Filtrado Espacial

Filtrado lineal y convolución

Alisamiento de imágenes

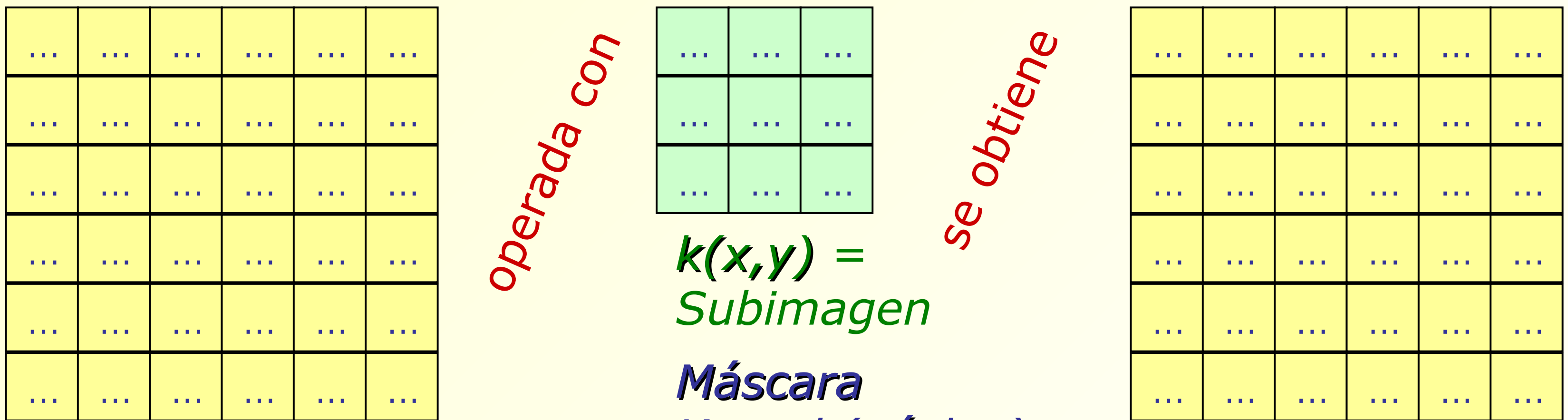
ugr Universidad de Granada

DECSAI

Computer Vision Group

En ocasiones, dado un píxel, es necesario **relacionarlo con sus vecinos** para obtener ciertos resultados.

Los píxeles de este vecindario son operados con los valores de una subimagen del tamaño de dicho vecindario.



