

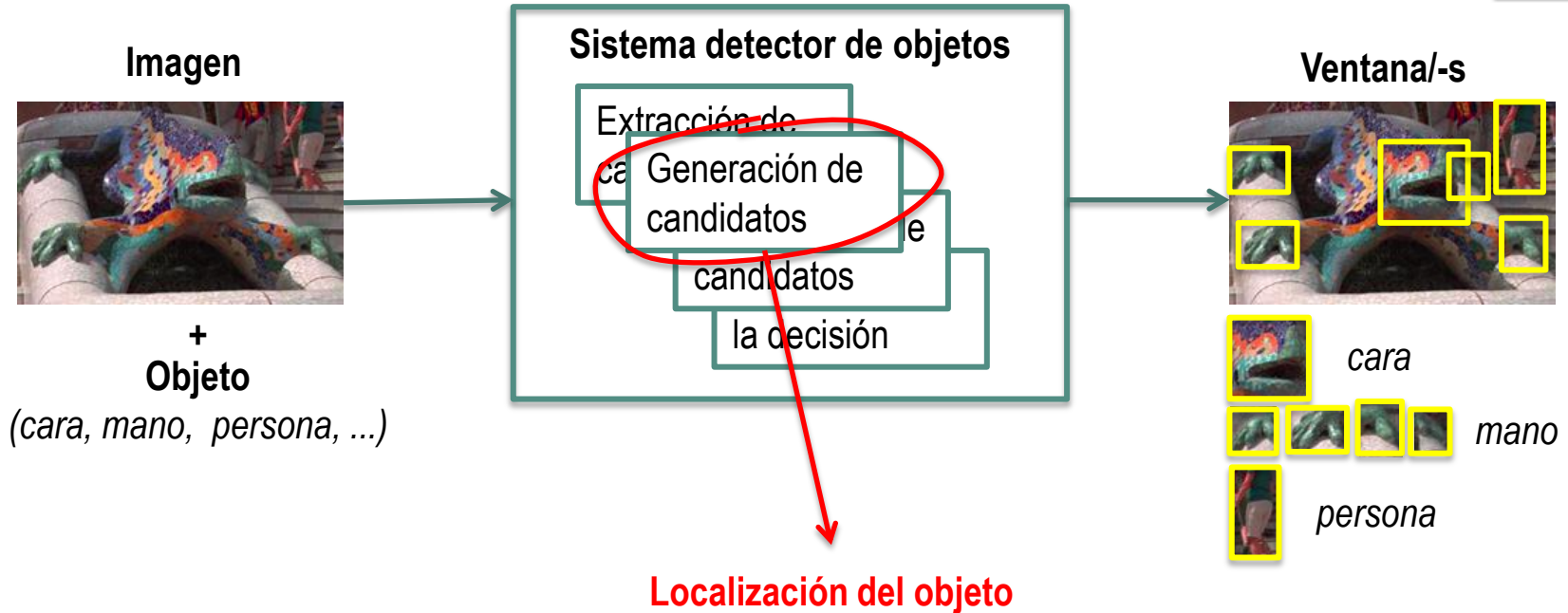
INTRODUCCIÓN A LA DETECCIÓN DE OBJETOS

Componentes conexas

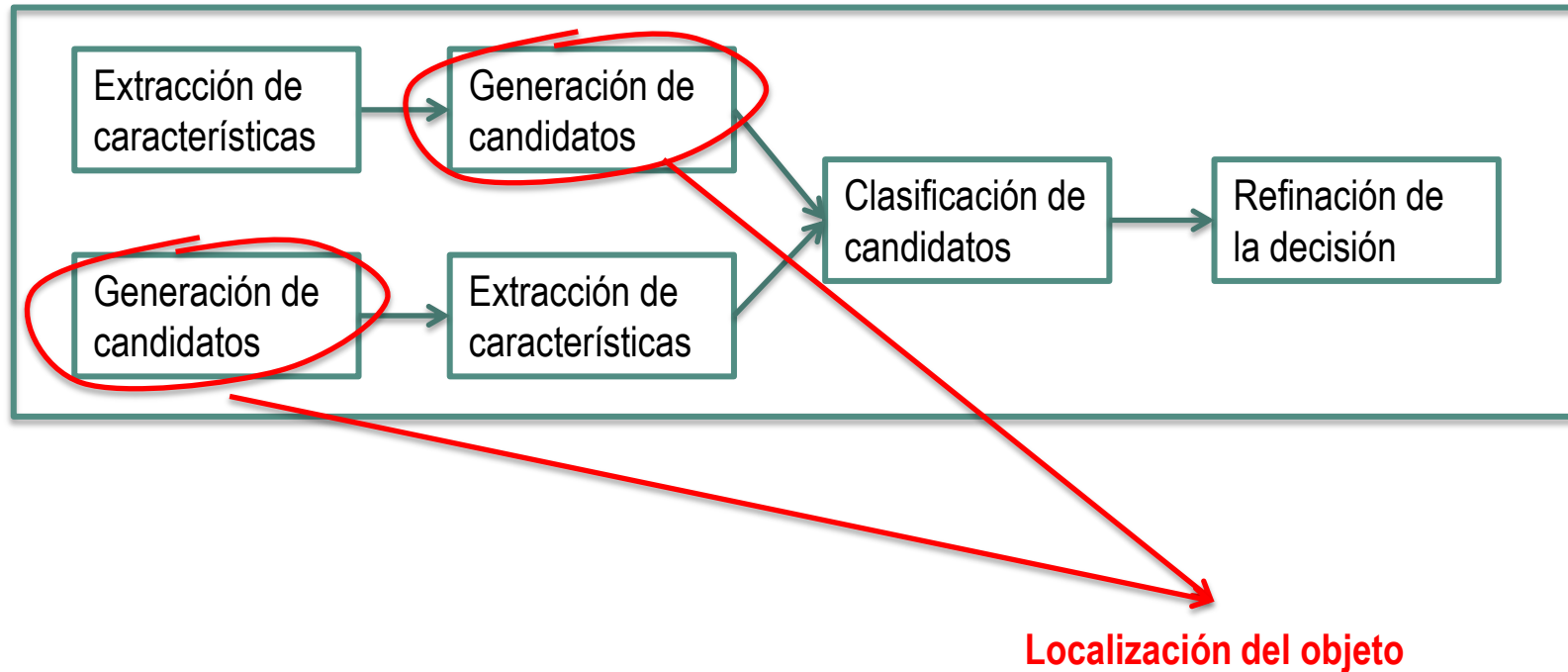
Maria Vanrell

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

