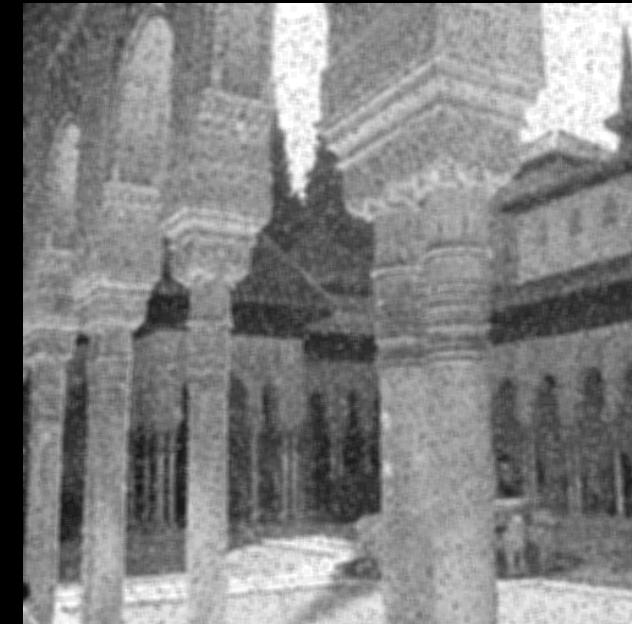
Mediana
 3×3



Promediado
 3×3

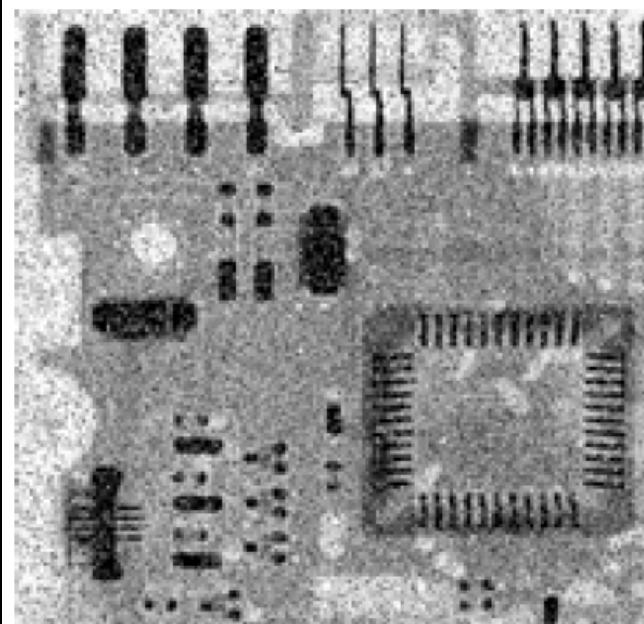


Promediado
 5×5

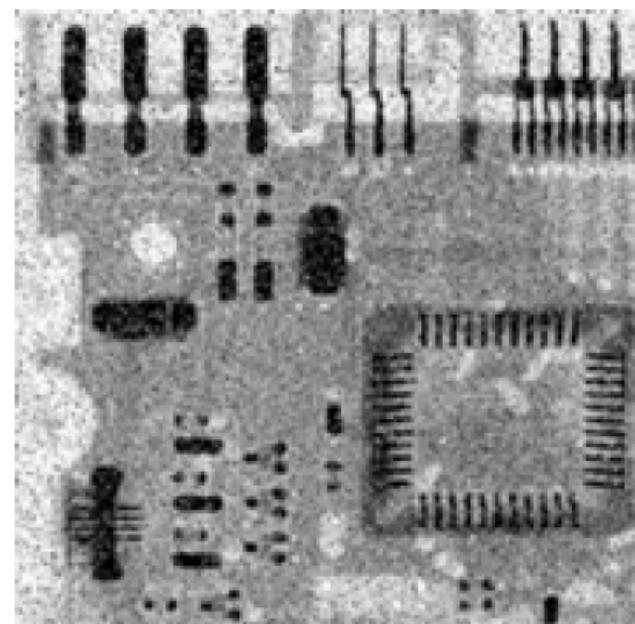


- Ejemplo: eliminación de ruido

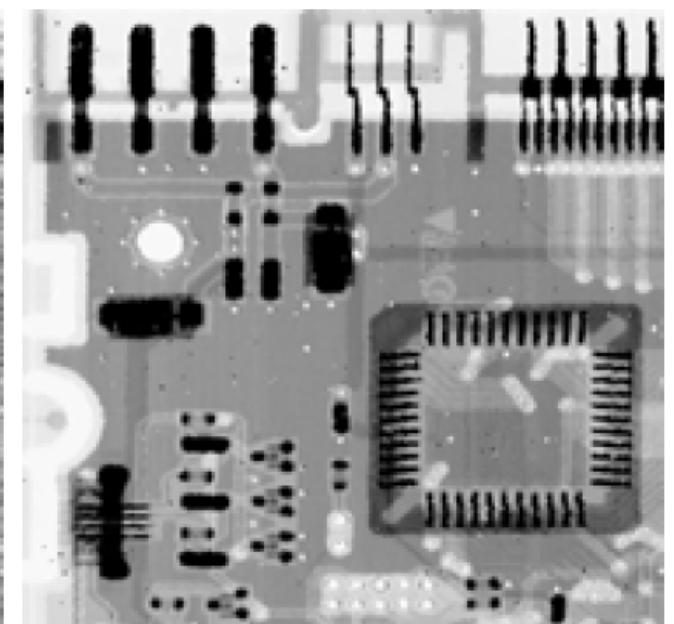
Imagen con ruido
(sal y pimienta)



3x3 promedio



3x3 mediana



Fuente: Gonzalez, Digital Imagen Processing, Prentice Hall, 2008

- Filtros utilizados para resaltar el detalle fino o realzar una imagen emborronada.
- Fundamento:
 - Del mismo modo que el promediado es parejo a la integración, la operación opuesta guarda relación con la derivación, que toma valores máximos en discontinuidades (y ruido); por lo tanto, es útil para resaltarlas. Uso habitual de la primera y segunda derivada.
 - En imágenes discretas el concepto de derivada se asocia al de diferencia, y la definición de diferencias sucesivas debe cumplir propiedades similares.

$$\frac{\partial f[n]}{\partial n} = f[n+1] - f[n] \quad \frac{\partial^2 f[n]}{\partial n^2} = f[n+1] + f[n-1] - 2f[n]$$

- Comparación de los efectos de la primera (1^{a}) y segunda (2^{a}) derivadas: la 1^{a} genera líneas más gruesas en los bordes de objetos, la 2^{a} responde más a puntos aislados y detalle fino, en escalones abruptos la 1^{a} responde más y la 2^{a} genera dos líneas, la 2^{a} responde más a una línea que a una rampa (borde) y más a un punto aislado (ruido) que a una línea.

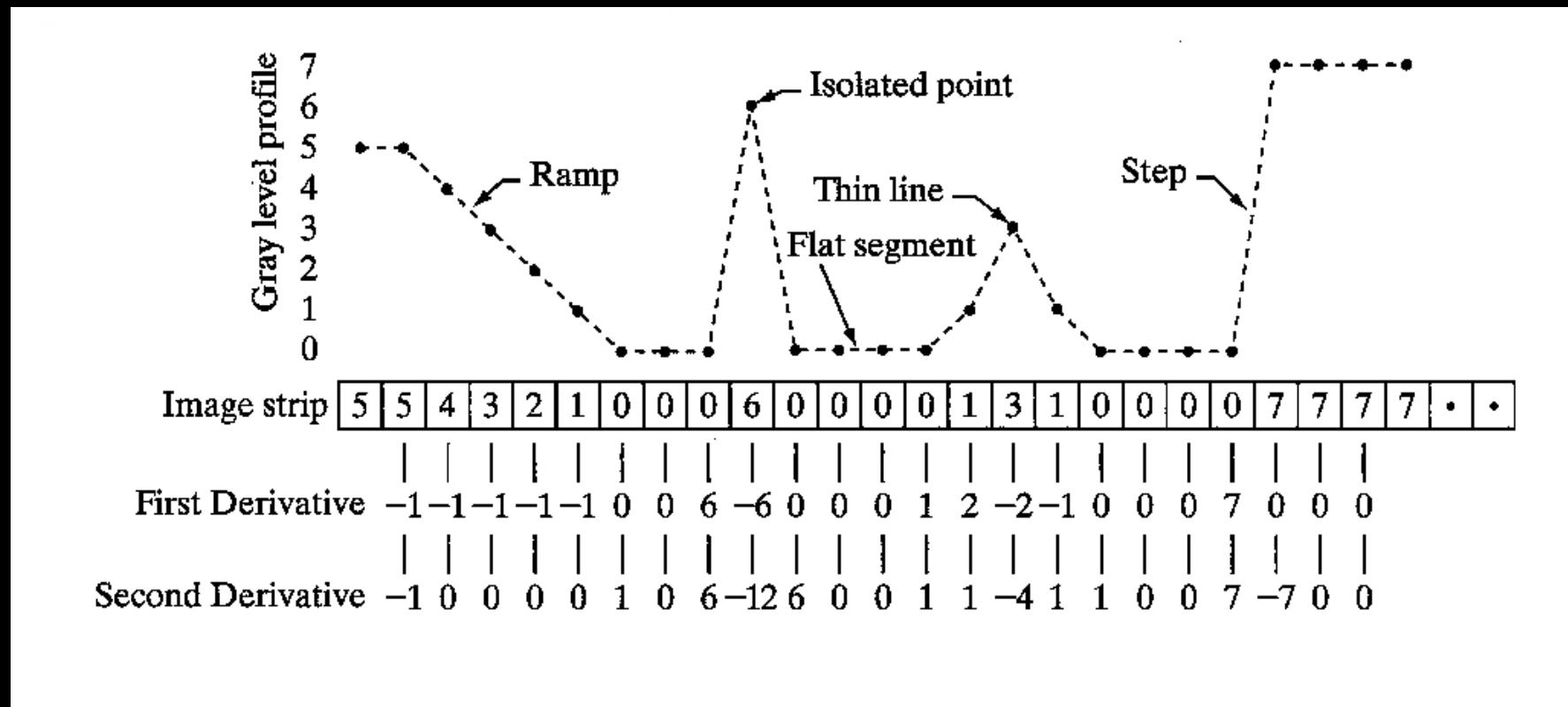


Imagen extraída de "Digital Image Processing", R.C. Gonzalez, Prentice Hall, 2002



$$\frac{\partial f}{\partial x}$$



$$\frac{\partial f}{\partial y}$$

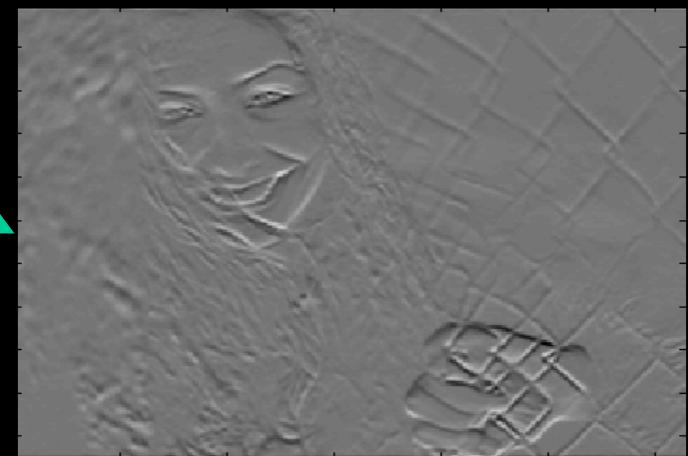


Imagen extraída de "Digital Image Processing", R.C. Gonzalez, Prentice Hall, 2002

- La Laplaciana

- Desarrollo de una máscara isótropa equivalente a una segunda derivada. El camino es hallar una versión discreta de la Laplaciana:

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

- Teniendo en cuenta que para el caso discreto y para cada dimensión:

$$\frac{\partial^2\psi[n, m]}{\partial n^2} = \psi[n + 1, m] + \psi[n - 1, m] - 2\psi[n, m]$$

$$\frac{\partial^2\psi[n, m]}{\partial m^2} = \psi[n, m + 1] + \psi[n, m - 1] - 2\psi[n, m]$$

- , resultaría una máscara de 3x3 isótropa a incrementos de 90°, que combinada con una similar rotada 45° da lugar a la máscara definitiva:

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

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

- La Laplaciana

- La Laplaciana elimina las zonas uniformes. Para realzar el detalle fino de una imagen (y sus contornos) se combina la imagen con su Laplaciana o se combinan usando una máscara conjunta; luego, para representarla, se re-escala el resultado:

$$\theta[n, m] = \psi[n, m] - \nabla^2\psi[n, m]$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