f(x,y) = *Imagen original*

k(x,y) = *Subimagen*

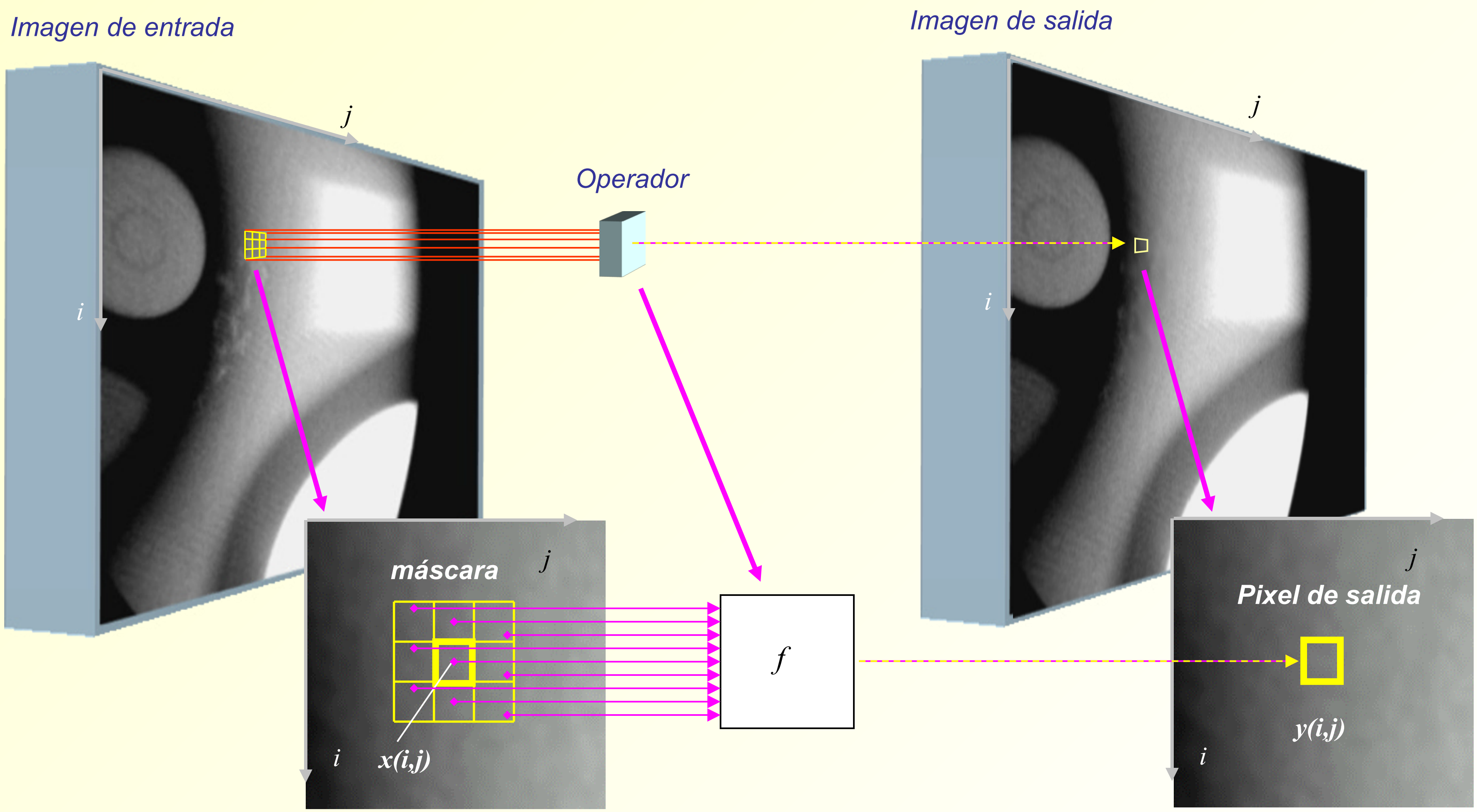
Máscara
Kernel (núcleo)
Filtro

r(x,y) = *Imagen resultante*

Uno por uno se van operando los píxeles de **f(x,y)** con el kernel **k** para obtener **r(x,y)**.

