

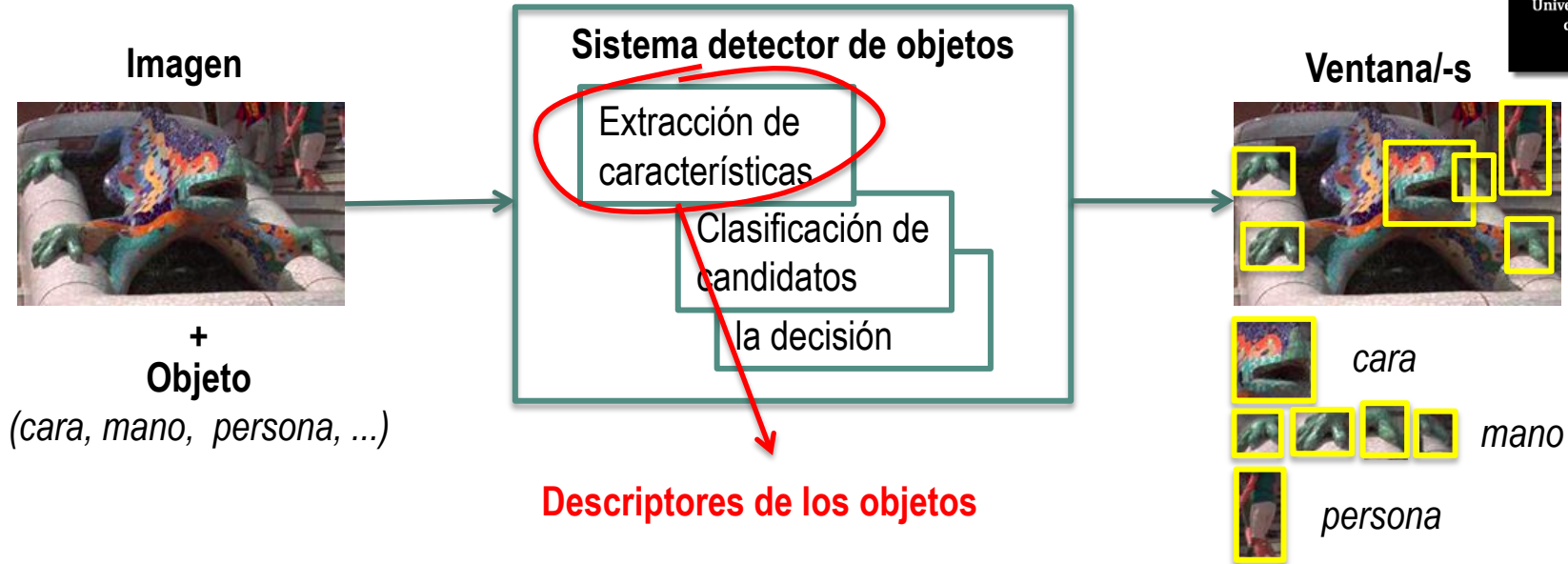
# INTRODUCCIÓN A LA DETECCIÓN DE OBJETOS

## Características locales

**Maria Vanrell**

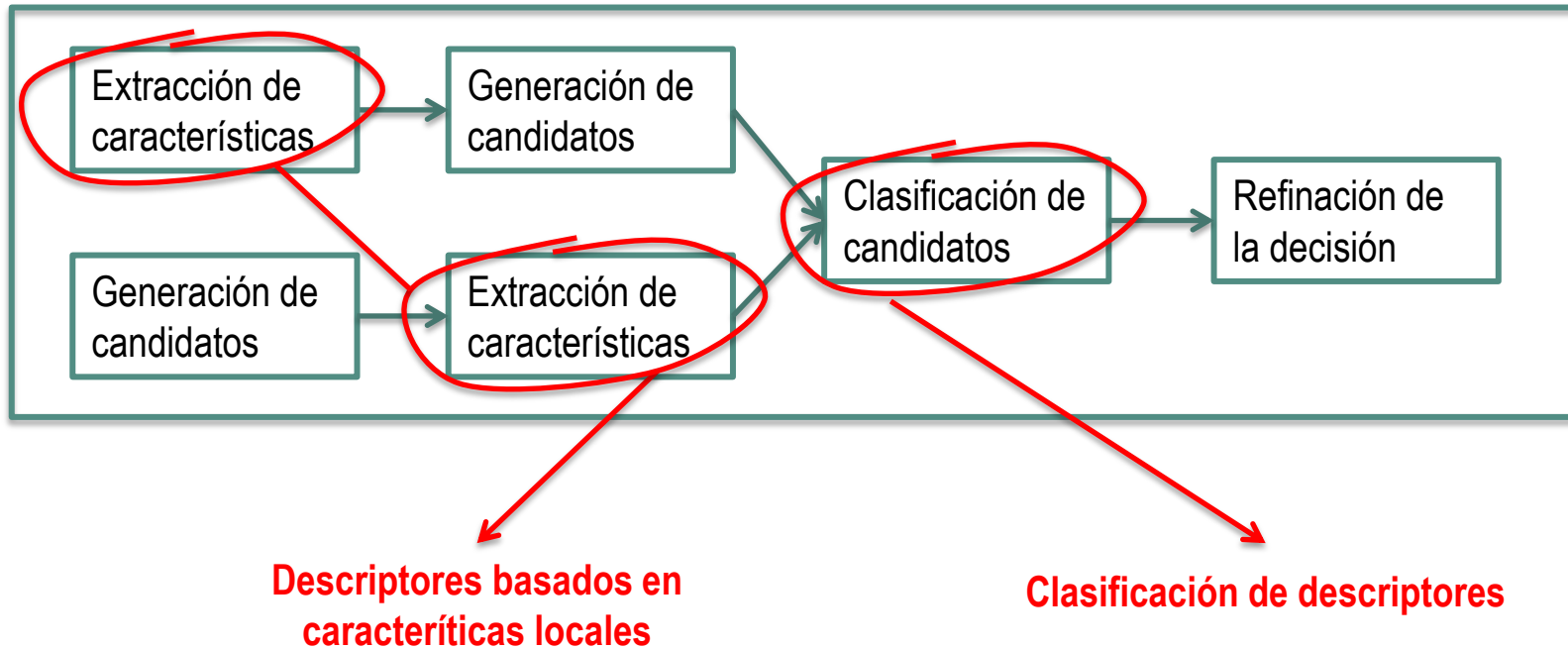
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