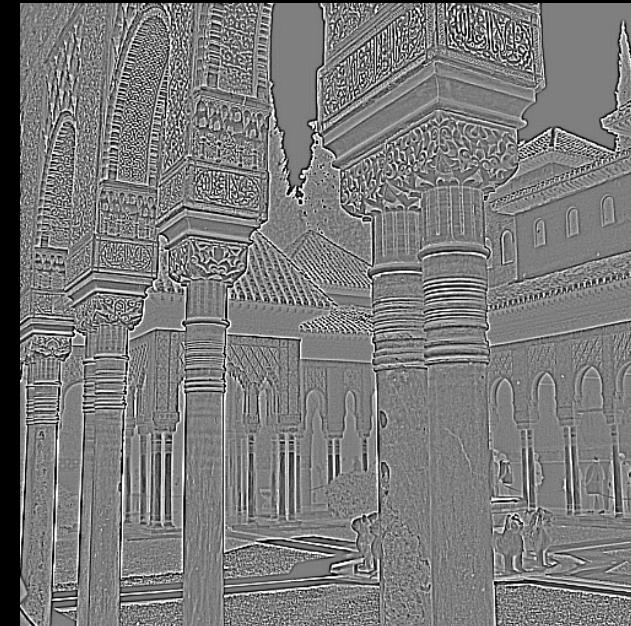
o la isotropa

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

REALCE DE CONTORNOS

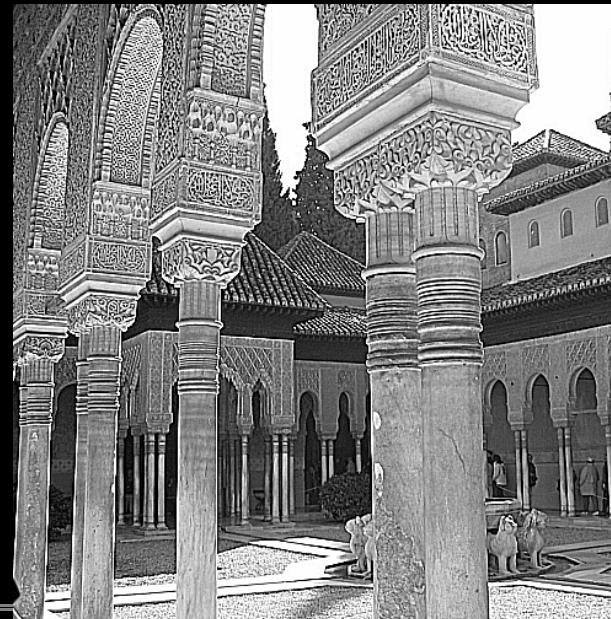
Realce del detalle fino

Original



Laplaciana
escalada

Realce
escalado



- El gradiente
 - La primera derivada se implementa con la magnitud del gradiente:

$$\nabla \Psi = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial \psi}{\partial x} \\ \frac{\partial \psi}{\partial y} \end{bmatrix} \quad \nabla \psi = [G_x^2 + G_y^2]^{1/2} = \left[\left(\frac{\partial \psi}{\partial x} \right)^2 + \left(\frac{\partial \psi}{\partial y} \right)^2 \right]^{1/2} \simeq |G_x| + |G_y|$$

- Versión discreta del gradiente. Para una vecindad de 3x3, en torno a un píxel de valor z_5 :

$$\begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & z_5 & z_6 \\ z_7 & z_8 & z_9 \end{bmatrix}$$

, podemos definir las derivadas direccionales:

$$G_n = (z_6 - z_5), G_m = (z_8 - z_5)$$

, o bien derivadas cruzadas (Roberts, 1965) :

$$G_{n,m} = (z_9 - z_5), G_{-n,m} = (z_8 - z_6)$$

- El gradiente
 - Por lo tanto:

$$\nabla\psi[n, m] = [(z_9 - z_5)^2 + (z_8 - z_6)^2]^{1/2} \simeq |(z_9 - z_5)| + |(z_8 - z_6)|$$

, que se puede implementar con las máscaras (operador de Roberts):

$$R_{n,m} = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$R_{n,m} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

- Aproximaciones con operadores 3x3

Operador de Prewitt

$$P_m = \frac{1}{6} \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad P_n = \frac{1}{6} \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

Operador de Sobel

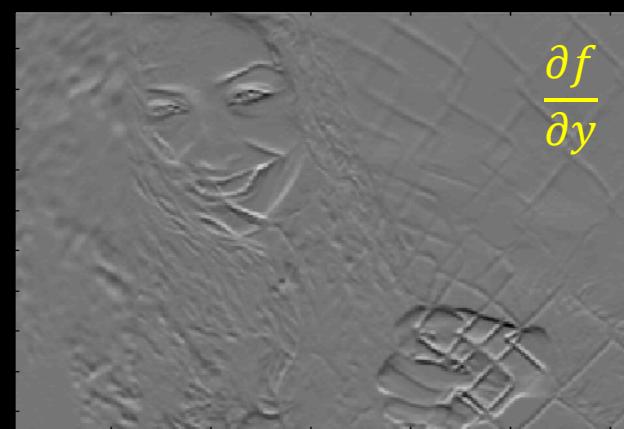
$$S_m = \frac{1}{8} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad S_n = \frac{1}{8} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Operador diagonal

$$P_{n,-m} = \frac{1}{6} \begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \quad P_{n,m} = \frac{1}{6} \begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}$$

Operador diagonal

$$S_{n,m} = \frac{1}{8} \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix} \quad S_{n,m} = \frac{1}{8} \begin{bmatrix} -2 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$



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



Detección de contornos

Original



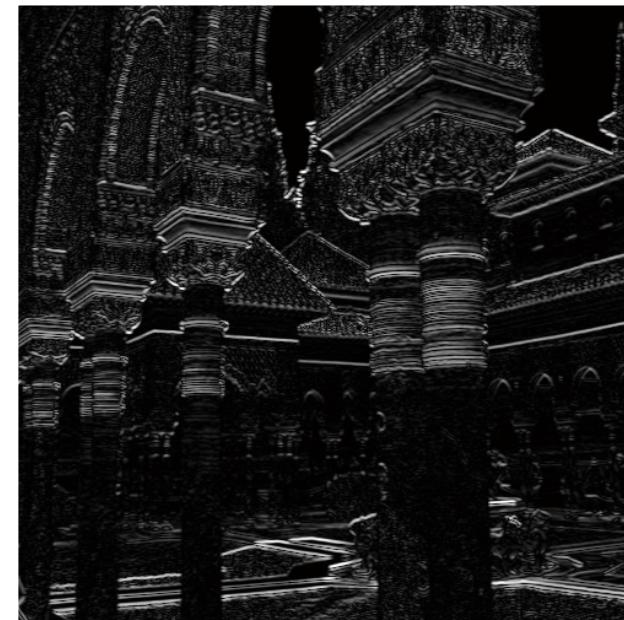
Prewitt



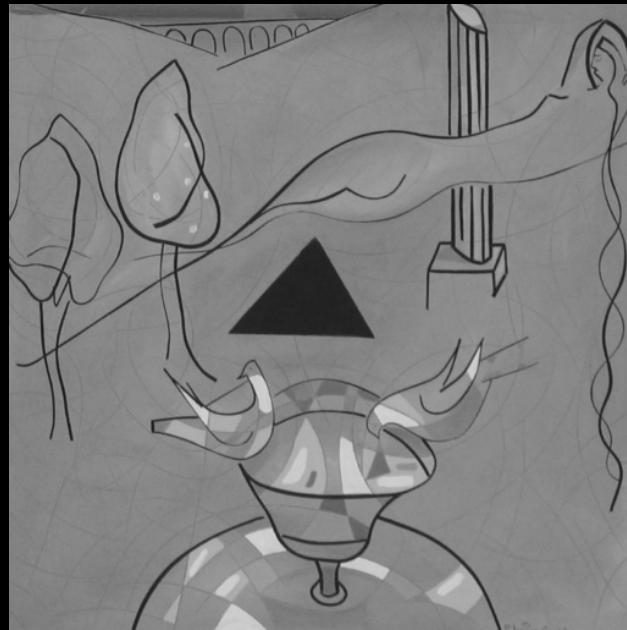
Prewitt - h



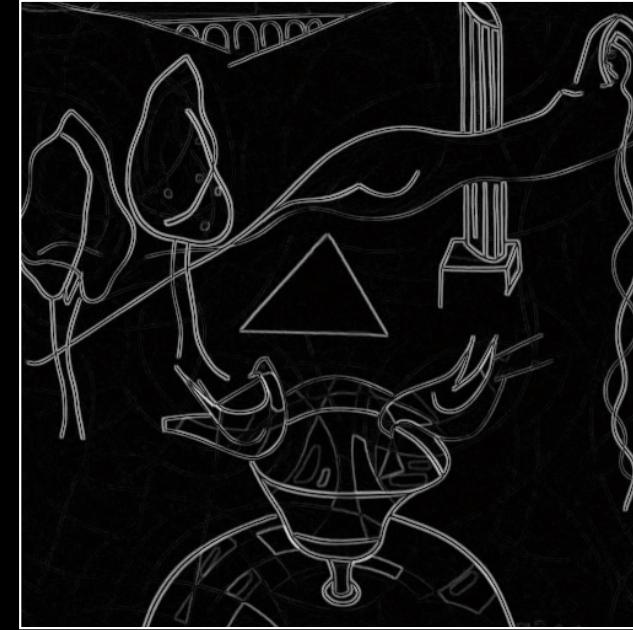
Prewitt - v



Original



Marcado de contornos



Gradiente
Prewitt

Marcado
de
contornos



- Diseño de filtros motivación frecuencial

- La operación de filtrado puede llevarse a cabo bien en el dominio “natural” o del píxel o bien en el dominio frecuencial, aplicando la propiedad de convolución:

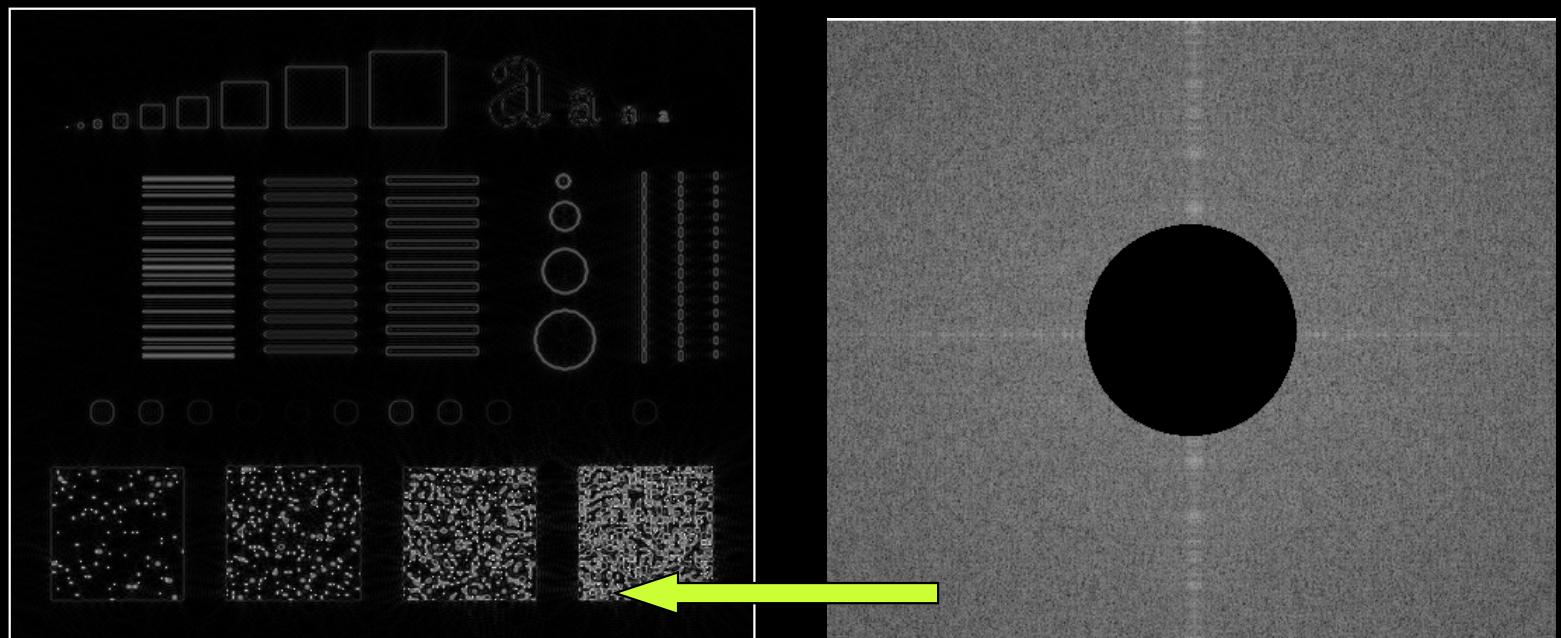
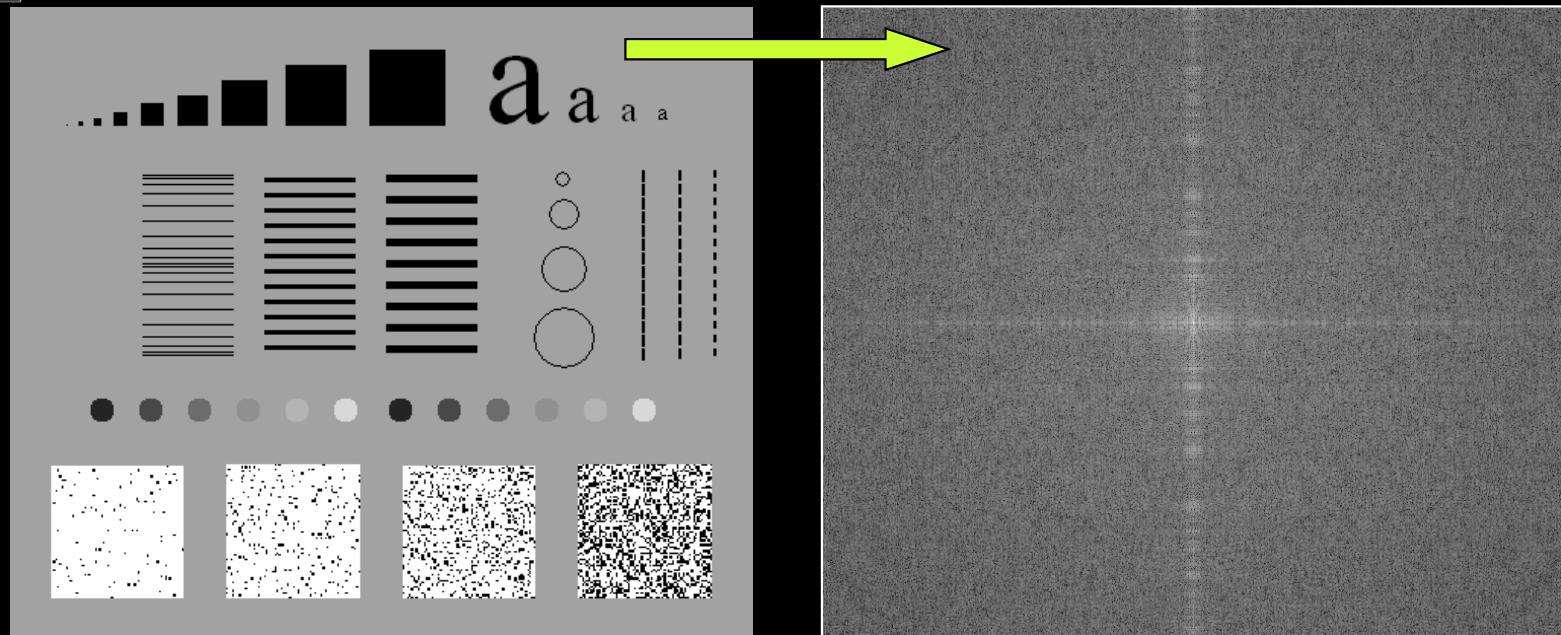
$$\theta[n, m] = \psi[n, m] * h[n, m] \xrightarrow{DSFT} O[u, v] = \Psi[u, v] \cdot H[u, v]$$