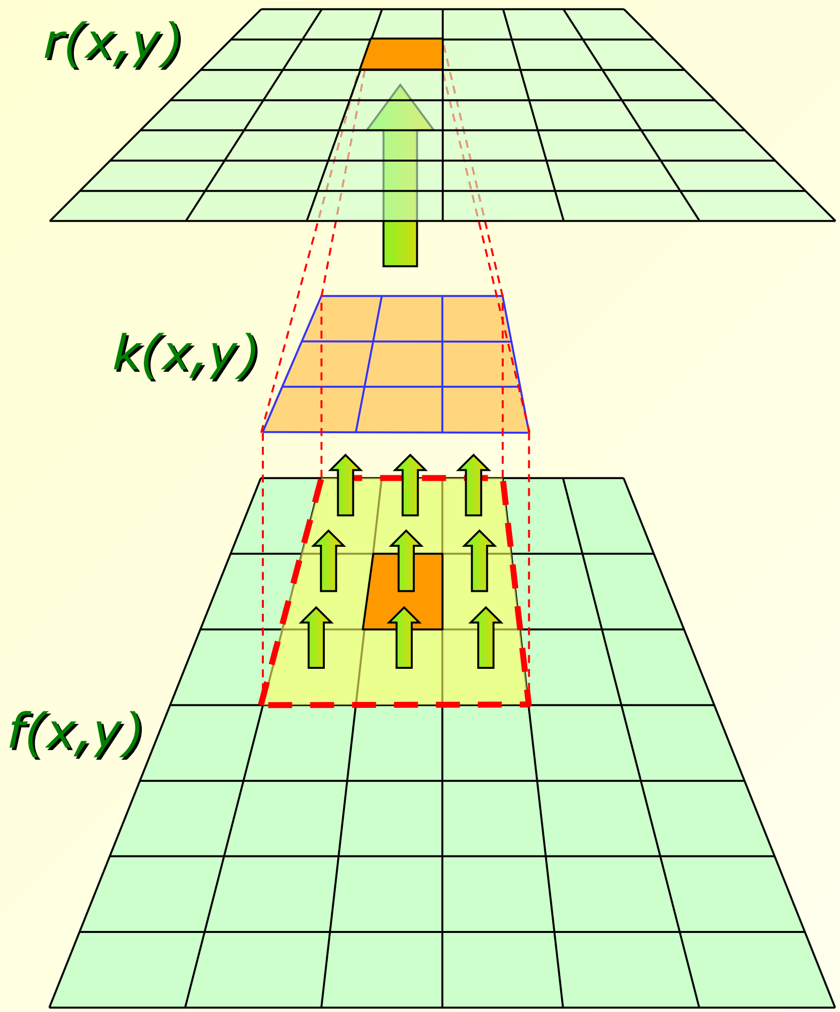
El filtrado espacial lineal

FILTROS



El filtrado espacial lineal: la convolución

El filtrado espacial **lineal** se define de esta forma (para cada píxel):



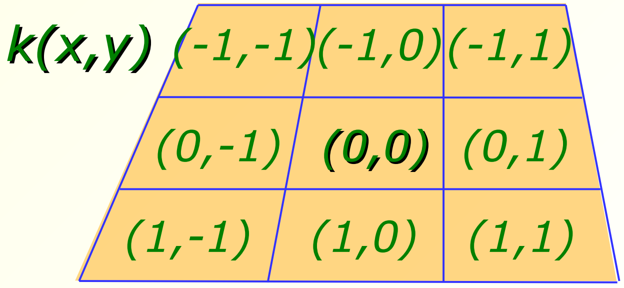
f(x,y) de tamaño *MxN*

k(x,y) de tamaño *mxn*

$m=2a+1 \qquad n=2b+1$

$$r(x,y)=\sum_{i=-a}^{a}\sum_{j=-b}^{b} k(i,j)*f(x-i,y-j)$$

Convolución = filtrado lineal espacial mediante la ecuación anterior.



Ejemplo de convolución

f(x,y)

3	1	2	6	5	2
4	6	1	0	8	4
4	8	2	3	7	1
5	7	3	9	1	0
2	5	8	1	8	6
3	6	4	5	2	1

k(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

r(x,y)

	4				

$$r(1,1)=\sum_{i=-1}^{1}\sum_{j=-1}^{1} k(i,j)*f(1+i,1+j)$$

r(1,1) = k(-1,-1)*f(0,0) + k(0,-1)*f(1,0) + k(1,-1)*f(2,0) +
+ k(-1,0)*f(0,1) + k(0,0)*f(1,1) + k(1,0)*f(2,1) +
+ k(-1,1)*f(0,2) + k(0,1)*f(1,2) + k(1,1)*f(2,2) = **3.7 ≈ 4**