## Descriptores de objetos

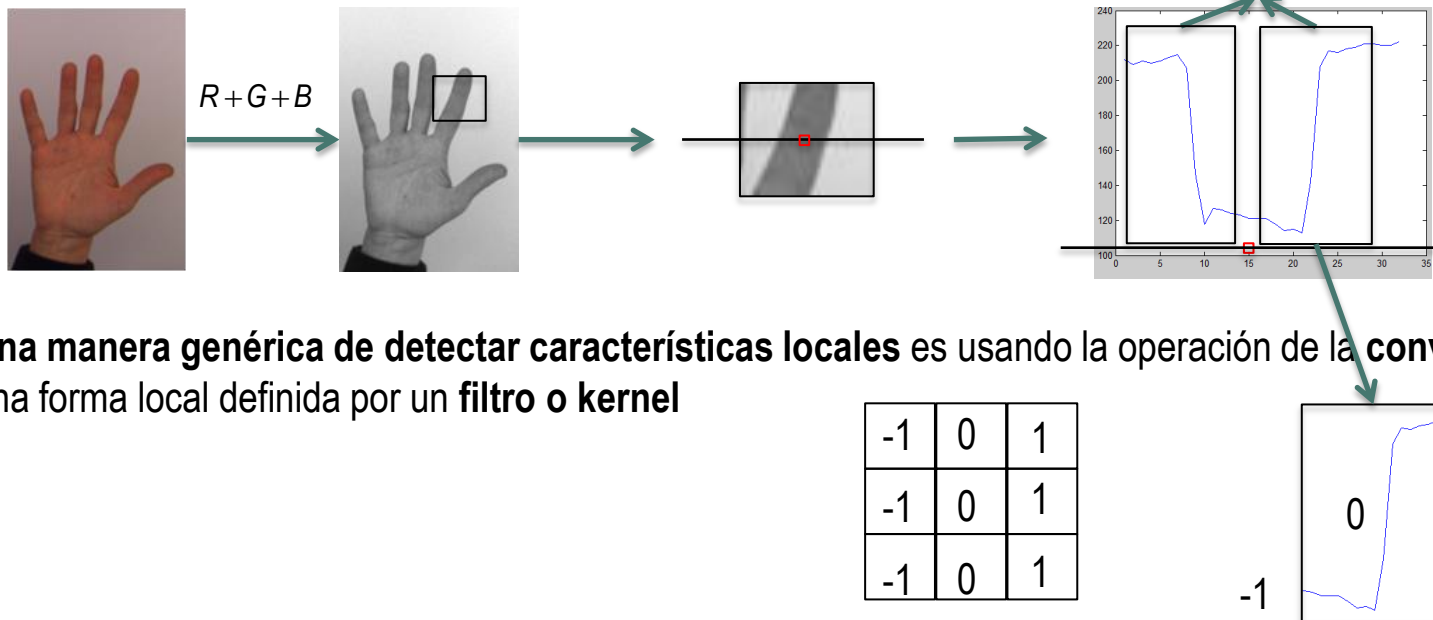


- basados en características del píxel
- basados en toda la imagen
- basados en características locales

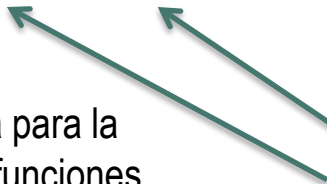
## Esquema general:



**Característica local** (*local feature*) : Una característica local es una propiedad que describe las propiedades de un píxel con relación a su vecindad. Estas características pueden presentar formas específicas con una estructura propia, tales como puntos (*blobs*) o contornos (*edges*), o cualquier otra estructura que se adapte a formas habituales.