- Las máscaras son versiones truncadas de las respuestas al impulso correspondientes a los filtros frecuenciales.
 - Ejemplo: filtro paso alto (high-pass vs low-pass filter)

$$H_{hp}[u, v] = 1 - H_{lp}[u, v]$$

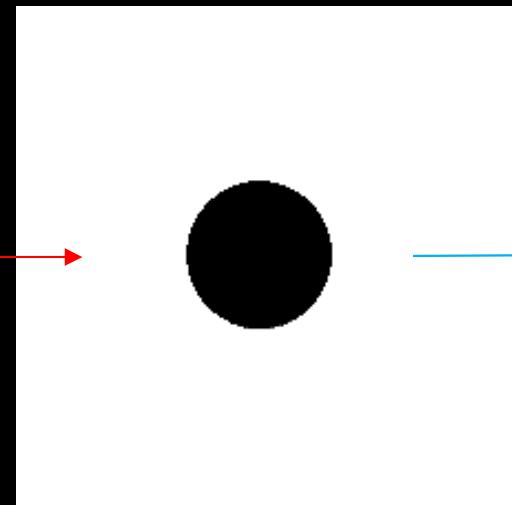
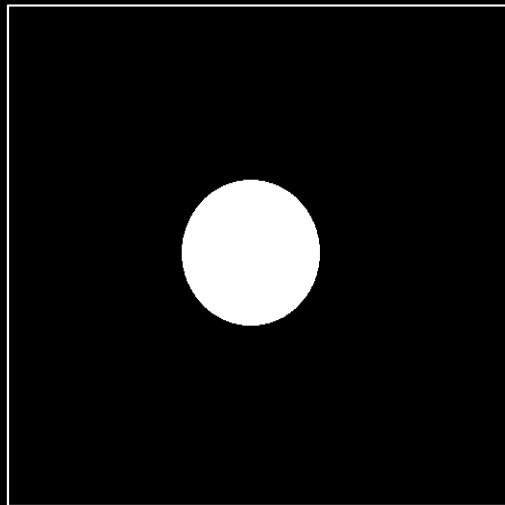
Dominio natural

Dominio frecuencial



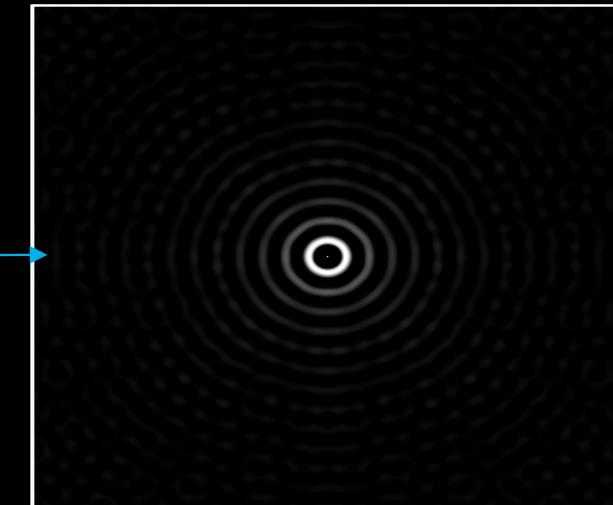
$$H_{hp}[u, v] = 1 - H_{lp}[u, v]$$

Filtro paso bajo



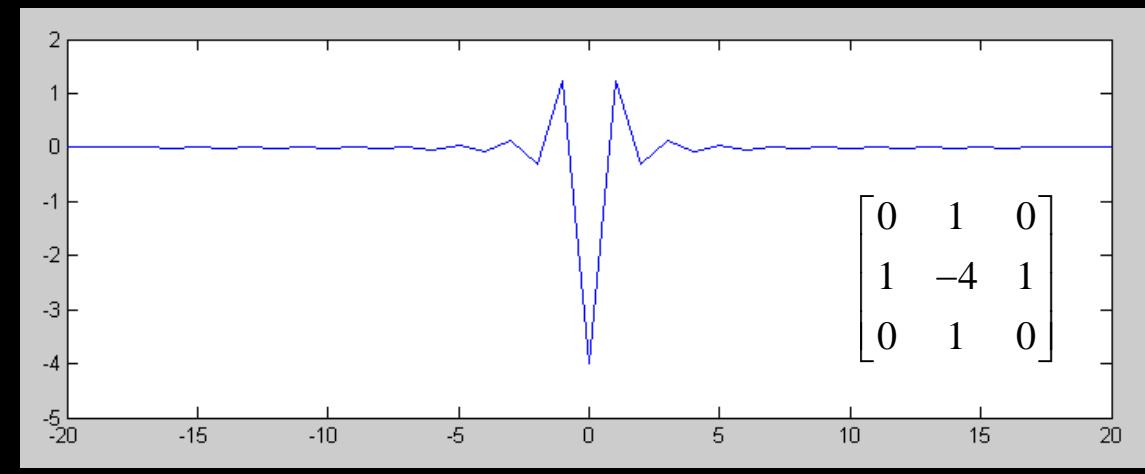
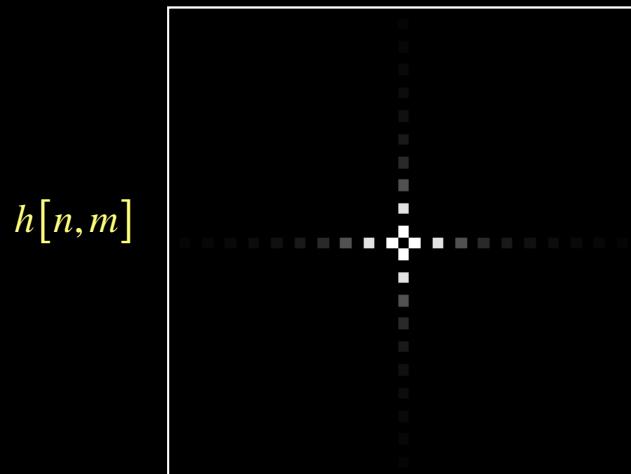
Filtro paso alto

$h[n, m]$



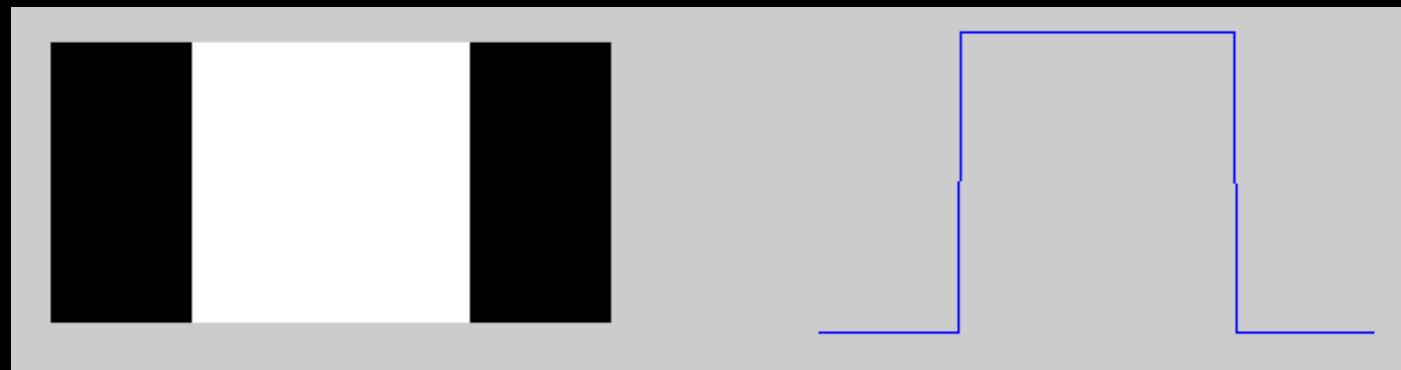
Detalle de la respuesta al impulso de la Laplaciana

Detalle 1D de la respuesta al impulso de la Laplaciana



- Base de técnicas de más alto nivel.
- Bordes (*edges*) y contornos (*boundaries*):
 - Borde: variación local significativa, es decir superior a un cierto umbral, del nivel de luminancia, en una dirección dada. Píxeles de borde, segmentos de borde.
 - Contorno: conjunto de píxeles conectados que limitan dos regiones contiguas.
- Modelado de bordes:
 - Bordes ideales.
 - Bordes no ideales y modelo en rampa.

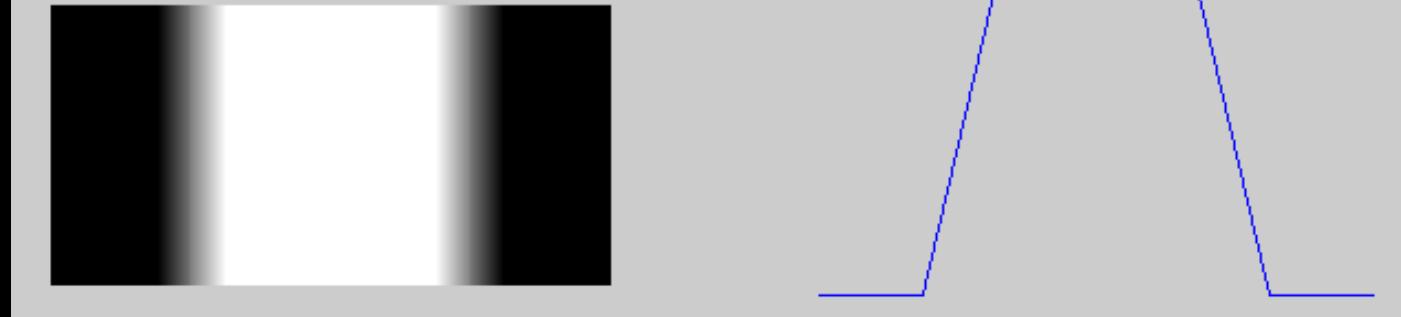
Ideal



Suavizado con
filtro binomial
de orden 5

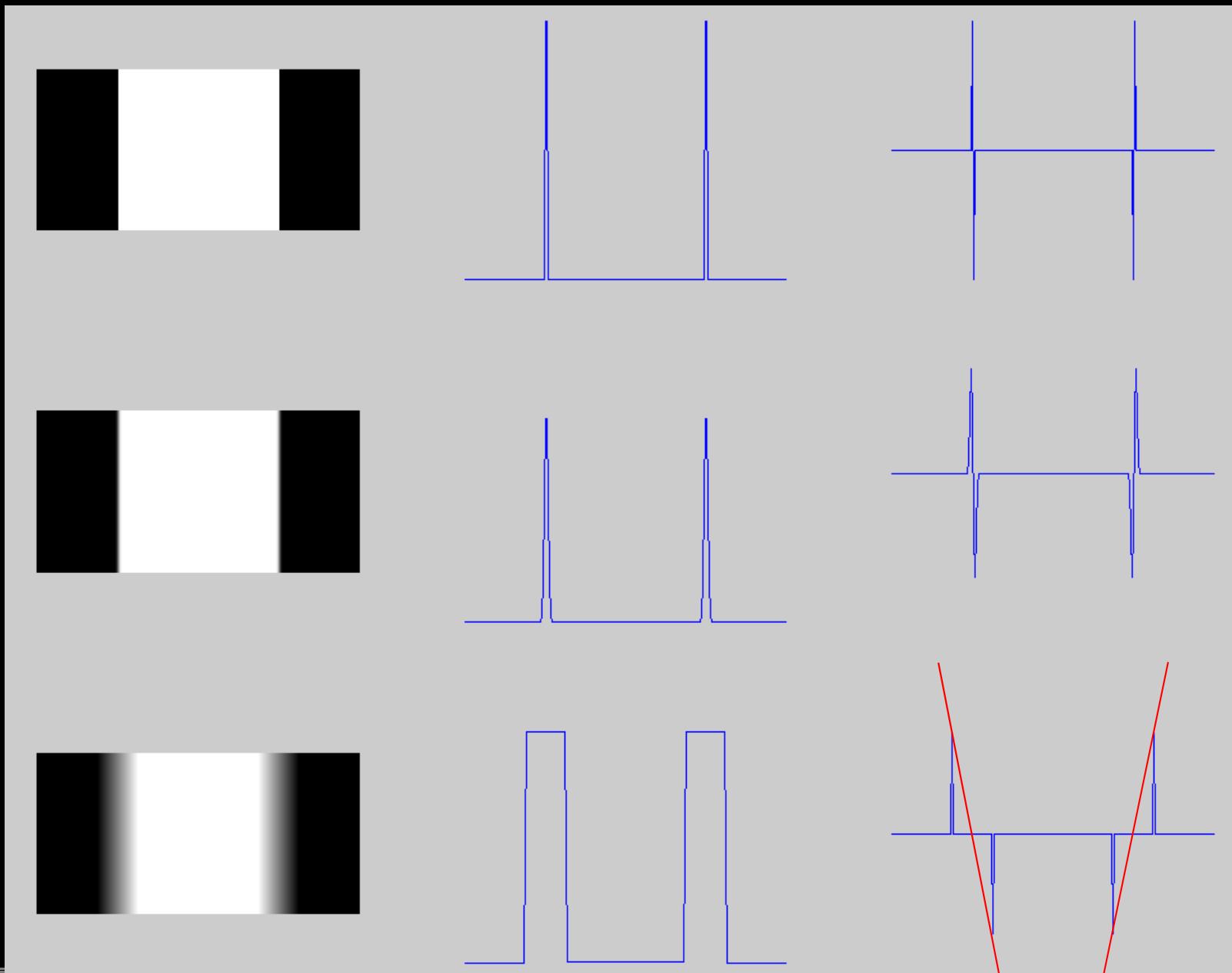


Modelo en
rampa



- Efecto de la primera y segunda derivada sobre un borde:
 - La primera derivada permite localizar variaciones de la luminancia
 - La segunda derivada indica los puntos en que comienza y termina la variación, y el punto 'medio' de ésta ('zero-crossing').
 - Posibilidad de utilizar estas herramientas como detectores.