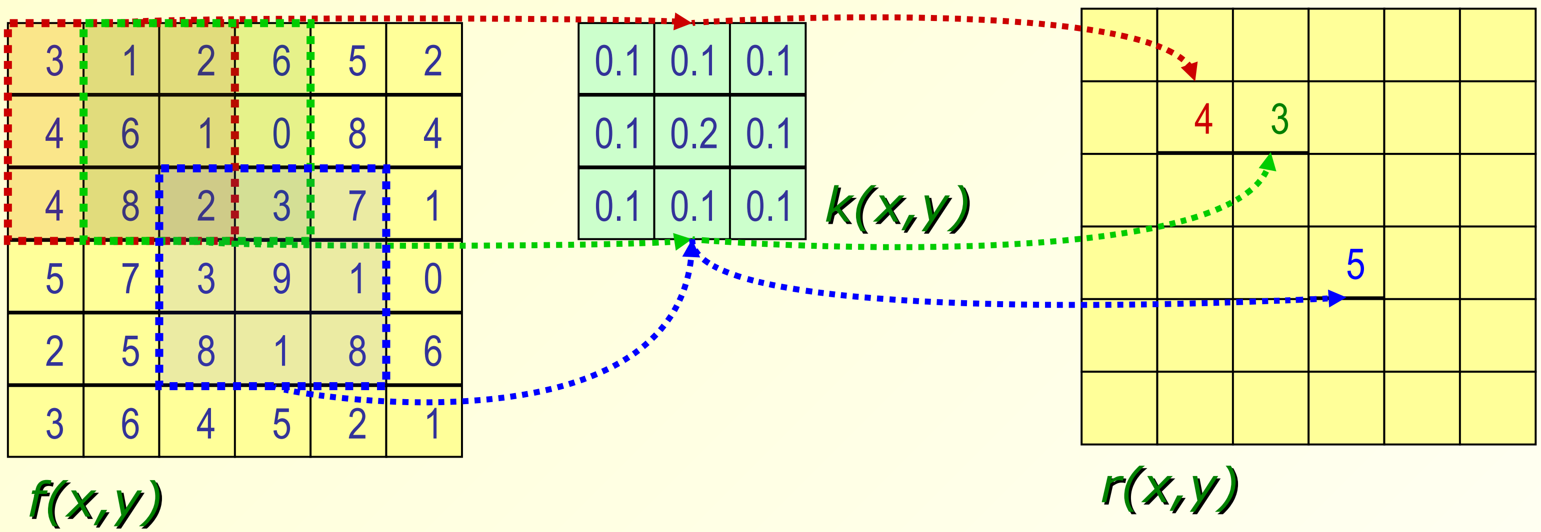
k(x,y)

(-1,-1)	(-1,0)	(-1,1)
(0,-1)	**(0,0)**	(0,1)
(1,-1)	(1,0)	(1,1)

f(x,y)

(0,0)	(0,1)	(0,2)
(1,0)	**(1,1)**	(1,2)
(2,0)	(2,1)	(2,2)

Ejemplo de convolución



$$
\begin{aligned}
r(1,1) = \; & f(2,2)*k(0,0) + f(2,1)*k(0,1) + f(2,0)*k(0,2) + \\
& + f(1,2)*k(1,0) + f(1,1)*k(1,1) + f(1,0)*k(1,2) + \\
& + f(0,2)*k(2,0) + f(0,1)*k(2,1) + f(0,0)*k(2,2) = 3.7 \approx 4
\end{aligned}
$$

$$
\begin{aligned}
r(1,2) = \; & f(2,3)*k(0,0) + f(2,2)*k(0,1) + f(2,1)*k(0,2) + \\
& + f(1,3)*k(1,0) + f(1,2)*k(1,1) + f(1,1)*k(1,2) + \\
& + f(0,3)*k(2,0) + f(0,2)*k(2,1) + f(0,1)*k(2,2) = 3
\end{aligned}
$$

Con este kernel cada nuevo píxel es la media (ponderada) de los píxeles vecinos.