**La convolución:** es una operación matemática entre dos funciones,  $f$  y  $g$ , que se representa como una nueva función que nos da el grado de solapamiento de la función  $g$  sobre la función  $f$ .

$$(f * g)(x, y) = \sum_{x', y'} g(x', y') \cdot f(x - x', y - y')$$


**En teoría de la señal**, la convolución es usada para la descomposición de una señal en una serie de funciones más simples.

$f$  : Señal de entrada

$g$  : Funciones básicas

---

**En visión por computador** se utiliza

- para suavizar la imagen (blurring),
- para mejorar la imagen (sharpening), o
- para extracción de características locales (contornos, puntos, ...)

En **visión por computador** la función  $f$  normalmente representa la imagen y la función  $g$  la característica local, de la que queremos medir su solapamiento con la imagen original.

$$(f * g)(x, y) = \sum_{x', y'} g(x', y') \cdot f(x - x', y - y')$$

$$r(x, y) = l(x, y) * k(x, y) = \sum_{x', y'} k(x', y') \cdot l(x - x', y - y')$$

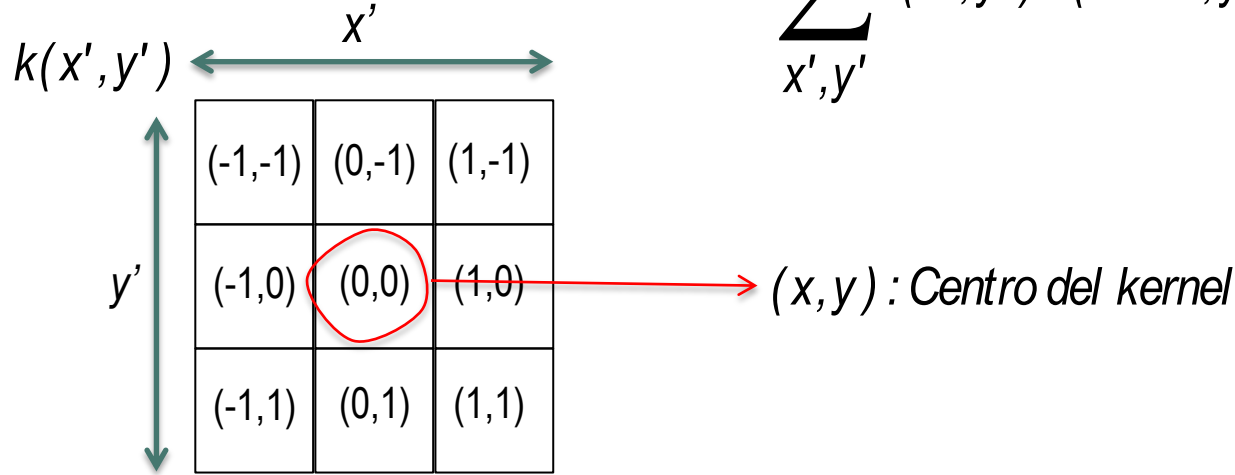
$k$  : kernel/filtro       $l$  : Imagen

**Observación:** sobre el cálculo de la convolución

$$r(x, y) = l(x, y) * k(x, y) =$$

$$= \sum_{x', y'} k(x', y') \cdot l(x - x', y - y')$$

**Ejemplo:** supongamos un kernel 3x3



$$\begin{aligned} r(x, y) = & k(-1, -1) \cdot l(x + 1, y + 1) + k(0, -1) \cdot l(x, y + 1) + k(1, -1) \cdot l(x - 1, y + 1) + \\ & + k(-1, 0) \cdot l(x + 1, y) + k(0, 0) \cdot l(x, y) + k(1, 0) \cdot l(x - 1, y) \\ & + k(-1, 1) \cdot l(x + 1, y - 1) + k(0, 1) \cdot l(x, y - 1) + k(1, 1) \cdot l(x - 1, y - 1) \end{aligned}$$

La **convolución** se basa en el producto de la ventana de la imagen con el filtro invertido

$$\begin{aligned}
 r(x,y) = & \boxed{k(-1,-1) \cdot I(x+1,y+1)} + k(0,-1) \cdot I(x,y+1) + \boxed{k(1,-1) \cdot I(x-1,y+1)} + \\
 & + k(-1,0) \cdot I(x+1,y) + \boxed{k(0,0) \cdot I(x,y)} + k(1,0) \cdot I(x-1,y) \\
 & + k(-1,1) \cdot I(x+1,y-1) + k(0,1) \cdot I(x,y-1) + k(1,1) \cdot I(x-1,y-1)
 \end{aligned}$$

$h(x',y')$

<span style="border: 2px solid red; padding: 2px;">(-1,-1)</span>	(0,-1)	<span style="border: 2px solid blue; padding: 2px;">(1,-1)</span>
(-1,0)	<span style="border: 2px solid green; padding: 2px;">(0,0)</span>	(1,0)
(-1,1)	(0,1)	(1,1)