Primera derivada

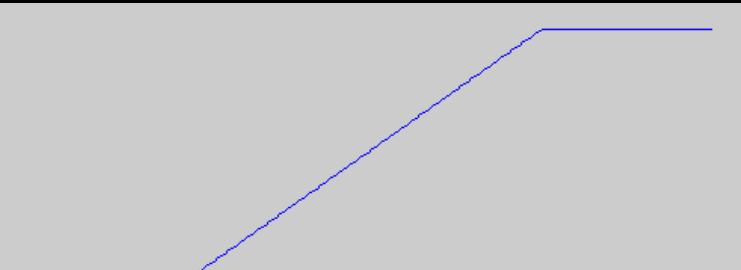


- Efecto del ruido sobre estos operadores:
 - Aunque el ruido sea apenas perceptible en una imagen, su efecto sobre la primera y segunda derivada es notable.

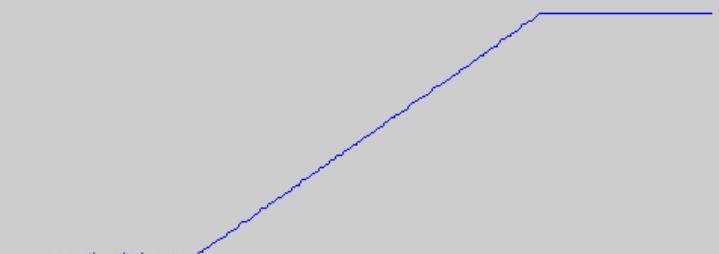
Borde con ruido gaussiano de distintas varianzas

Perfil

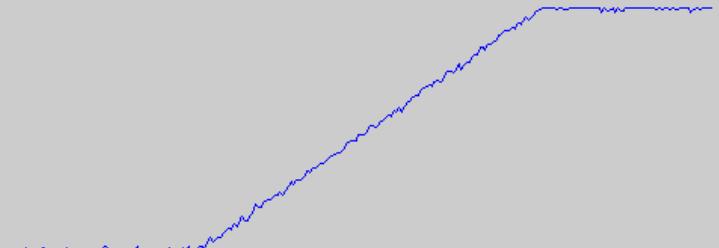
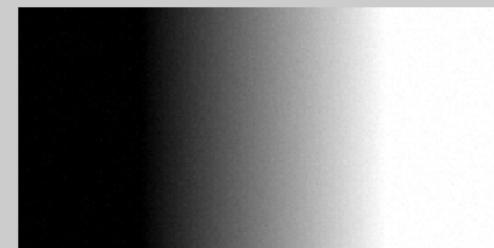
$$\sigma^2 = 0$$



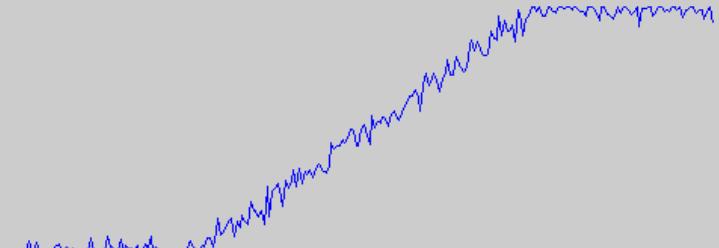
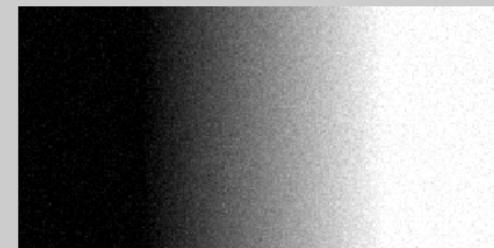
$$\sigma^2 = 10^{-5}$$



$$\sigma^2 = 10^{-4}$$



$$\sigma^2 = 10^{-3}$$



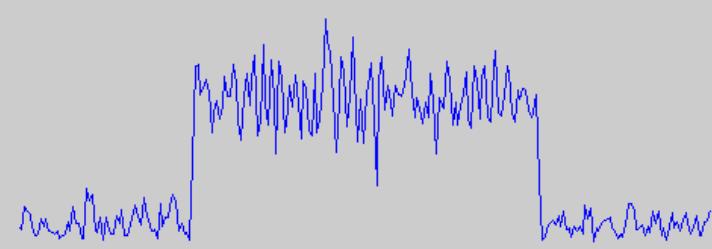
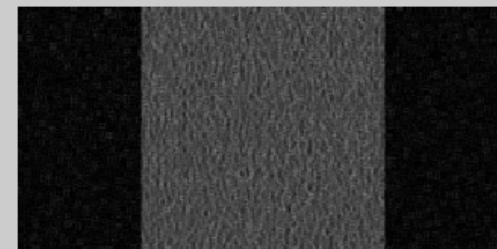
Primera derivada (Prewitt)

Perfil

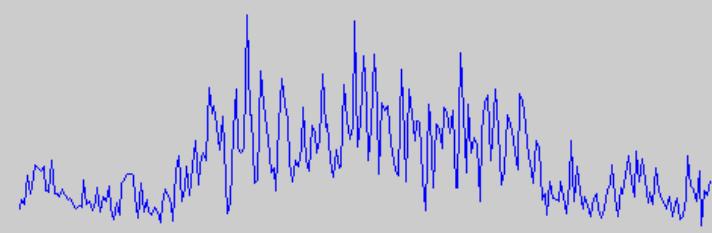
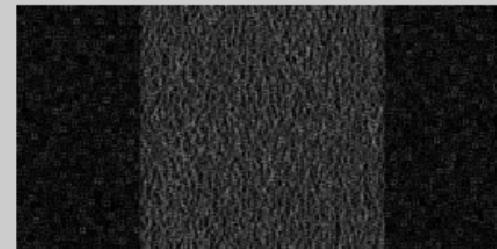
$$\sigma^2 = 0$$



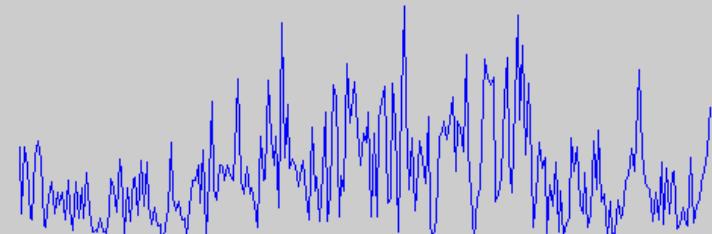
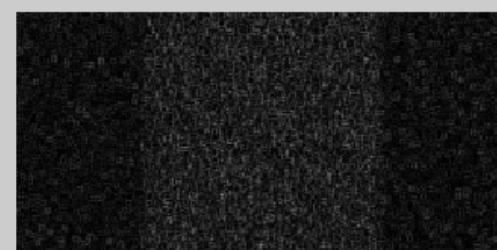
$$\sigma^2 = 10^{-5}$$



$$\sigma^2 = 10^{-4}$$

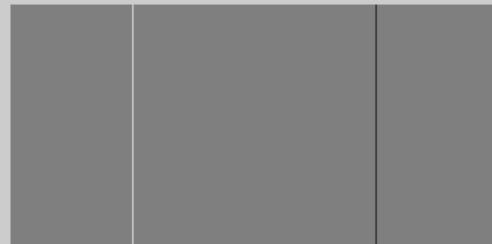


$$\sigma^2 = 10^{-3}$$

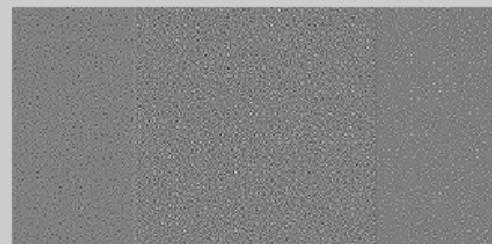


Segunda derivada (Laplaciana)

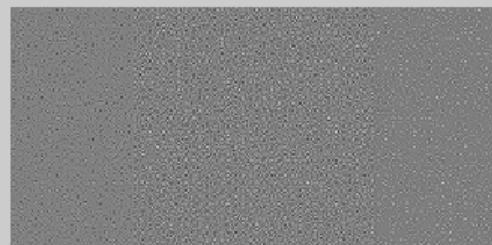
$$\sigma^2 = 0$$



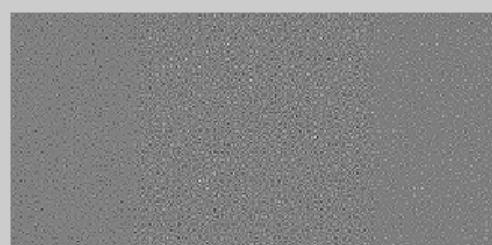
$$\sigma^2 = 10^{-5}$$



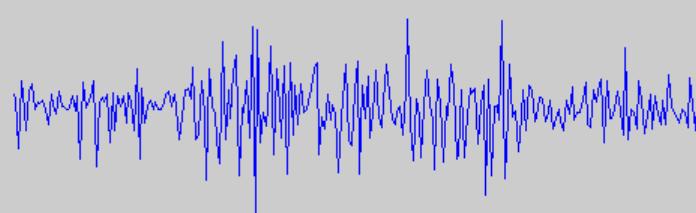
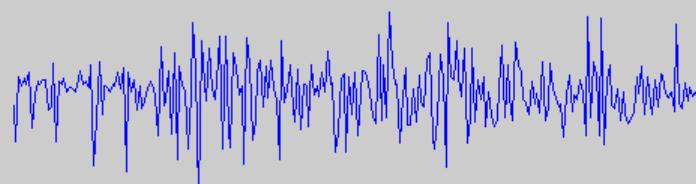
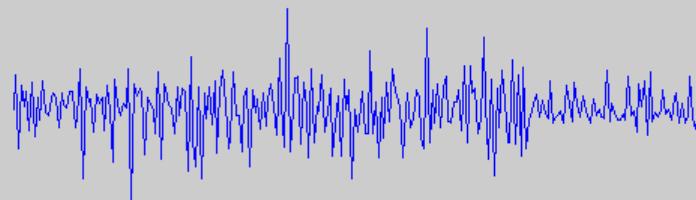
$$\sigma^2 = 10^{-4}$$



$$\sigma^2 = 10^{-3}$$



Perfil



- Detección de bordes

- Basada en el gradiente: suavizado ligero del operador de Sobel, suavizado previo para controlar el nivel de detalle, bordes gruesos, posibilidad de conocer la dirección del borde:

$$\alpha_{\nabla} = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad , \text{ con } G_x = \frac{\partial \psi}{\partial x}, G_y = \frac{\partial \psi}{\partial y}$$

- Basada en la Laplaciana: localización precisa del borde a partir de los cruces por cero, imprescindible un suavizado previo que resulta en el operador LoG (*Laplacian of Gaussian*).