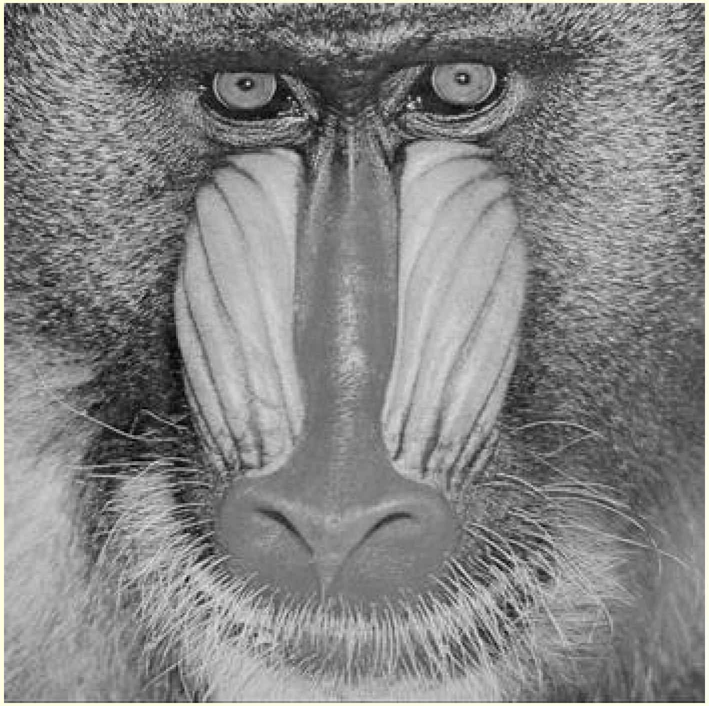
0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

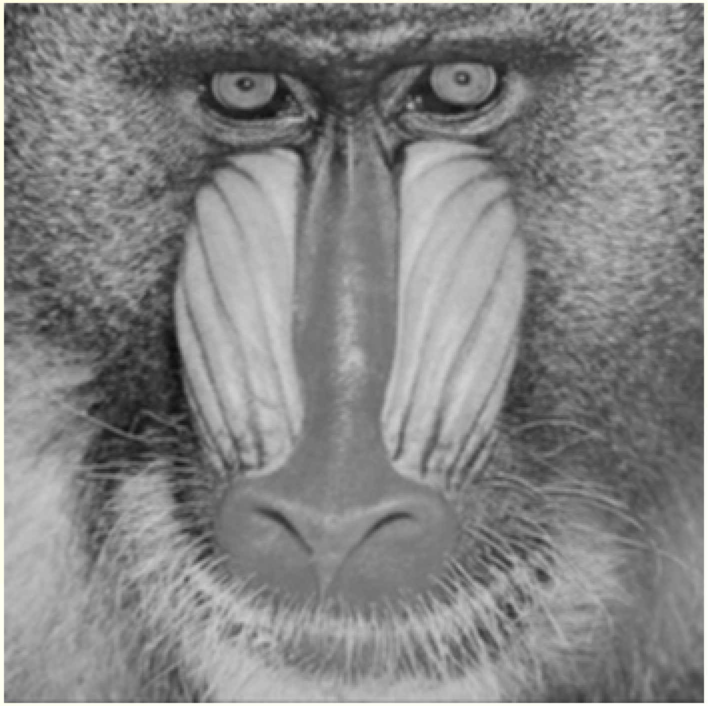
$k(x,y)$

$$\frac{1}{10}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

El núcleo suele estar compuesto por números enteros.

Su tamaño suele ser impar y cuadrado.





El tratamiento de los bordes

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	?
3	6	4	5	2	1	?
				?	?	?

f(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

k(x,y)

	4	3			
			5		
					?

r(x,y)

¿Qué ocurre con los píxeles de los bordes?

1.- No considerarlos. Ponerlos a cero.

2.- Suponer que valen 0 en f(x,y).

3.- Suponer que valen el último valor de la fila o columna.

4.- Considerar la imagen toroidal.

Etc.