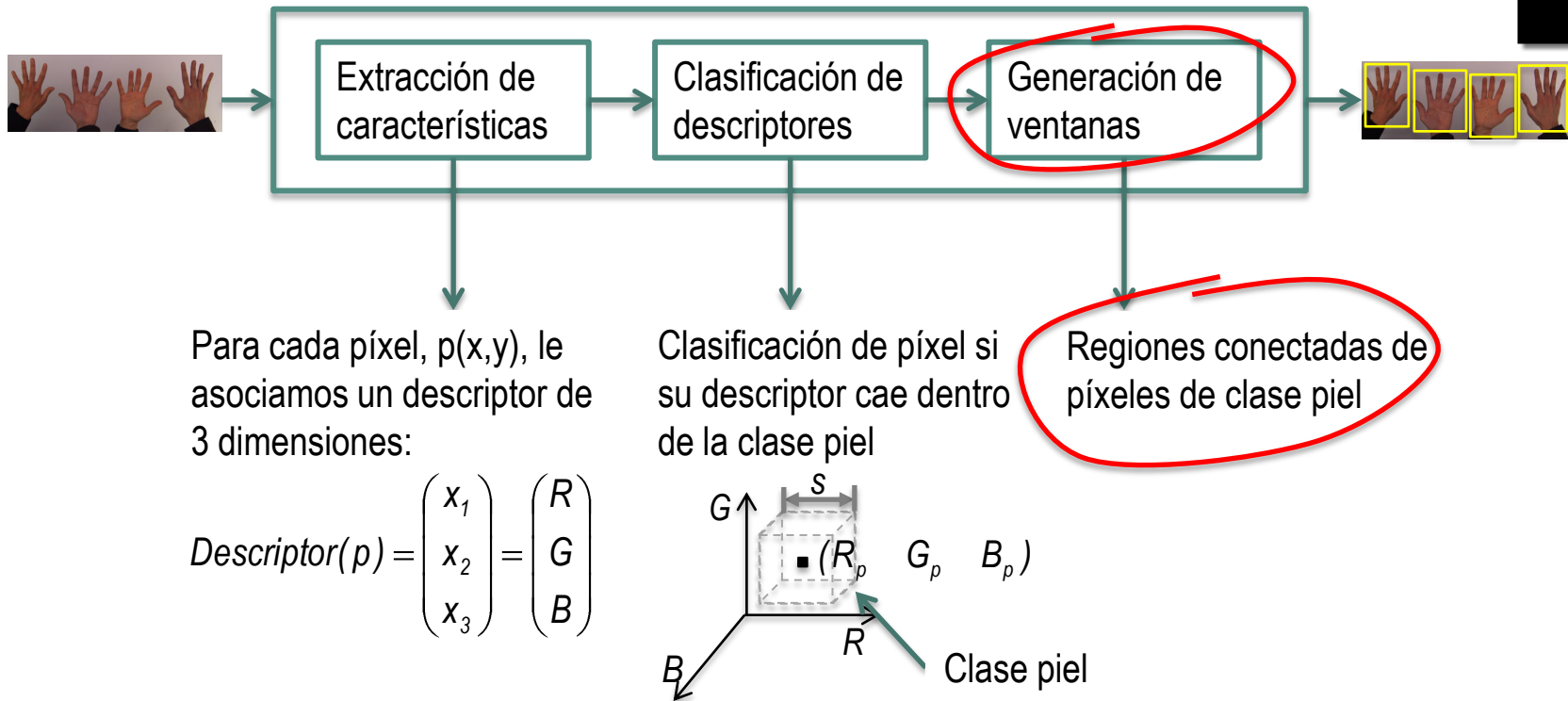
La **generación de candidatos** es el paso que define las regiones de interés en la imagen y ayuda a resolver el problema de la **localización del objeto** dentro de la imagen.