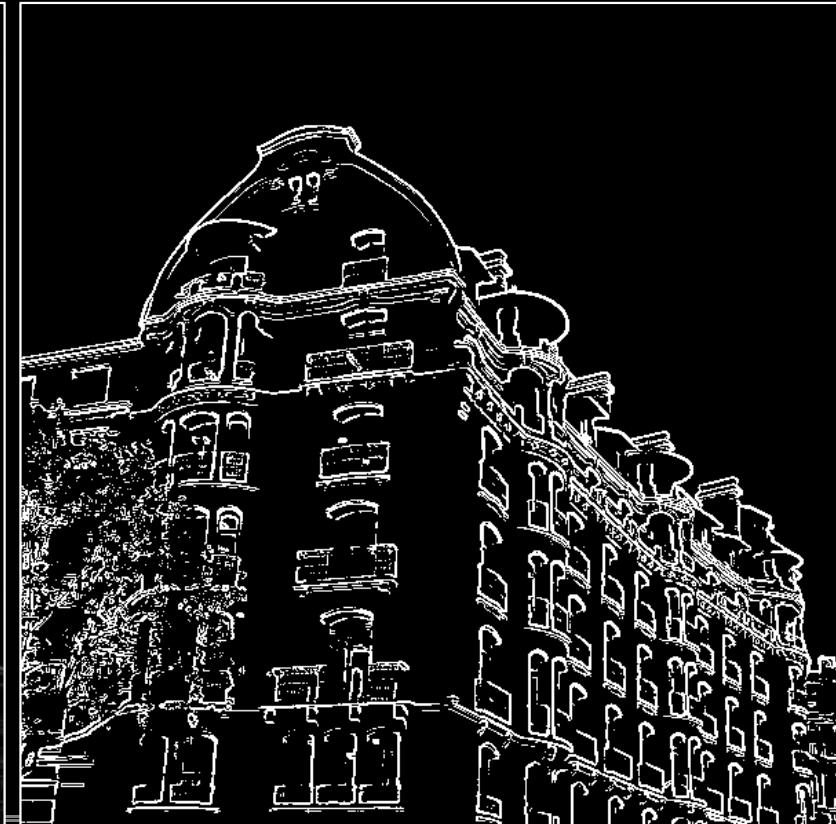
– Basado en gradiente



Gradiente Sobel



Umbralizada



– Basado en LoG



LoG

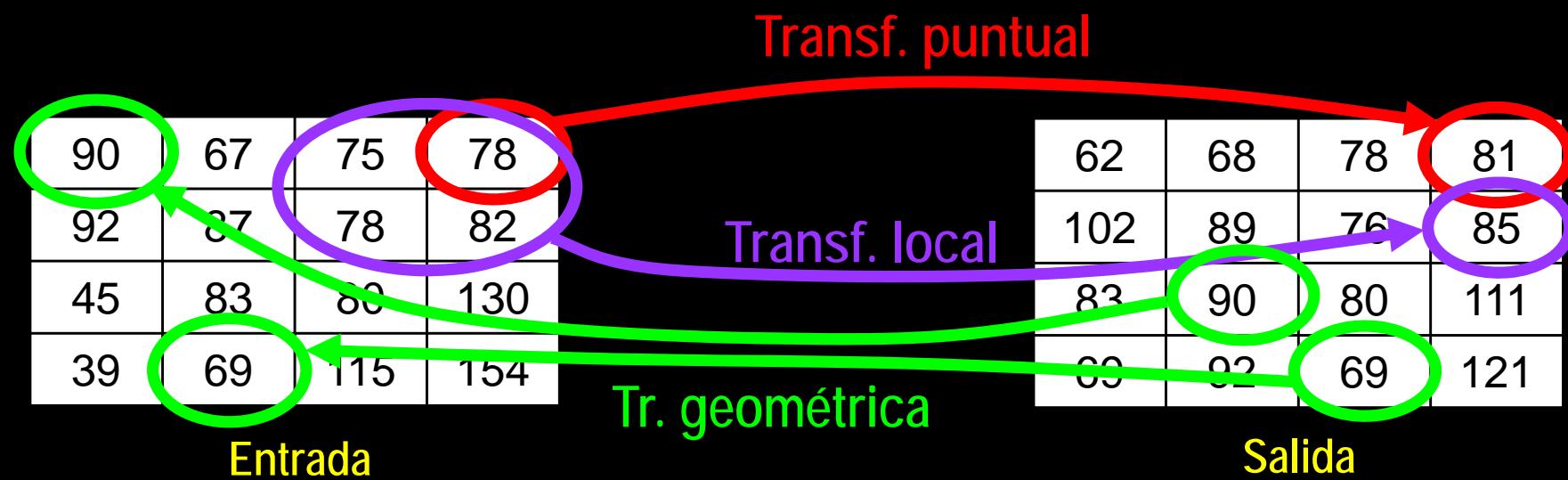


Umbralización y cruces
por cero



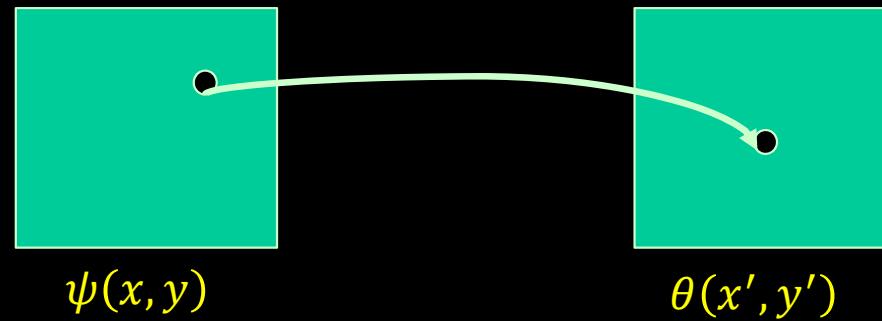
1. Introducción
2. Aspectos operativos
3. Filtrado espacial
4. Ajustes geométricos
 - Introducción
 - Tipos de transformaciones
 - Ajustes 2D
 - Ajustes 3D
5. Restauración lineal

- Transformadas geométricas vs locales vs puntuales



- Transformaciones que modifican la posición de los píxeles de la imagen pero, en el caso de imágenes continuas, no sus valores

$$\psi(x, y) \xrightarrow{T} \theta(x', y') / (x', y') = T(x, y)$$



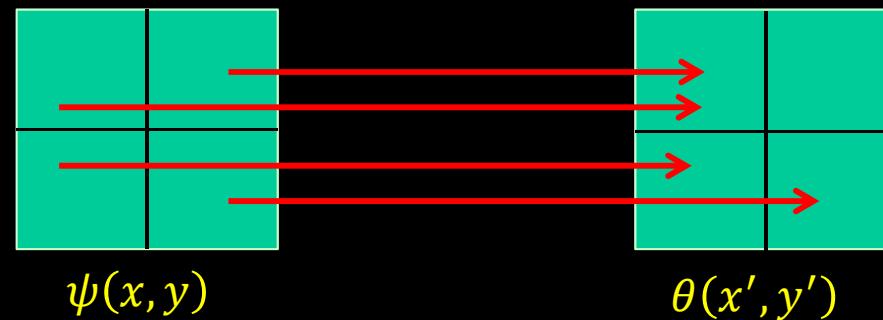
– Ejemplo:

$$\begin{aligned}x' &= f_x(\psi[x, y]) = x + 3 \\y' &= f_y(\psi[x, y]) = y - 1\end{aligned}$$

$$\theta(x + 3, y - 1) = \psi(x, y)$$

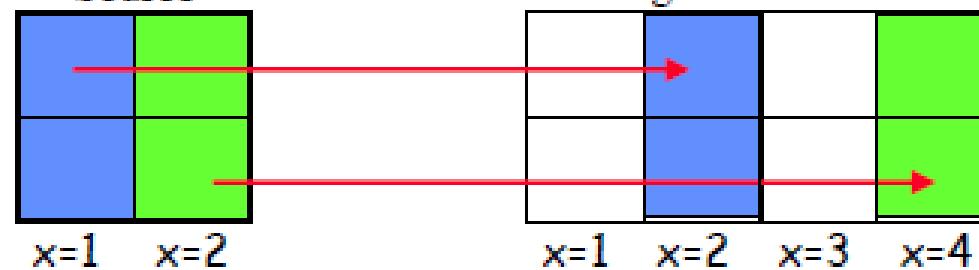
- Direccionalidad de la transformación
 - Transformación hacia adelante (*forward mapping*)

$$\psi(x, y) \xrightarrow{T} \theta(x', y') / (x', y') = T(x, y)$$

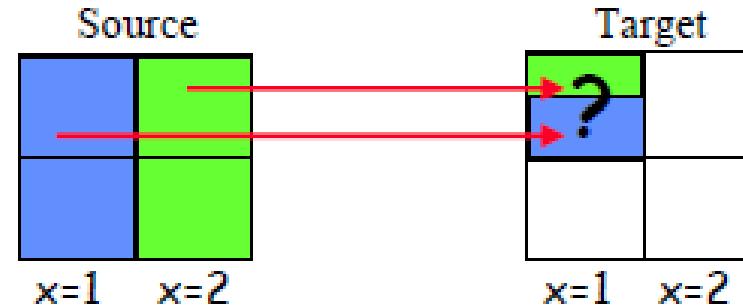


- Problemas debido a:
 - Agujeros (*holes*): píxeles vacíos en $\theta(x', y')$
 - Solapamiento (*overlap*): varios pixeles de $\psi(x, y)$ asignados al mismo píxel
 - Posiciones (x', y') no enteras
- Interpolación para resolver problemas

$$x' = f_x(x, y) = 2x \quad y' = f_y(x, y) = y$$



$$x' = f_x(x, y) = 0.7x \quad y' = f_y(x, y) = y$$

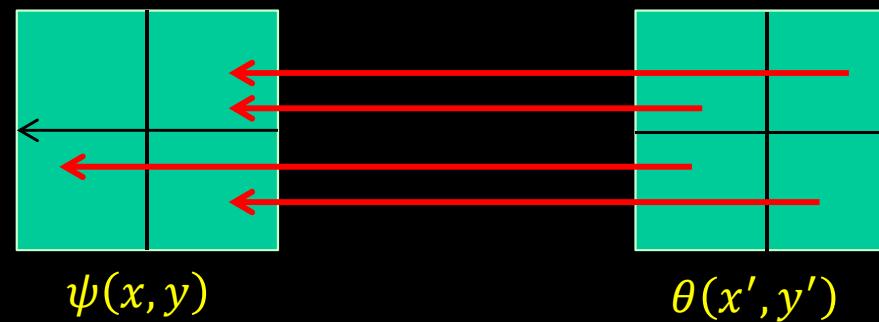


Fuente: Image Processing (<http://cs.haifa.ac.il/hagit/courses/ip/>)

- Transformación hacia atrás (*reverse mapping*)

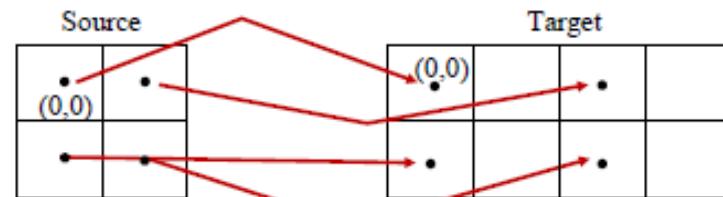
- No hay agujeros/solapamiento en $\theta(x', y')$
- Requiere interpolación