El tratamiento de los bordes: ignorarlos

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	?
3	6	4	5	2	1	?
				?	?	?

f(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

k(x,y)

	4	3			
			5		
					?

r(x,y)

3	1	2	6	5	2
4	6	1	0	8	4
4	8	2	3	7	1
5	7	3	9	1	0
2	5	8	1	8	6
3	6	4	5	2	1

f(x,y)

0	**0**	**0**	**0**	**0**	**0**
0	4	3			**0**
0					**0**
0			5		**0**
0					**0**
0	**0**	**0**	**0**	**0**	**0**

r(x,y)

El tratamiento de los bordes: f(x,y)=0

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	?
3	6	4	5	2	1	?
				?	?	?

f(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

k(x,y)

	4	3			
			5		
					?

r(x,y)

2	3	7	1	
3	9	1	0	
8	1	8	6	
4	5	2	1	0
			0	0

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	**0**
3	6	4	5	2	1	**0**
				0	**0**	**0**

f(x,y)

	4	3			
			5		
					2

r(x,y)

El tratamiento de los bordes: continuación

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	?
3	6	4	5	2	1	?
				?	?	?

f(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

k(x,y)

	4	3			
			5		
					?

r(x,y)

2	3	7	1		
3	9	1	0		
8	1	8	6	6	6
4	5	2	1	1	1
		2	1	1	1
		2	1	1	1

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	**6**
3	6	4	5	2	1	**1**
				2	**1**	**1**

f(x,y)

	4	3			
			5		
					3

r(x,y)

El tratamiento de los bordes: imagen toroidal

3	1	2	6	5	2	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	?
3	6	4	5	2	1	?
				?	?	?

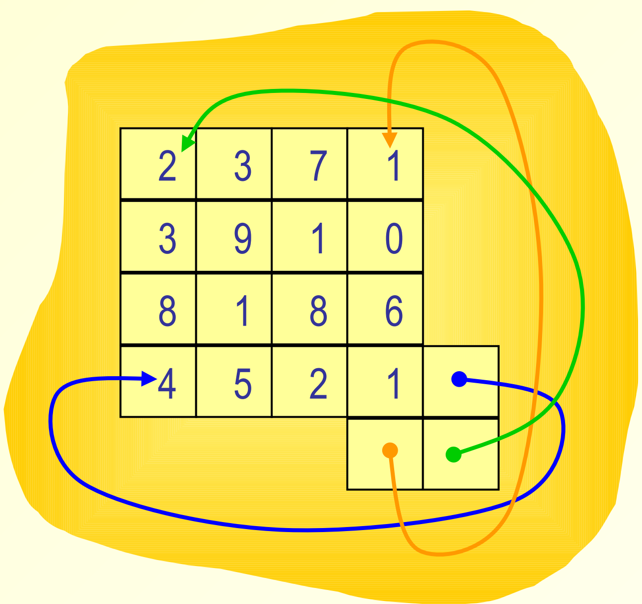
f(x,y)

0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

k(x,y)

	4	3			
			5		
					?

r(x,y)



3	1	2	6	**5**	**2**	
4	6	1	0	8	4	
4	8	2	3	7	1	
5	7	3	9	1	0	
2	5	8	1	8	6	**2**
3	6	4	5	2	1	**3**
				5	**2**	**3**

f(x,y)

	4	3			
			5		
					3

r(x,y)

Filtrado Espacial

Filtrado lineal y convolución

Alisamiento de imágenes

Universidad de Granada

DECSAI

Computer Vision Group

Filtros lineales de suavizado (media)

El objetivo de los filtros de suavizado es eliminar parte del ruido presente en las imágenes.

Añadido ruido gaussiano con $\sigma^2=21$

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

Filtros lineales de suavizado (media)

El objetivo de los filtros de suavizado es eliminar parte del ruido presente en las imágenes.

Añadido ruido gaussiano con $\sigma^2=21$

$\frac{1}{25}$

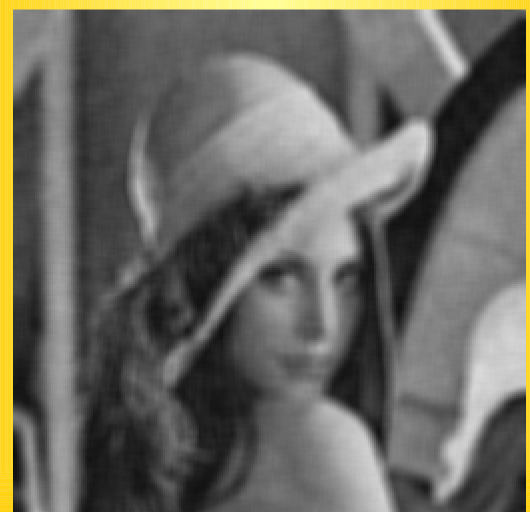
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

Filtros lineales de suavizado (media)

El objetivo de los filtros de suavizado es eliminar parte del ruido presente en las imágenes.