Esquema general de un sistema detector de objetos



Esquema:



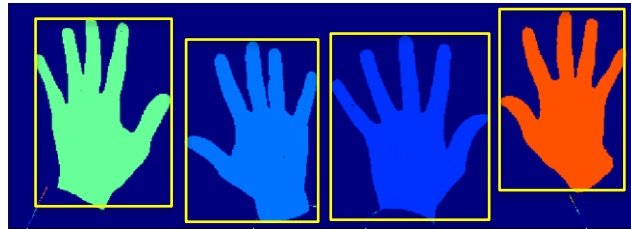


Imagen entrada



Clase piel: $(R_p \ G_p \ B_p) \ s_1 \ s_2 \ s_3$

Blanco: píxeles dentro de la clase piel

Negro: píxeles fuera de la clase piel



Generación de ventanas

Dos pasos:

1. Etiquetaje de regiones conectadas (*Labelling*)
2. Localización de ventanas

La generación de ventanas en dos pasos:

- ✓ 1. Etiquetaje de regiones conectadas (*Labelling*)
2. Localización de ventanas

Extracción de componentes conexas (Etiquetaje/Labelling)

Algoritmo de Etiquetaje: Dada una imagen binaria retorna un imagen con etiquetas numéricas, de manera que todos los píxeles que pertenecen a una misma región conectada compartan la misma etiqueta.

Dos recorridos de la imagen

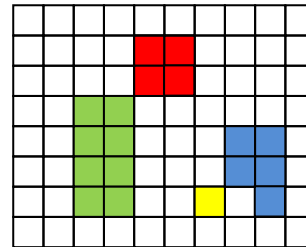
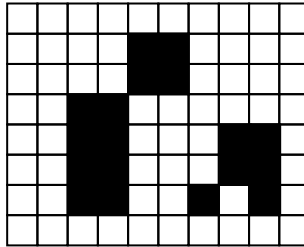
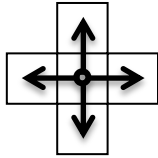
(Un recorrido pasa por cada píxel de la imagen de izquierda a derecha y de arriba abajo)

Paso 1: Etiqueta cada píxel de la imagen atendiendo a sus vecinos superior e izquierdo según la conectividad y guarda posibles equivalencias.

Paso 2: Resuelve todas las equivalencias detectadas de etiquetas y selecciona una etiqueta para cada equivalencia, el segundo recorrido las resuelve y asigna la etiqueta seleccionada.