$$\psi(x, y) \leftarrow \theta(x', y') / (x', y') = T(x, y)$$

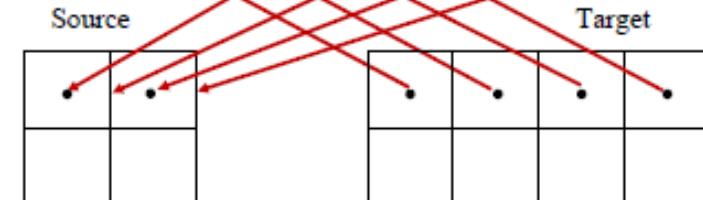


Escalado en el eje X

– Forward mapping: $x' = 2x ; y' = y$

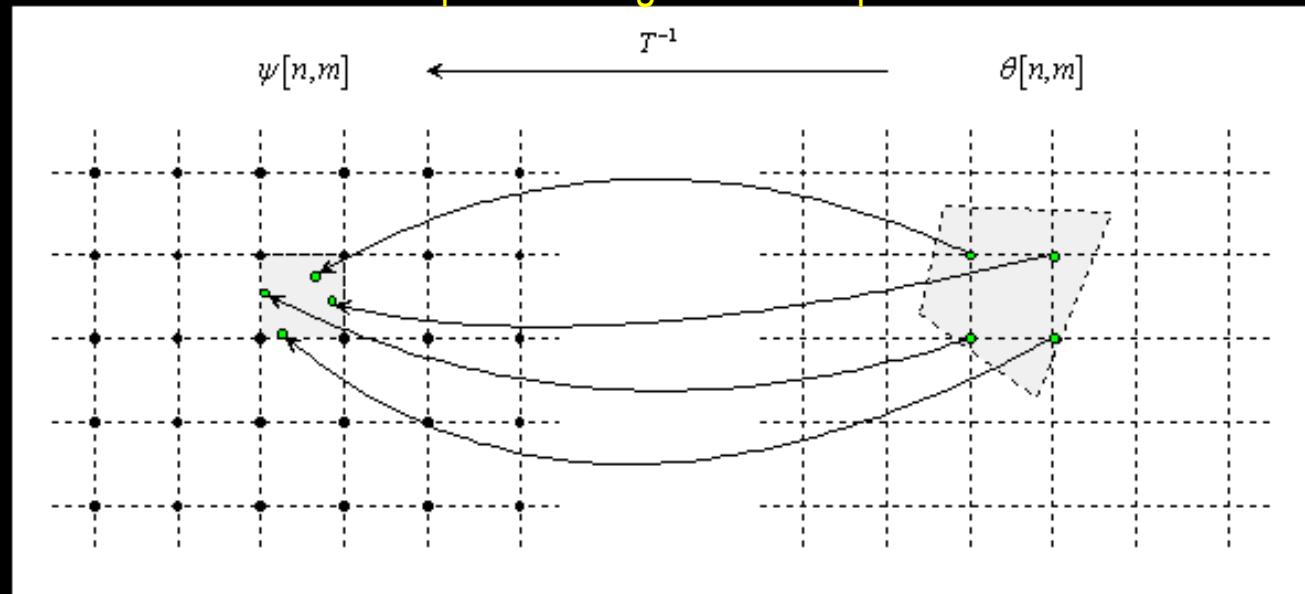


– Inverse mapping: $x = x'/2 ; y = y'$



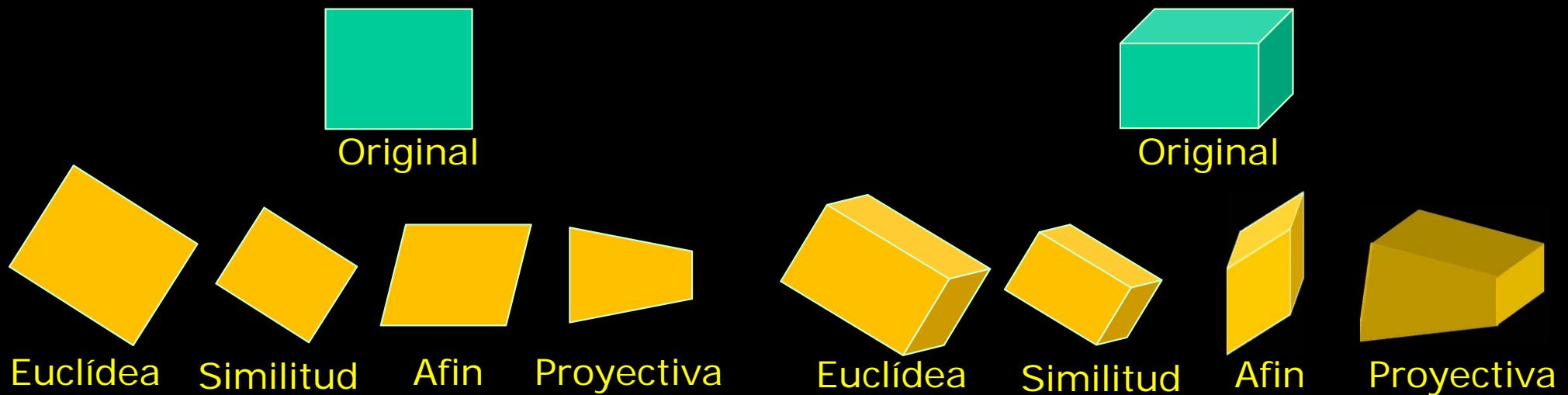
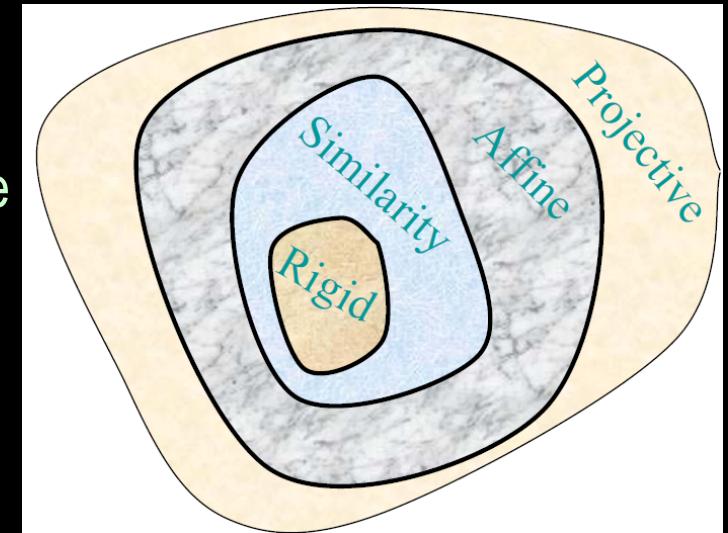
- La posición transformada en general no resulta en un par de coordenadas enteras, lo que en el caso de imágenes discretas exige interpolar.

Obtención de la posición origen de cada punto transformado



■ Tipos

- Isométrica (Euclídea/Rígida): traslación + rotación
- Similitud: traslación + rotación + escalado uniforme
- Afín: traslación + rotación + escalado no uniforme
- Proyectiva: general



Demo: <http://www.cs.technion.ac.il/~cs234325/Applets/applets/transformation/GermanApplet.html>

- Transformaciones afines: desplazamientos, rotaciones y escalados
 - Enfoque alternativo:

Transformación afín

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} c_1 & c_2 & c_3 \\ c_4 & c_5 & c_6 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

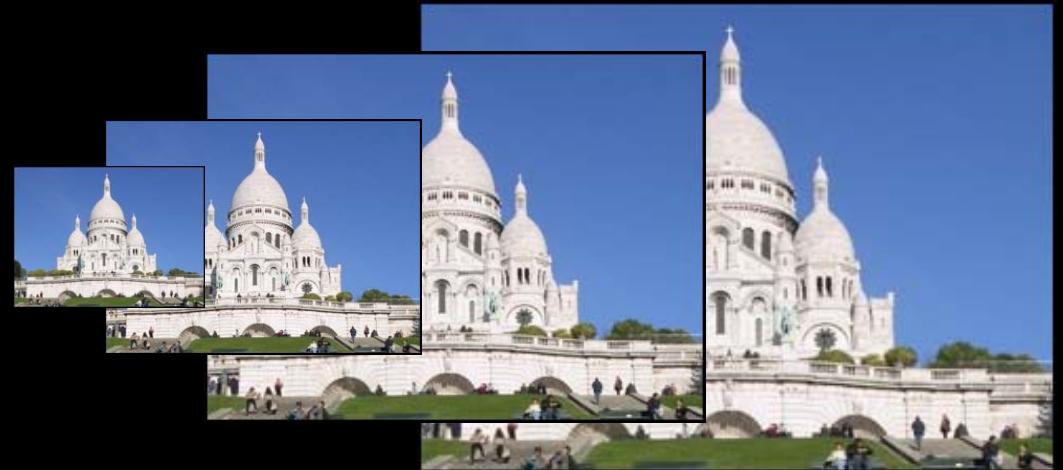
Rotación	Escalado	Desplazamiento	Inclinación
$\begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \end{bmatrix}$	$\begin{bmatrix} \rho_x & 0 & 0 \\ 0 & \rho_y & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \end{bmatrix}$	$\begin{bmatrix} 1 & -i_x & 0 \\ -i_y & 1 & 0 \end{bmatrix}$

- Transformación afines:

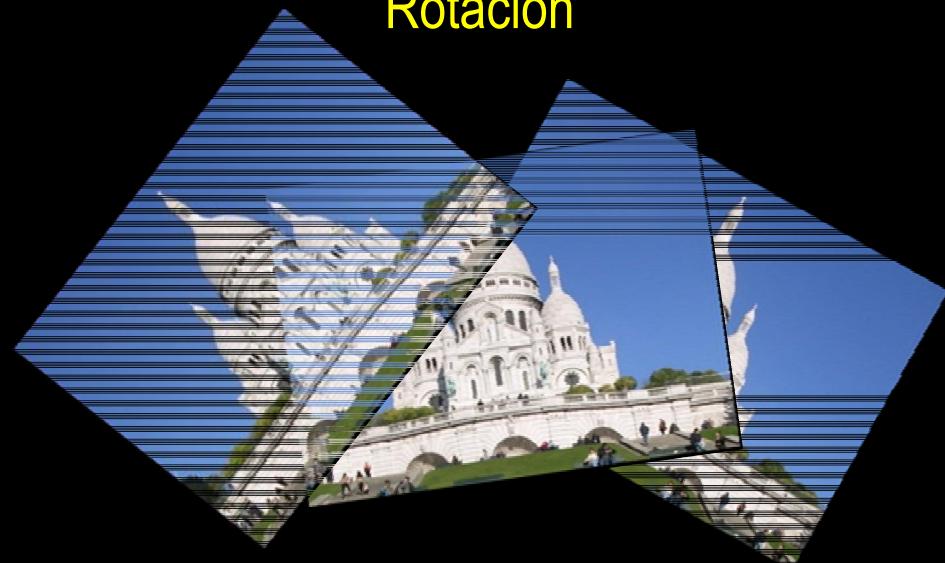
Desplazamiento (traslación)



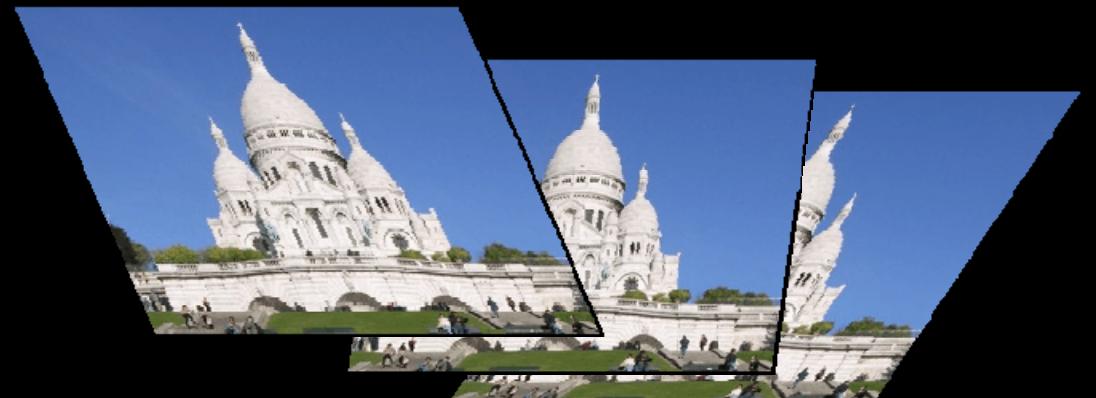
Escala



Rotación



Inclinación



Fuente: Ginés García Mateos, <http://dis.um.es/profesores/ginesgm>

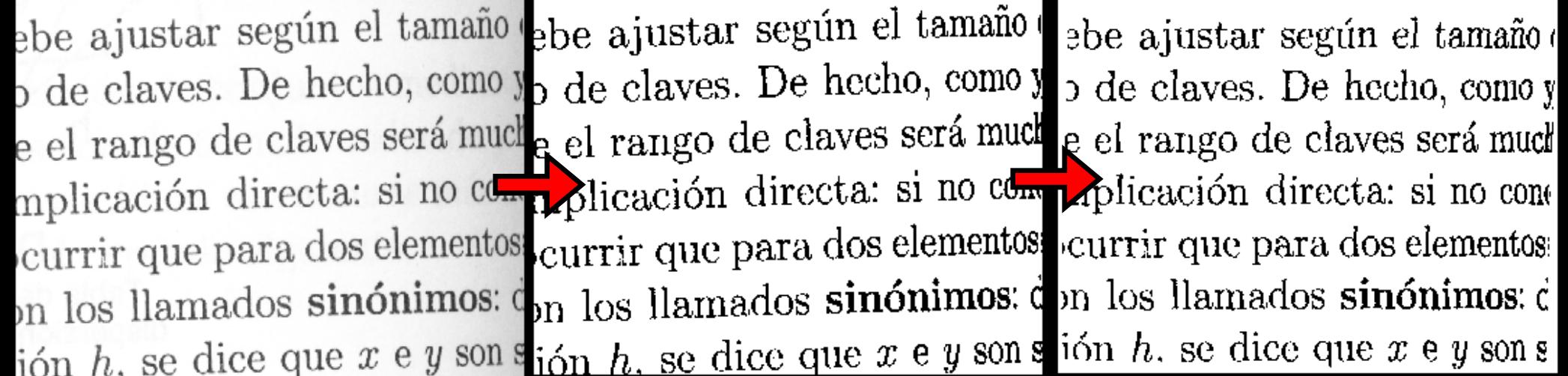
- Transformación afines: aplicación en reconocimiento de objetos (OCR)

Imagen de entrada

Umbralizar

Rectificar: detectar orientación y rotar

Se debe ajustar según el tamaño de las imágenes de claves. De hecho, como y se el rango de claves será mucho más amplio que para dos elementos con los llamados sinónimos: cuando se dice que x e y son sinónimos en la función h , se dice que x e y son sinónimos en la función h .



- Distorsión radial

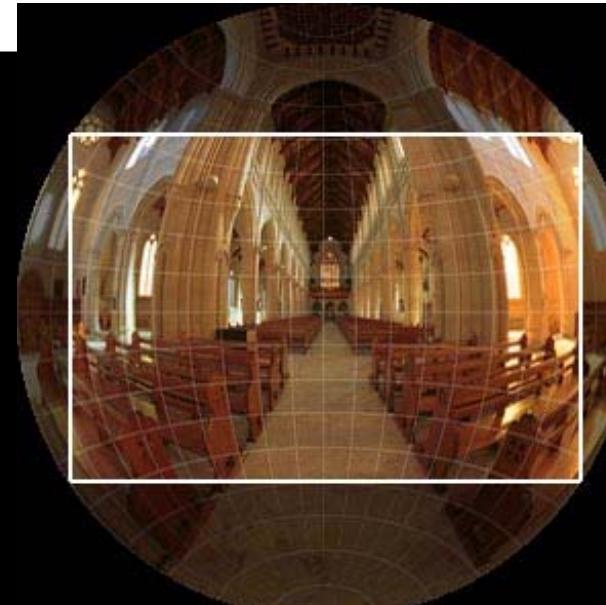
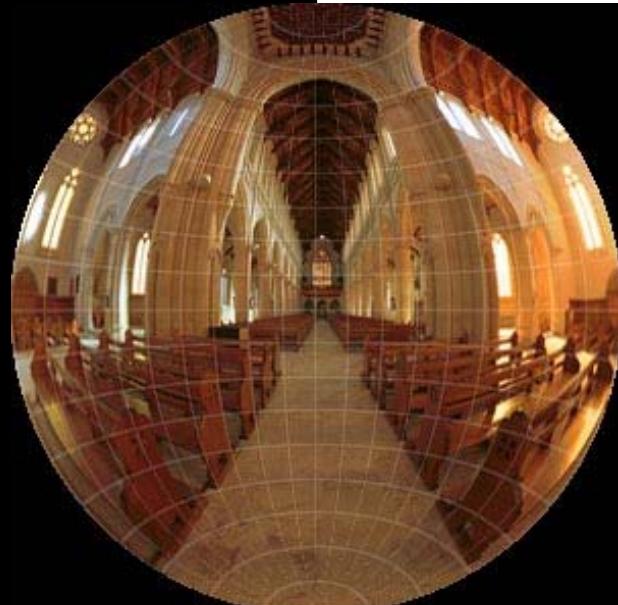
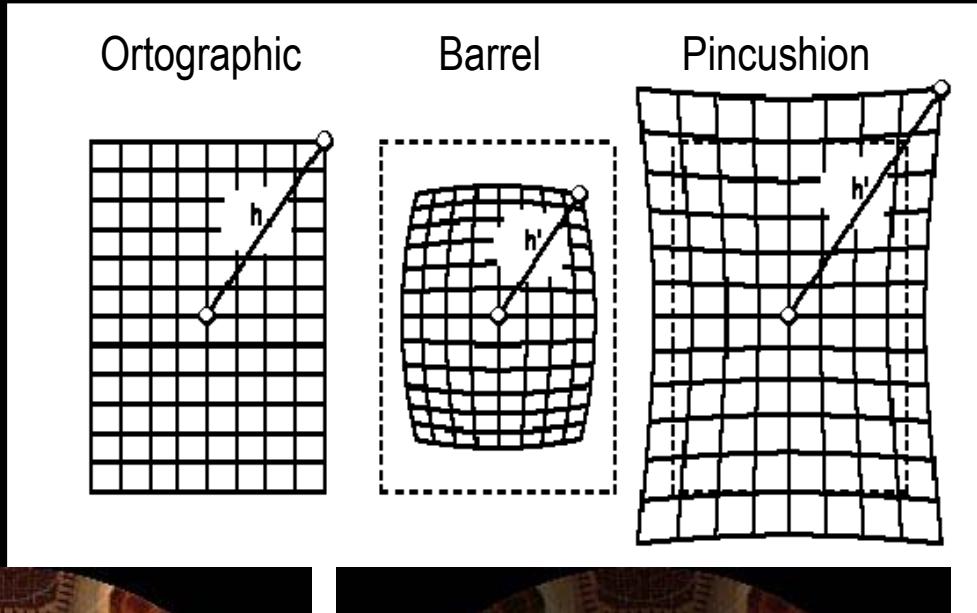
- La distorsión radial es una deformación introducida por las lentes de las cámaras, que da lugar a un *curvado* de las zonas periféricas de las imágenes.