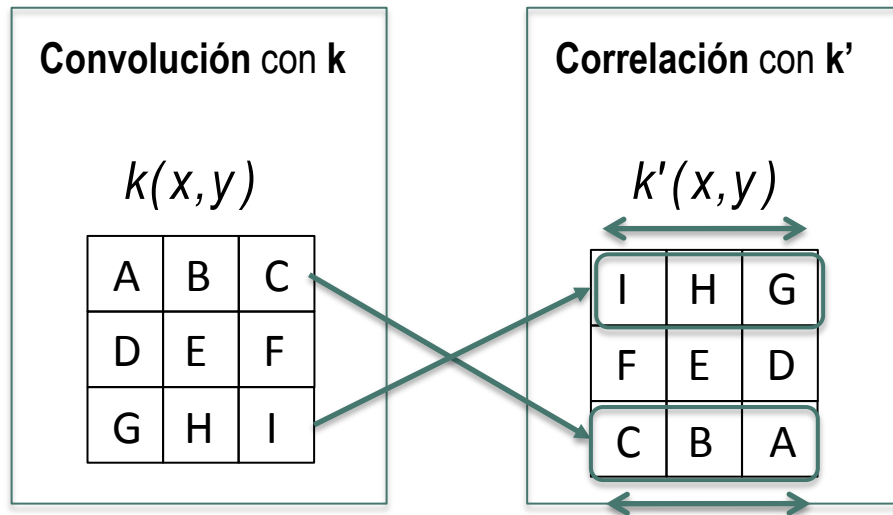
$I(x,y)$

$(x-1,y-1)$	$(x,y-1)$	$(x+1,y-1)$
$(x-1,y)$	<span style="border: 2px solid green; padding: 2px;"><math>(x,y)</math></span>	$(x+1,y)$
<span style="border: 2px solid blue; padding: 2px;"><math>(x-1,y+1)</math></span>	$(x,y+1)$	<span style="border: 2px solid red; padding: 2px;"><math>(x+1,y+1)</math></span>



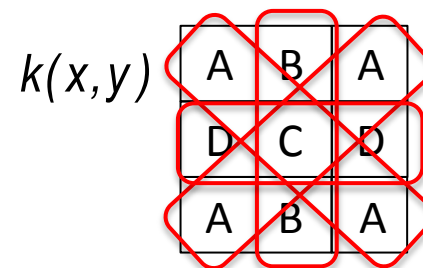
## Relación entre la convolución y la correlación:

La convolución es una correlación con el filtro invertido.