Añadido ruido gaussiano con $\sigma^2=21$

$$\frac{1}{49}\begin{array}{|c|c|c|c|c|c|c|}\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline\end{array}$$



Filtros lineales de suavizado (media ponderada)

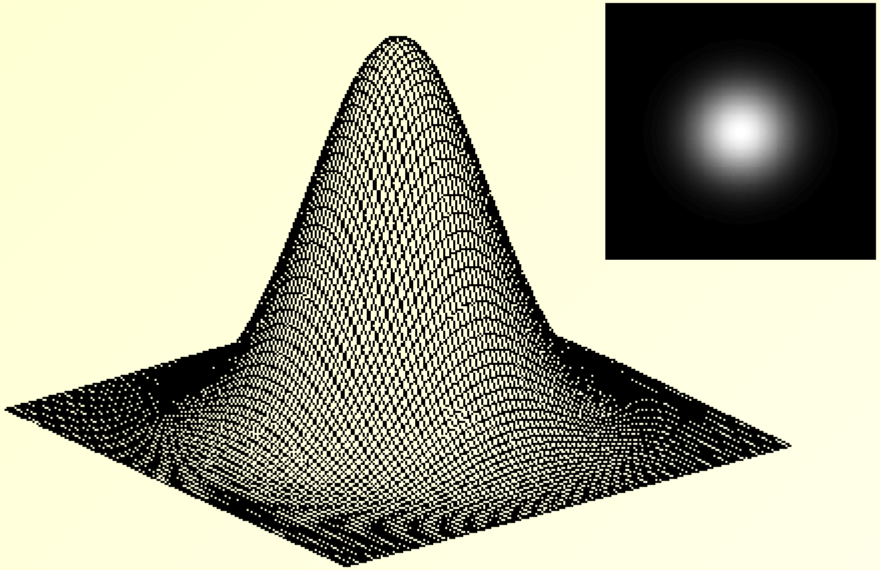
Podemos poner los coeficientes de manera que unos tengan más peso que otros: media ponderada.

$$\frac{1}{66}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 5 & 5 & 5 & 1 \\ 1 & 5 & 10 & 5 & 1 \\ 1 & 5 & 5 & 5 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Al dar más importancia al centro el entorno tiene menos influencia y se conserva mejor la imagen.

Filtros lineales de suavizado (gaussiana)

Es una media ponderada.



$$k(x,y) = A \exp\left(\frac{-(x^2+y^2)}{2\sigma^2}\right)$$

A se calcula para que la suma de todos los coeficientes sea 1.

Es el filtro de suavizado más utilizado. Tiene buenas propiedades matemáticas.

Caso 3x3, σ=0.5:

0.011343736	0.083819506	0.011343736
0.083819506	0.619347030	0.083819506
0.011343736	0.083819506	0.011343736

Filtros lineales de suavizado (gaussiana)

Alisamiento de imágenes

Filtros no lineales de suavizado (mediana)

La mediana es un filtro no lineal. Cada píxel se calcula como la mediana de los píxeles de su entorno.