Concepto preliminar: CONECTIVIDAD, en las imágenes digitales hay de dos tipos

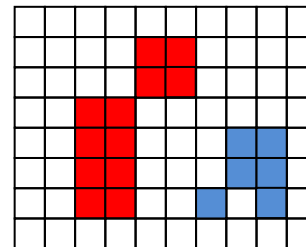
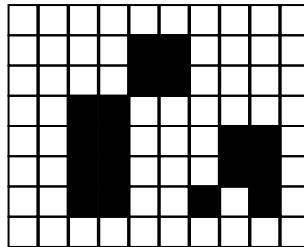
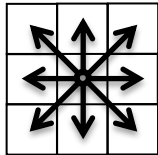
4-Conectividad: Considera la conectividad horizontal y vertical.



4 Etiquetas asignadas



8-Conectividad: Considera la conectividad horizontal, vertical y diagonal.



2 Etiquetas asignadas



Algoritmo de etiquetaje de regiones:

Paso 1: Etiqueta cada píxel de la imagen atendiendo a sus vecinos superior e izquierdo según la conectividad y guarda posibles equivalencias.

Consideramos solamente los píxeles etiquetados a 1:

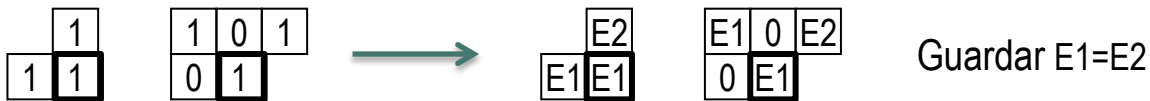
Caso 1: Si (Vecinos=0) entonces Asignar etiqueta NUEVA



Caso 2: Si (Un Vecino \neq 0) entonces Asignar la etiqueta del VECINO



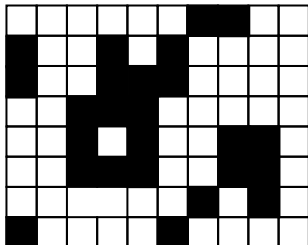
Caso 3: Si (Más de un Vecino \neq 0) entonces Asignar una etiqueta del VECINO e indicar equivalencia



Paso 2: Resuelve todas las equivalencias detectadas de etiquetas y selecciona una etiqueta para cada equivalencia, el segundo recorrido las resuelve y asigna la etiqueta seleccionada.

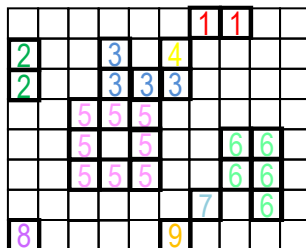
Ejemplo: aplicación del algoritmo

Imagen



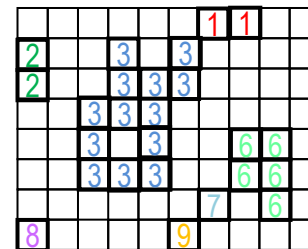
4-conectividad

Paso 1:



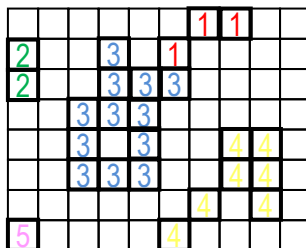
Paso 2:

Equiv: Resolución
 3=4
 3=5 Etiqueta: 3



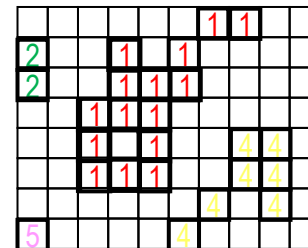
8-conectividad

Paso 1:



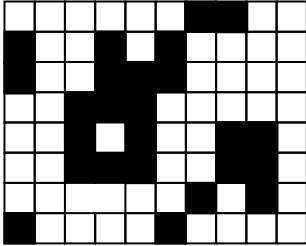
Paso 2:

1=3
 Resolución
 Equiv:
 1=3 Etiqueta: 1

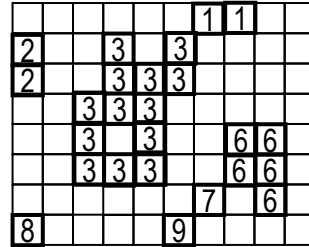


Ejemplo: aplicación del algoritmo

Imagen



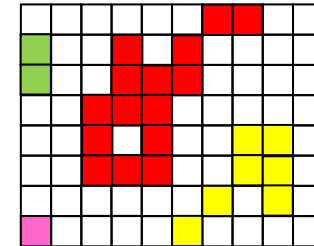
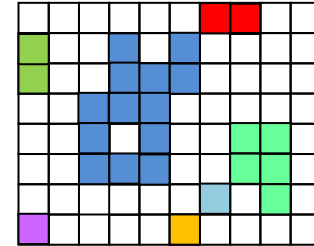
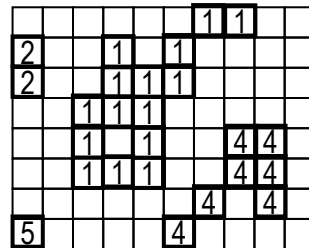
Resultado 4-conectividad:



Etiquetas:

- 1 ■
- 2 ■
- 3 ■
- 4 ■
- 5 ■
- 6 ■
- 7 ■
- 8 ■
- 9 ■

Resultado 8-conectividad:



La generación de ventanas en dos pasos:

- ✓ 1. Etiquetaje de regiones conectadas (*Labelling*)
- ✓ 2. Localización de ventanas

Localización de las ventanas es el paso más simple

Para cada etiqueta: E_k

$$x_1^k = \min_{x_i} \{(x_i, y_j) : \text{etiqueta}(x_i, y_j) = E_k\}$$

$$y_1^k = \min_{y_j} \{(x_i, y_j) : \text{etiqueta}(x_i, y_j) = E_k\}$$

$$x_2^k = \max_{x_i} \{(x_i, y_j) : \text{etiqueta}(x_i, y_j) = E_k\}$$

$$y_2^k = \max_{y_j} \{(x_i, y_j) : \text{etiqueta}(x_i, y_j) = E_k\}$$

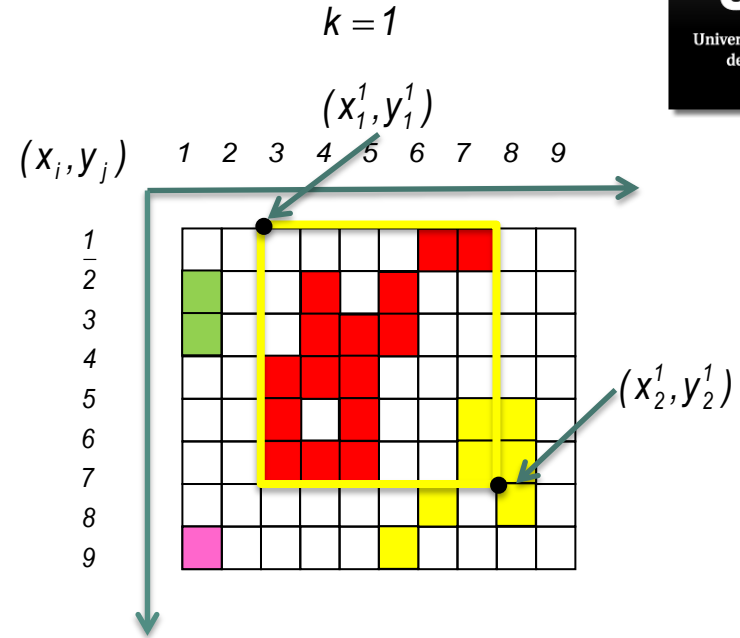


Imagen
entrada



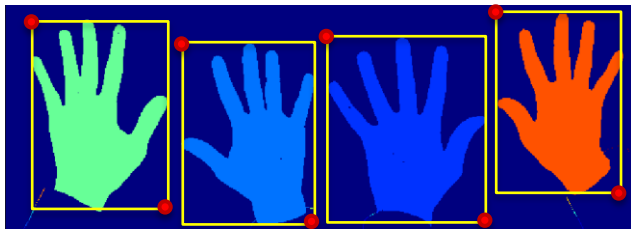
Clasificación



Etiquetaje



Localización



Etiquetas:



En resumen:

- Se ha tratado el problema de la generación de ventanas candidatas a contener objetos, partiendo del resultado obtenido por un clasificador a nivel de píxel.
- Se ha definido el problema del etiquetaje de componentes conexas.
- Se ha visto un algoritmo de etiquetaje de regiones conexas.
- Se ha calculado la ventana que contiene la componente conexa.