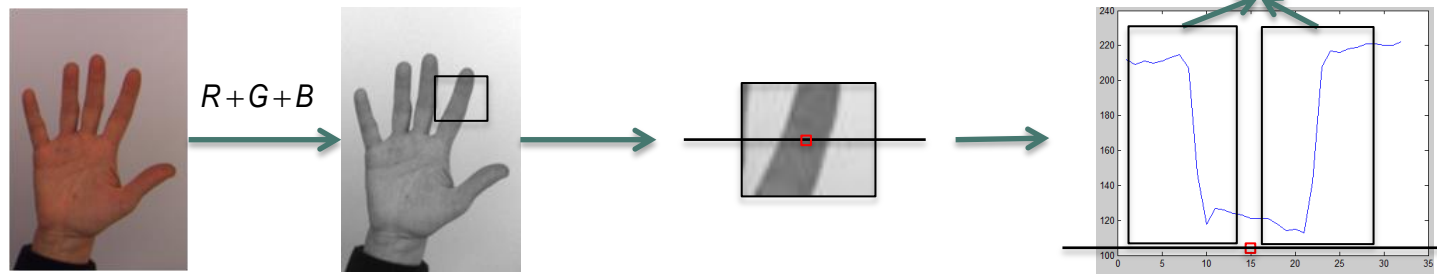
**Importante:**

**Filtro simétrico**



**Convolución = Correlación**

## Característica local (*local feature*) : el contorno



## Posibles filtros detectores de contorno:

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



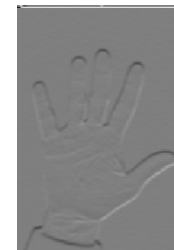
1	0	-1
1	0	-1
1	0	-1



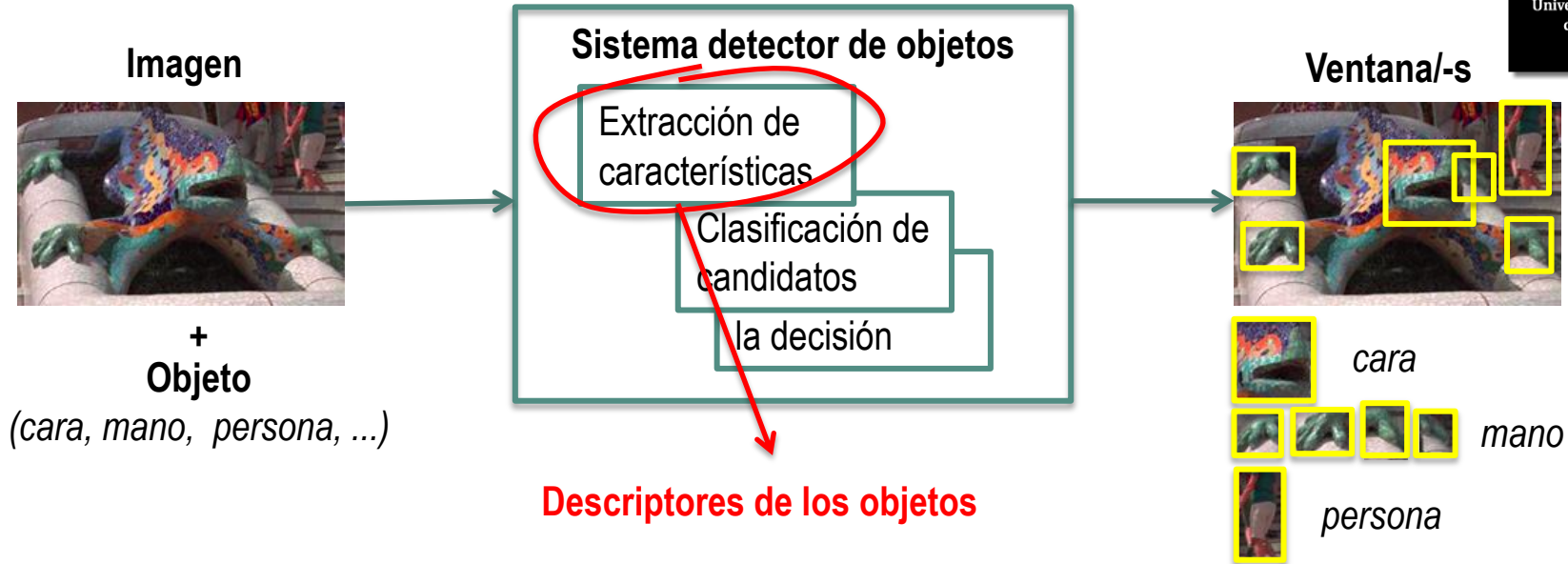
-1	-1	-1
0	0	0
1	1	1



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



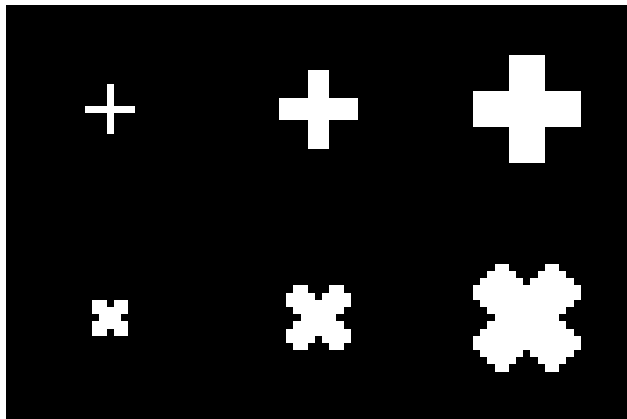
## Descriptores de objetos



- basados en características del píxel
- basados en toda la imagen
- basados en características locales

**Ejemplo:** Representación de objetos simples usando la convolución.

Imagen:  $I(x, y)$



Negro: 0  
Blanco: 255

Kernels:  $k(x, y)$

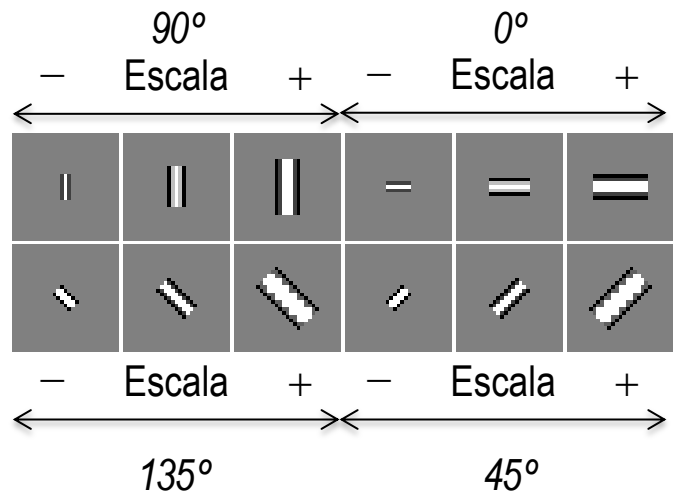
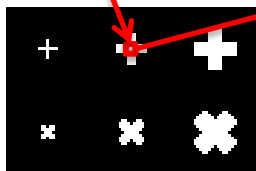