*1, 1, 2, 2, **3**, 4, 4, 6, 8*

*0, 0, 1, 1, **3**, 4, 7, 8, 9*

*1, 2, 3, 3, **5**, 7, 8, 8, 9*

f(x,y)

3	1	2	6	5	2
4	6	1	0	8	4
4	8	2	3	7	1
5	7	3	9	1	0
2	5	8	1	8	6
3	6	4	5	2	1

r(x,y)

	3				
				3	
		5			

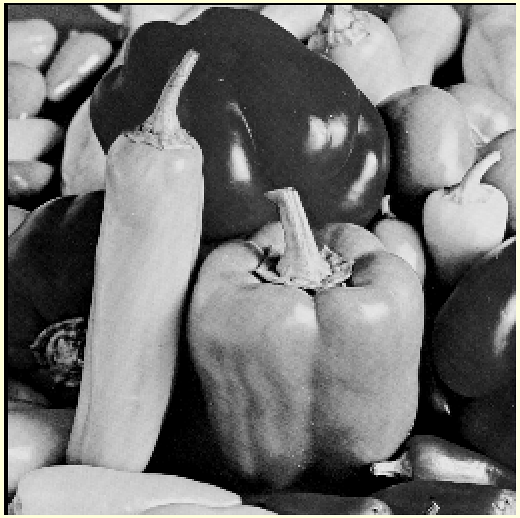
Es especialmente bueno eliminando ruido de tipo sal y pimienta.

Los puntos de ruido suelen tener valores muy alejados de los de su entorno por lo que al ordenarlos se quedan en los extremos de la secuencia ordenada.

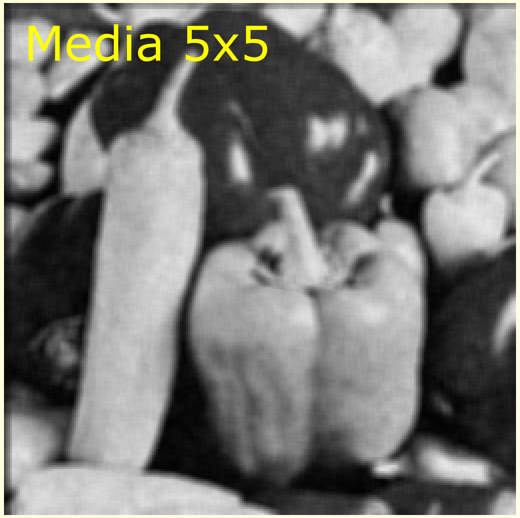
Filtros no lineales de suavizado (mediana)

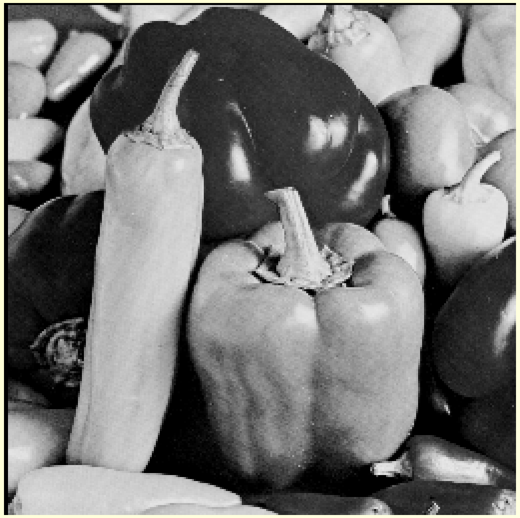
Filtros no lineales de suavizado (mediana)

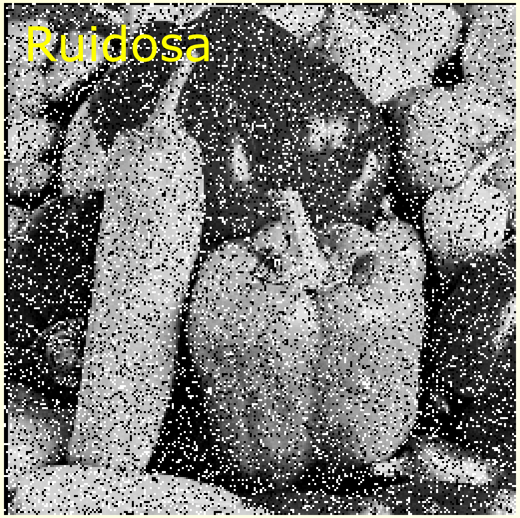
Añadido ruido "sal y pimienta" al 25% de la imagen



Ruido gaussiano
(σ^2=21)







Ruido "sal y pimienta" (25%)