Observar la curvatura de la puerta



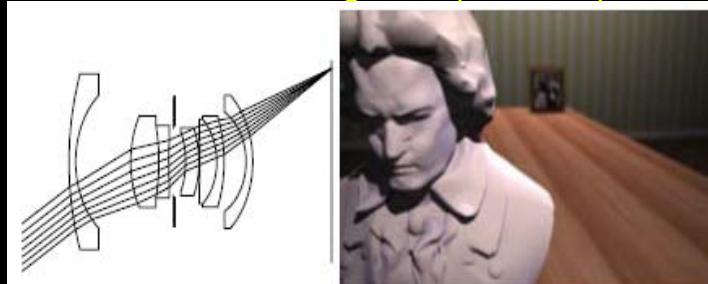
- La distorsión radial se modela como un desplazamiento radial, según la distancia, r , al centro de la imagen.

- Distorsión radial



- Distorsión radial

Gran angular (35 mm)



Doble gausiana (50 mm)

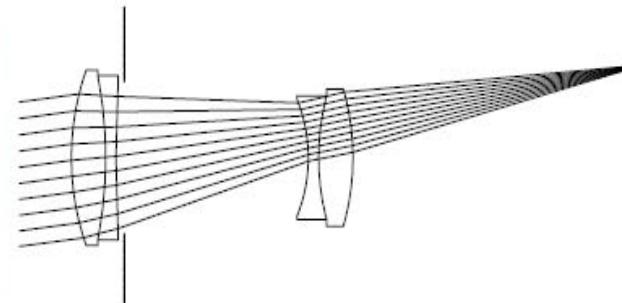
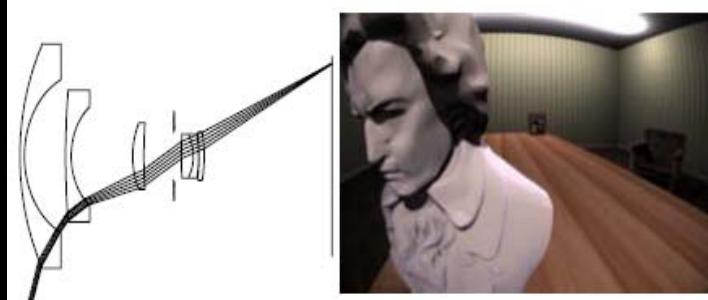
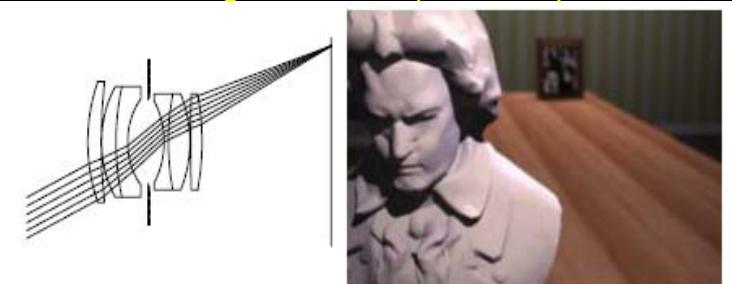


Figure 8: Four views of the same scene taken with a 16mm fisheye lens (bottom left), 35mm wide-angle lens (top left), 50mm double-Gauss lens (top right), and a 200mm telephoto lens (bottom right). A profile view of the lens system used to take each image is shown on the left. As with physical lenses, perspective is compressed with long focal lengths and expanded with short focal lengths. The fisheye image shows the lens' signature barrel distortion.

Ojo de pez (16 mm)

Teleobjetivo (200 mm)

Fuente: Paul E. Haeberli, Kurt Akeley, "The accumulation Buffer: Hardware support for high-quality rendering", Computer Graphics, SIGGRAPH '90

- Corrección radial
 - Esquemas basados en funciones de proyección radial

Perspectiva	Estereográfica	Equidistancia	Angulo sólido	Ley del seno
$r(\alpha) = f \cdot \tan(\alpha)$	$r(\alpha) = f \cdot \tan\left(\frac{\alpha}{2}\right)$	$r(\alpha) = f \cdot \alpha$	$r(\alpha) = f \cdot \sin\left(\frac{\alpha}{2}\right)$	$r(\alpha) = \sin\left(\frac{\alpha}{2}\right)$



- Corrección radial
 - Esquemas basados en aproximaciones polinómicas

Patrón de ajuste capturado



Patrón de ajuste corregido



Modelo polinómico de deformación:

$$d(\mathbf{x}, \mathbf{x}') = k_1 r_{\mathbf{x}}^2 + k_2 r_{\mathbf{x}}^4 + \dots + k_n r_{\mathbf{x}}^{2n}, \quad r_{\mathbf{x}} = d(\mathbf{x}, \mathbf{0}), \mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \mathbf{x}' = \begin{bmatrix} x' \\ y' \end{bmatrix}, \mathbf{0} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

Obtención de los parámetros del modelo mediante el ajuste a rectas de puntos capturados que se sabe 'a priori' que están alineados.

- Corrección radial
 - Esquemas basados en aproximaciones polinómicas

Imagen captada con un 'tele'



Corrección polinómica



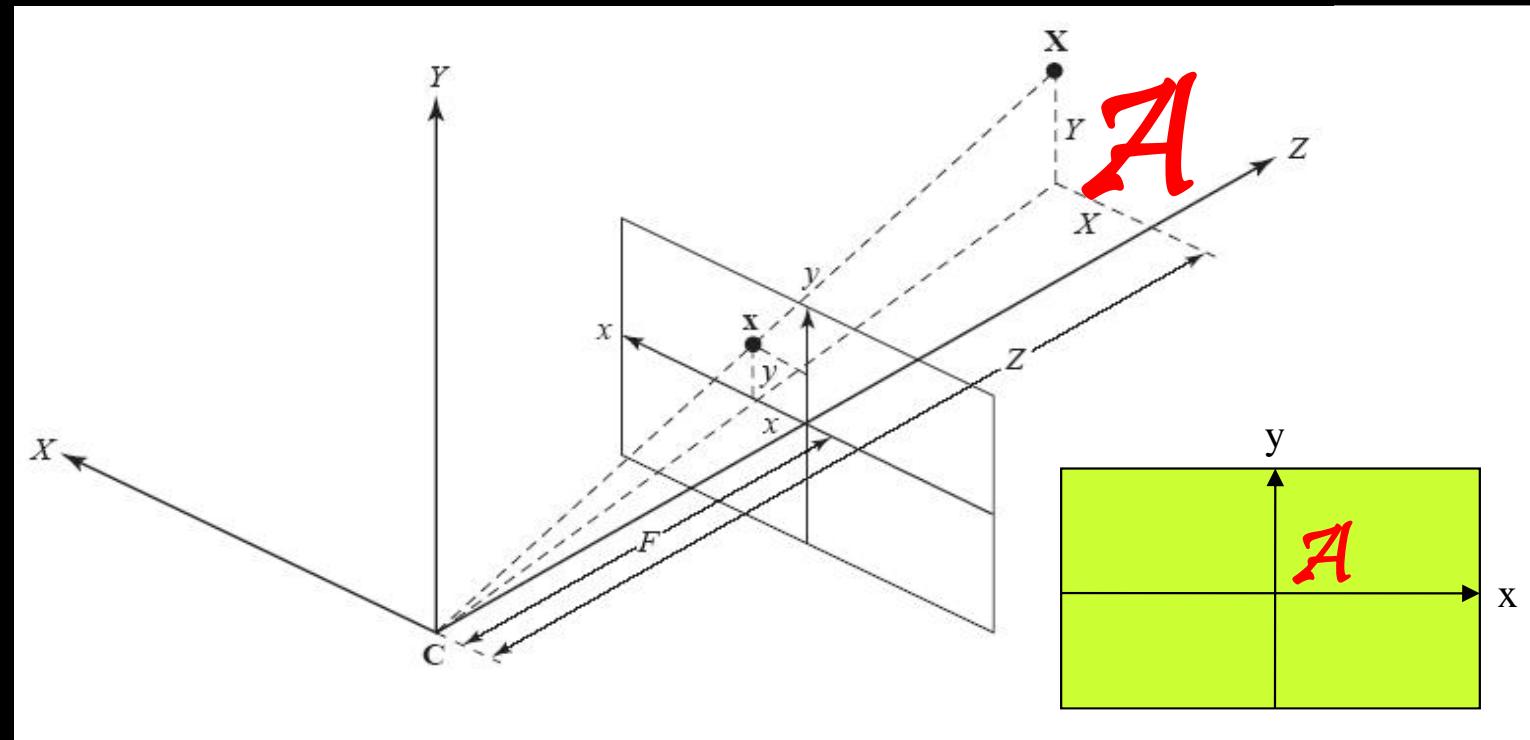
$$k_1 = 0.5 \cdot 10^{-6}, k_2 = 2.5 \cdot 10^{-11}$$

$$d(\mathbf{x}, \mathbf{x}') = k_1 r_{\mathbf{x}}^2 + k_2 r_{\mathbf{x}}^4 + \dots + k_n r_{\mathbf{x}}^{2n},$$

$$r_{\mathbf{x}} = d(\mathbf{x}, \mathbf{0}), \mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \mathbf{0} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}$$

$$\mathbf{x}' = \begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} x - x/2 \\ y - y/2 \end{bmatrix} \cdot d(\mathbf{x}, \mathbf{x}')$$

- Imágenes resultantes tras un proceso de proyección



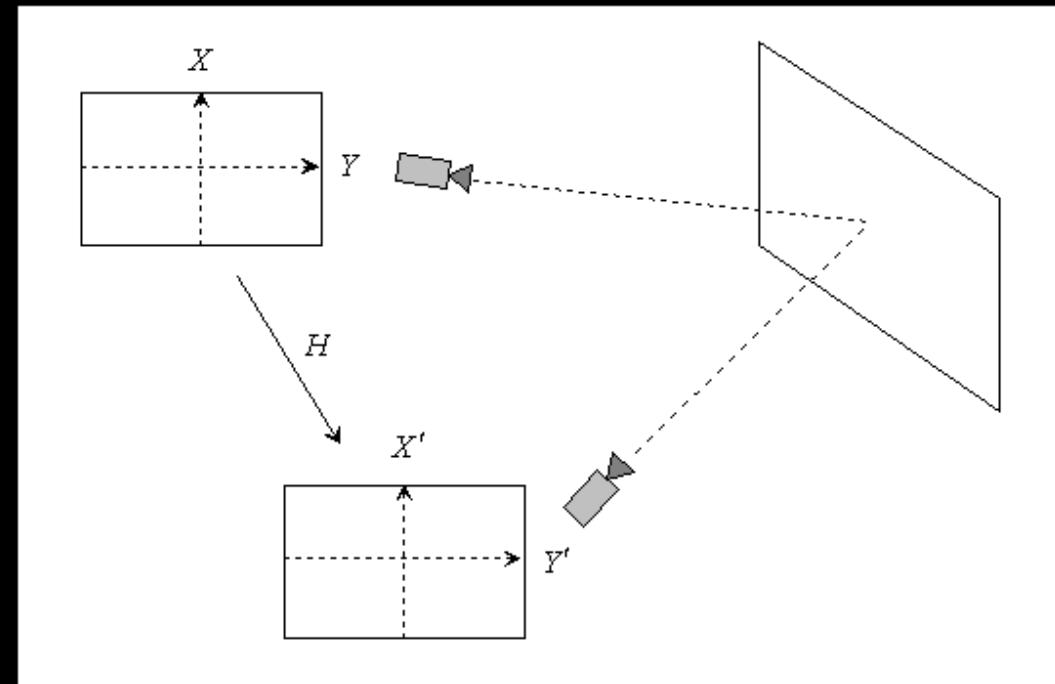
Ecuaciones que modelan el proceso de proyección en coordenadas afines:

$$\begin{aligned} \frac{x}{F} &= \frac{X}{Z} \Rightarrow x = F \frac{X}{Z} \\ \frac{y}{F} &= \frac{Y}{Z} \Rightarrow y = F \frac{Y}{Z} \end{aligned}$$

$$\mathbf{p} = \mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \frac{F}{Z} \begin{bmatrix} X \\ Y \end{bmatrix}$$