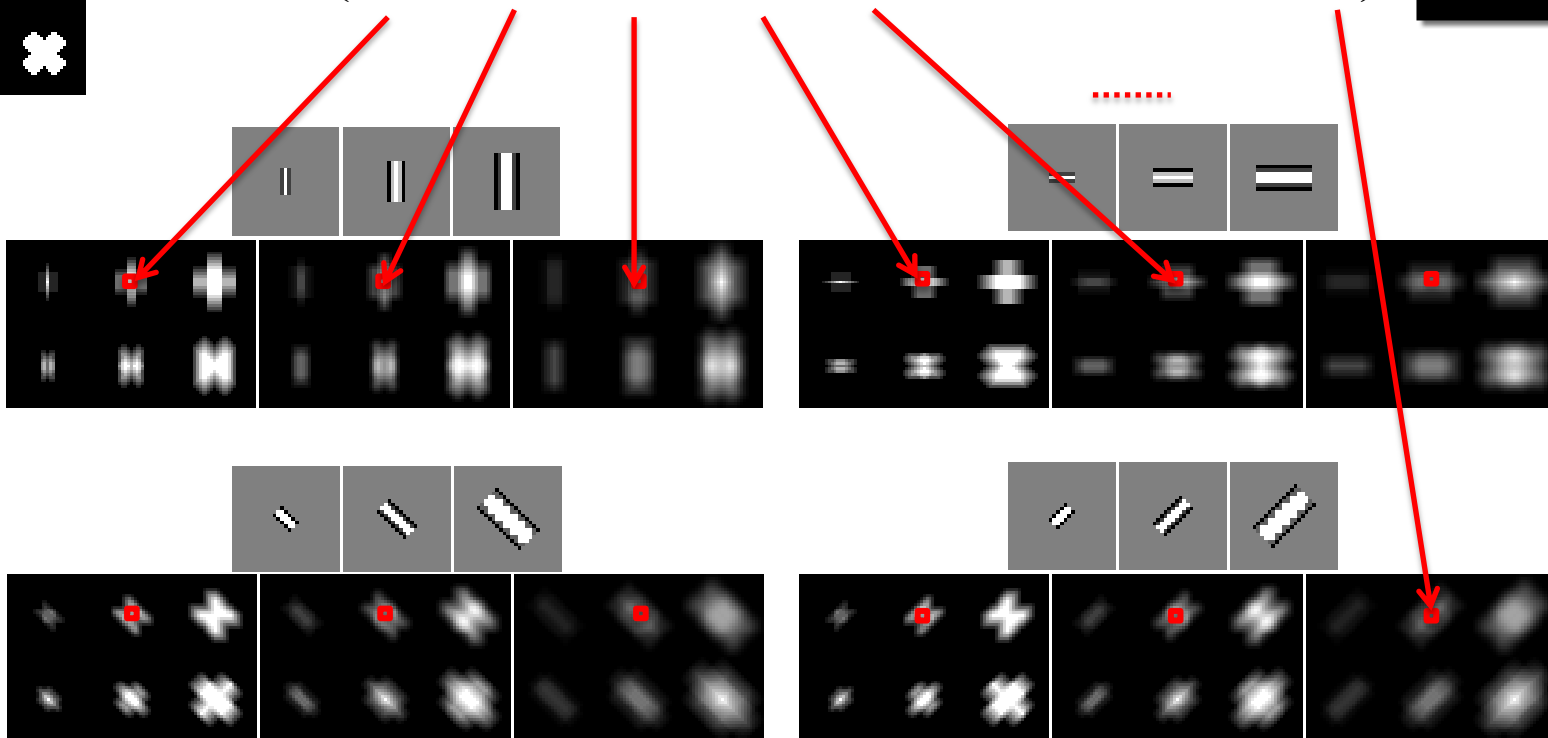
Imagen:  $I(x, y)$

$p : \text{píxel}$



$\text{Descriptor}(p) = (x_1 \ x_2 \ \dots \ x_n) =$

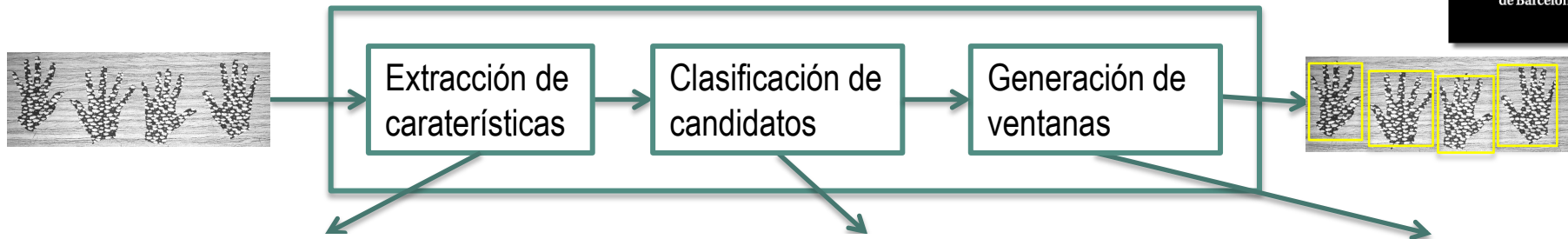
$$= (x_{s=1}^{90^\circ} \ x_{s=2}^{90^\circ} \ x_{s=3}^{90^\circ} \ x_{s=1}^{0^\circ} \ x_{s=2}^{0^\circ} \ x_{s=3}^{0^\circ} \ x_{s=1}^{135^\circ} \ \dots \ x_{s=3}^{45^\circ})$$



**UAB**

Universitat Autònoma  
de Barcelona

**Esquema:** (el mismo esquema usado en características de pixel)



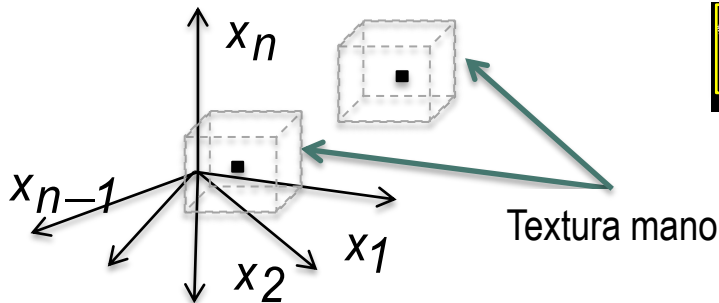
Para cada píxel,  $p(x,y)$ , le asociamos un descriptor basado en un conjunto de características locales

$$\text{Descriptor}(p) = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} \text{filtro}_1 \\ \vdots \\ \text{filtro}_n \end{pmatrix}$$

Filtros:



Cada píxel se clasificará dentro o fuera de la clase definida por cada textura en el espacio de dimensión  $n$

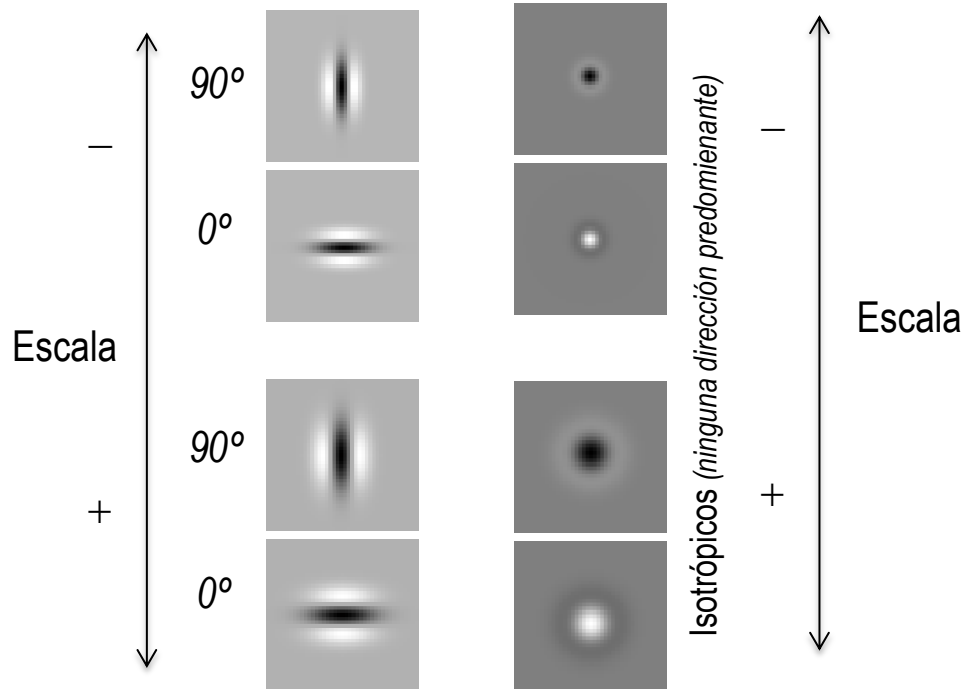


Cerrar las regiones etiquetadas y localizar las ventanas



**Filtros seleccionados:** para la detección de las manos con textura se ha trabajado con un conjunto de 8 filtros básicos.