- Geometría proyectiva: marco lineal para el tratamiento de estas situaciones

- Homografías o transformaciones proyectivas
 - Describen la transformación que experimenta un plano proyectado cuando la posición del observador (o de la cámara varía).



$$\begin{bmatrix} X' \\ Y' \\ T' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \mathbf{x}' \approx \mathbf{H} \cdot \mathbf{x} \Rightarrow \begin{cases} x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \\ y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \end{cases}$$

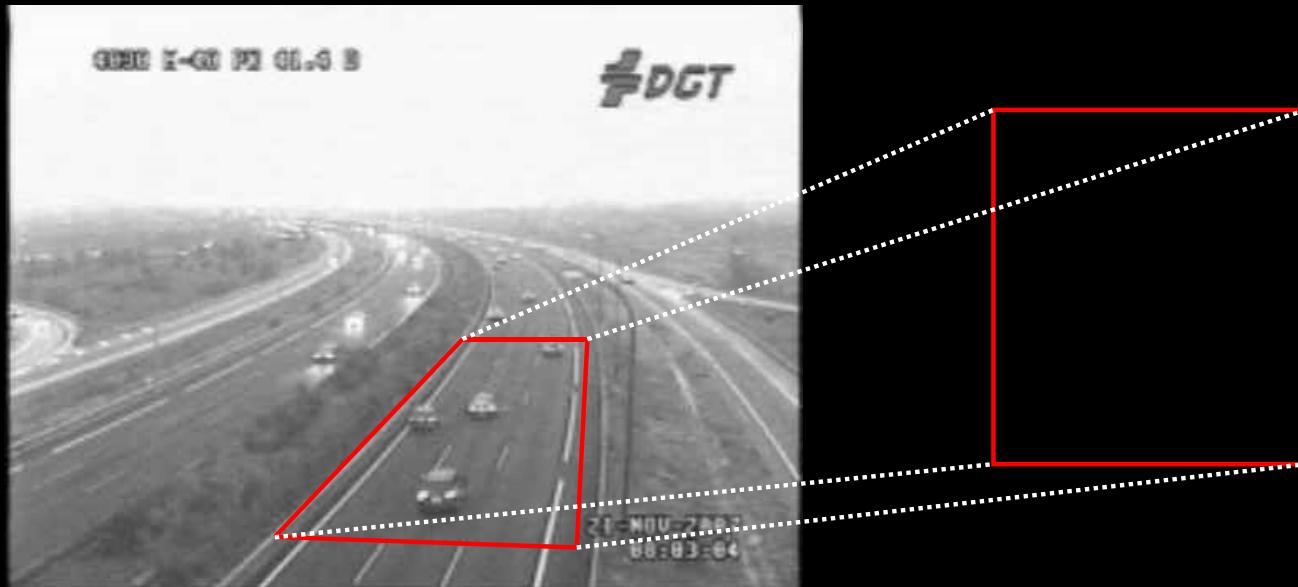
- Ajustes basados en homografías:
 - Cambio del punto de vista para facilitar la tarea de análisis



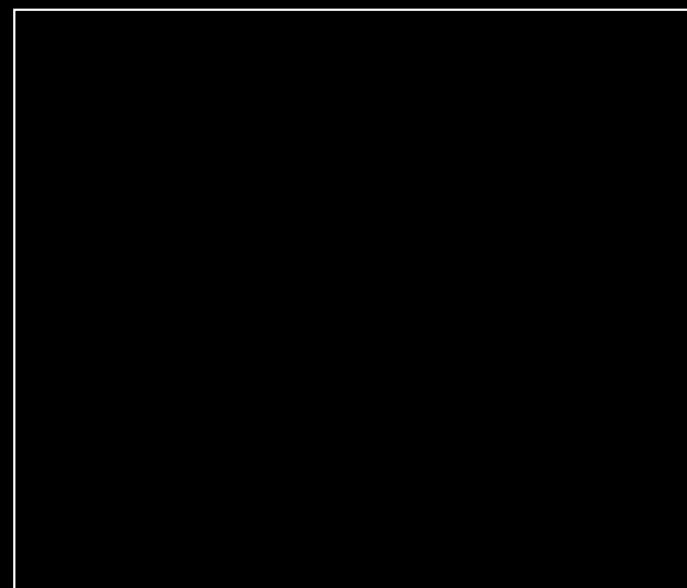
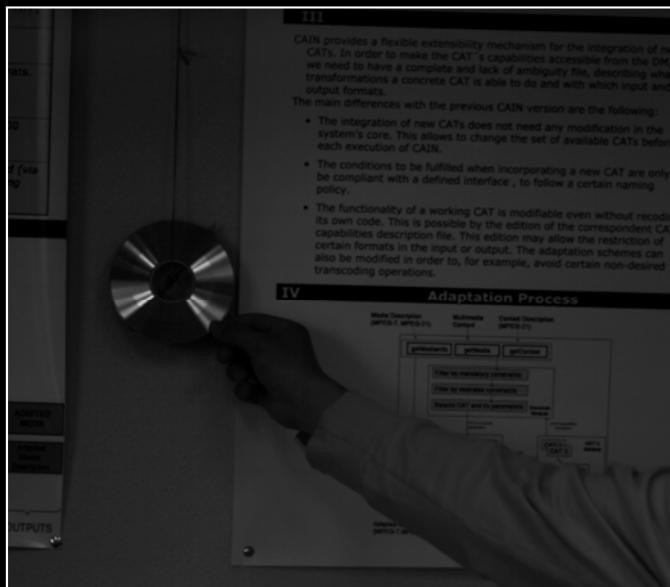
Se pueden tomar distancias y ángulos.
En la original no

- Ajustes basados en homografías:
 - Cambio del punto de vista para facilitar la tarea de análisis

Corrección del punto de vista para análisis cenital



- Ajuste imágenes captadas por distintas posiciones de la cámara



1. Introducción
2. Aspectos operativos
3. Filtrado espacial
4. Ajustes geométricos
5. Restauración lineal



■ Restauración lineal

- El objetivo es recuperar una imagen que ha sido degradada de una manera “conocida”. El método básico consiste en modelar linealmente la degradación y aplicar el proceso inverso.

$$\psi(x, y) \xrightarrow{T} \theta(x, y) = \psi(x, y) * h(x, y) + \eta(x, y)$$

$$\theta(x, y) \xrightarrow{iT?} \hat{\psi}(x, y) \square \psi(x, y)$$

- Modelos de ruido y estimación del ruido en la imagen degradada.
- Degradaciones LSI:

$$\Theta(u, v) = \Psi(u, v) \cdot H(u, v) + N(u, v)$$

- Filtrado inverso

$$\hat{\Psi}(u, v) = \frac{\Theta(u, v)}{H(u, v)} = \Psi(u, v) + \frac{N(u, v)}{H(u, v)}$$

- Asumiendo ruido nulo

$$\theta(x, y) = \psi(x, y) * h(x, y), \quad \eta(x, y) \square 0$$

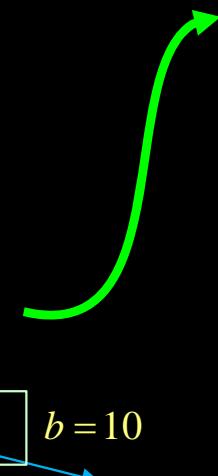
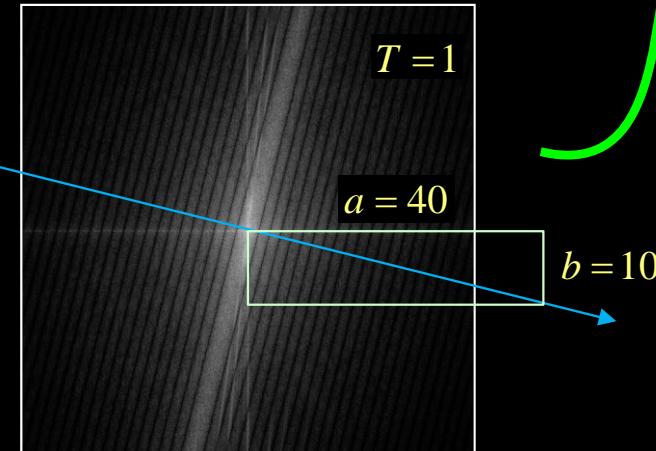


$$\theta(x, y) = \int_0^T \psi\left(x - \frac{at}{T}, y - \frac{bt}{T}\right) \cdot dt \Rightarrow h(x, y) = \int_0^T \delta\left(x - \frac{at}{T}, y - \frac{bt}{T}\right) \cdot dt \Rightarrow$$

$$\Rightarrow H(u, v) = \frac{T}{\pi(ua + vb)} \sin[\pi(ua + vb)] e^{-j\pi(ua + vb)}$$

$$\hat{\Psi}(u, v) = \frac{\Theta(u, v)}{H(u, v)} = \Psi(u, v) \xrightarrow{IFT} \psi(x, y)$$

$$\Theta(u, v) = \Psi(u, v) \cdot H(u, v)$$



— Ruido no nulo

$$\theta(x, y) = \psi(x, y) * h(x, y) + \eta(x, y)$$

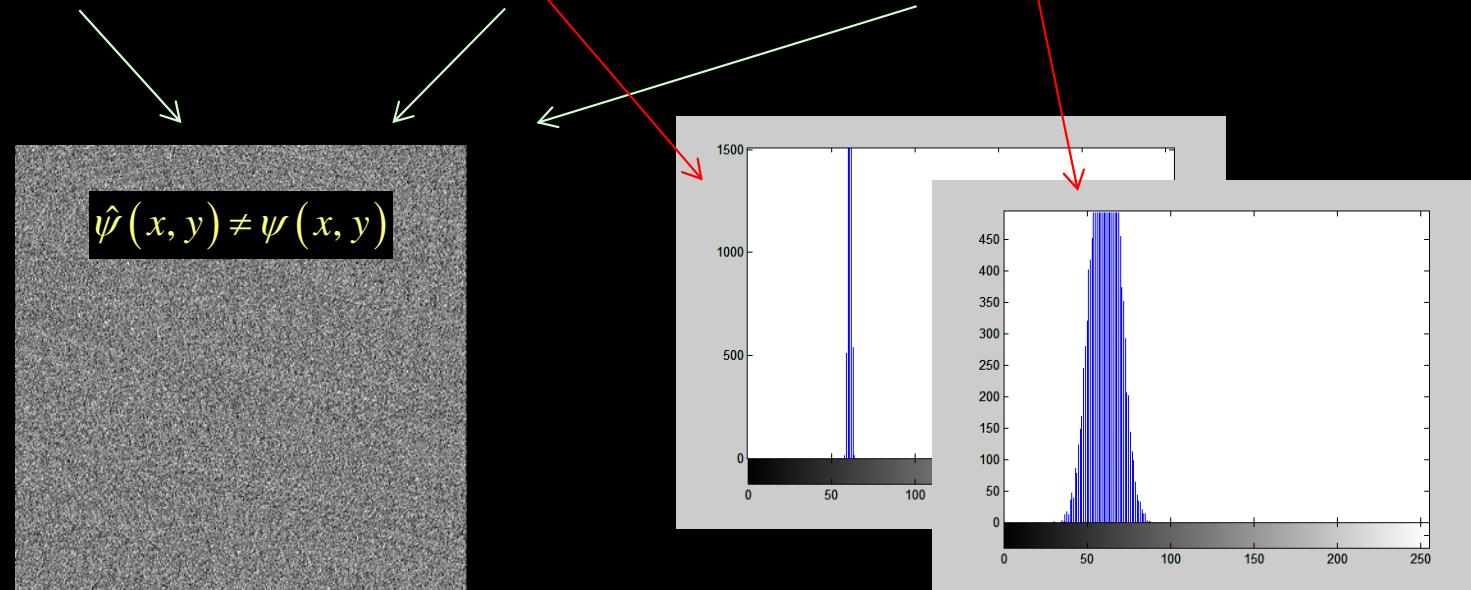
Ruido de cuantificación



Ruido de gaussiano ($1e-5$)



Ruido de gaussiano ($1e-3$)



- Otras soluciones: Filtros MMSE/Wiener o CLS
 - Filtro MMSE (Minimum Mean Square Error) o de Wiener
 - Basado en la consideración de que imágenes y ruido son procesos estacionarios, incorrelados entre sí y ambos de media nula, y en que los niveles de la imagen estimada dependen linealmente de los de la imagen degradada.
 - En la práctica se aplica en el dominio frecuencial mediante la expresión:

$$\hat{\Psi}(u,v) = \left[\frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K} \right] \Theta(u,v)$$

- Donde K es factor estimado de relación de potencia señal/ruido.

Restauración lineal

$$\theta(x, y) = \psi(x, y) * h(x, y) + \eta(x, y)$$

Ruido de cuantificación



Ruido de gaussiano (1e-5)



Ruido de gaussiano (1e-3)



$$\hat{\psi}(x, y)$$



— Filtro CLS (Constrained Least Squares)

- Se basa en maximizar la ‘suavidad’ de la imagen estimada resultante minimizando la energía de su laplaciana.
- En la práctica se aplica en el dominio frecuencial mediante la expresión:

$$\hat{\Psi}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma |L(u,v)|^2} \right] \Theta(u,v)$$

- Donde L es el filtro laplaciano y γ es el escalar que hay que seleccionar, o bien calcular para optimizar la solución.

Restauración lineal

$$\theta(x, y) = \psi(x, y) * h(x, y) + \eta(x, y)$$

Ruido de cuantificación



Ruido de gaussiano (1e-5)



Ruido de gaussiano (1e-3)



$$\hat{\psi}(x, y)$$