Kernels:  $k(x, y)$



**Nota:**

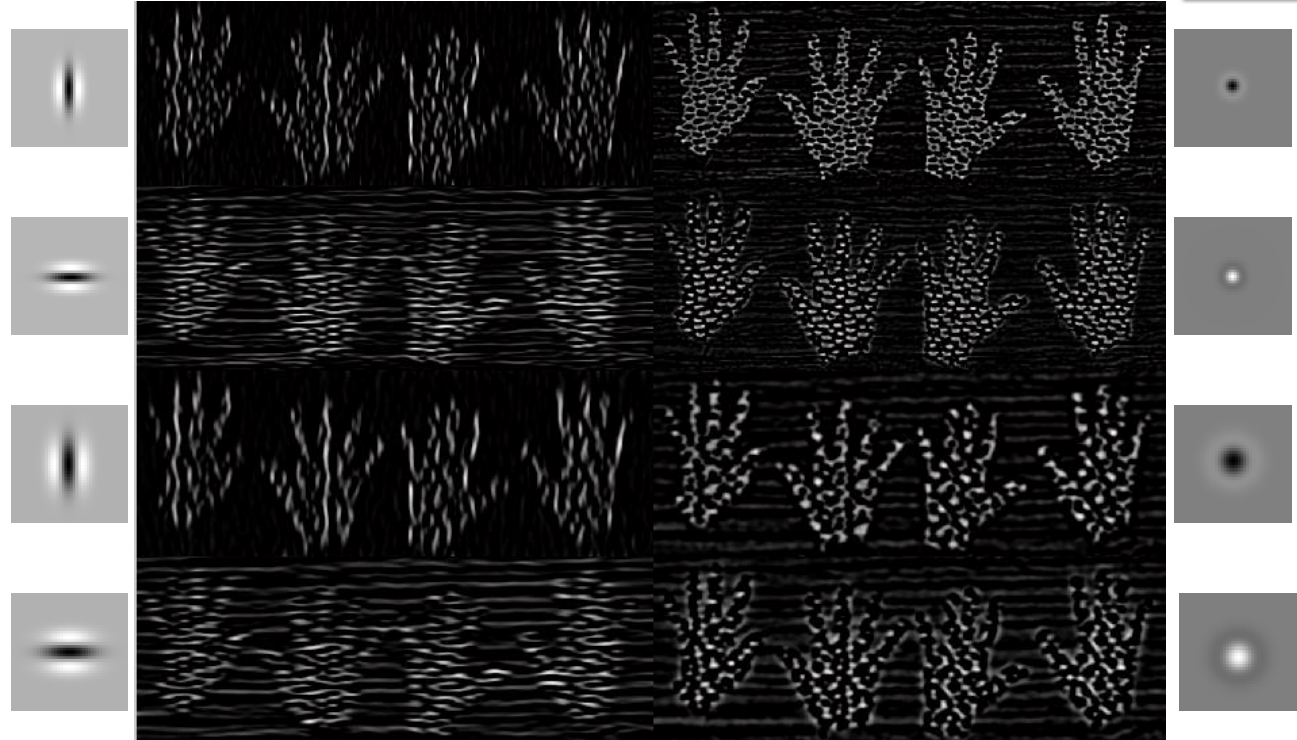
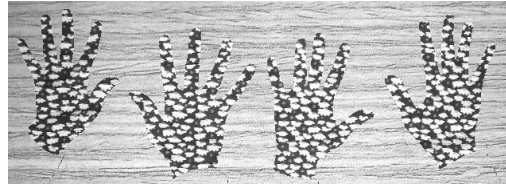
Estos filtros se han generado a partir de **derivadas parciales o combinaciones de éstas** de la función de Gauss en 2D.

Se ha relacionado la forma de estos filtros con los campos receptivos del sistema visual humano, que muestran selectividad a determinados patrones.

Lectura recomendada sobre este tema:  
D. Marr. **Vision**, 1982 Ed. W.H.Freeman and Company.

## Resultados de la convolución:

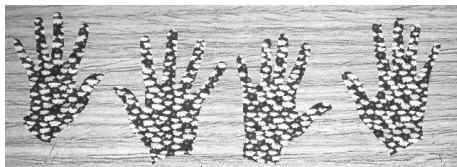
Cada filtro ha detectado unas características diferentes en función de la orientación o la escala.





## Resultado final:

Imagen  
original

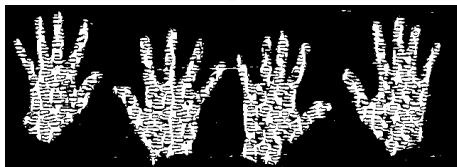


Descriptor  
de c. locales

$$\begin{aligned} \text{Descriptor}(p) &= (x_1 \quad x_2 \quad \dots \quad x_8) = \\ &= \left( x_{s=1}^{90^\circ} \quad x_{s=1}^{45^\circ} \quad x_{s=2}^{90^\circ} \quad x_{s=2}^{45^\circ} \quad x_{s=1}^{iso-} \quad x_{s=1}^{iso+} \quad x_{s=2}^{iso-} \quad x_{s=2}^{iso+} \right) \end{aligned}$$



Resultado  
clasificación



## En resumen:

- Se ha definido el concepto de característica local a partir del concepto de contorno.
- Se ha definido la operación de convolución con un filtro.
- Se han visto algunas cuestiones de implementación de la convolución y se ha relacionado con la correlación .
- Se ha definido un descriptor basado en la concatenación de las respuestas de la convolución con una familia de filtros.

## Resumen de la semana:

- Introducción a una serie de conceptos muy básicos de la visión por computador que creemos que son relevantes para la detección de objetos.
- Tres maneras de construir descriptores de objetos ad-hoc:
  - basados en características del píxel
  - basados en toda la imagen
  - basados en características locales
- Dos algoritmos básicos de la detección de objetos: el template matching y el etiquetaje de regiones conexas.
- Análisis del comportamiento de los algoritmos frente a variaciones de iluminación, de tamaño, o de forma.