



Visión Artificial

Tema 4: Segmentación de imágenes

- ▼ Técnicas basadas en regiones
- ▼ Técnicas basadas en agrupaciones
- ▼ Segmentación semántica

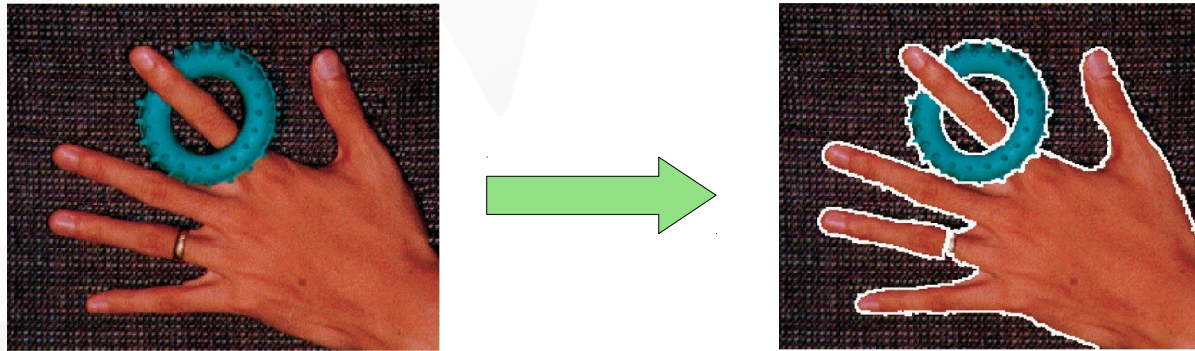
Segmentación de imágenes

- ▶ Una vez que la imagen ha sido procesada adecuadamente, la segmentación constituye un paso fundamental en la interpretación de la imagen.
- ▶ La segmentación de la imagen es un proceso en el que los píxels que comparten características similares se agrupan formando partes independientes.

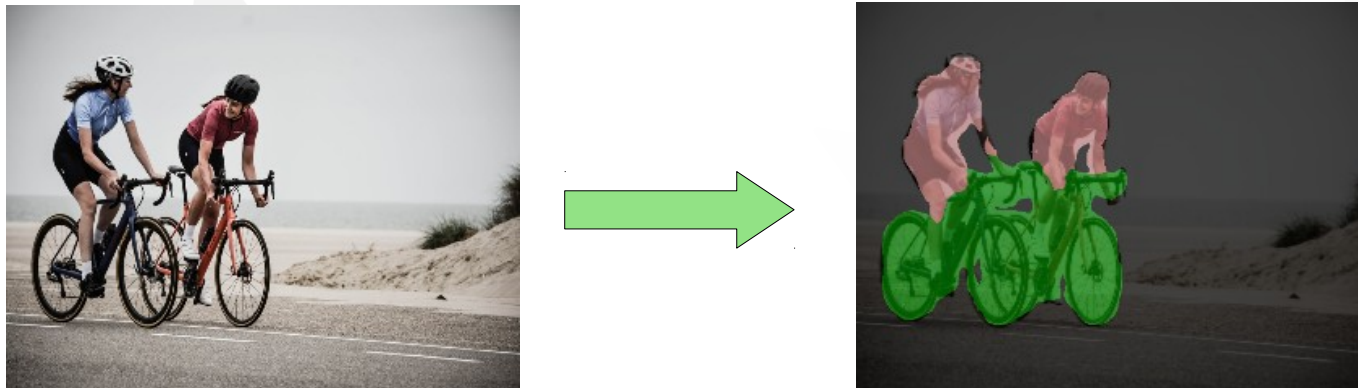


Segmentación de imágenes

- ▼ **Segmentación *bottom-up*:** etiquetar cada píxel en base a sus propiedades (técnicas de clasificación estadística, umbralización, detección de bordes, detección de regiones o una combinación de ellas).



- ▼ **Segmentación semántica:** etiquetar cada píxel de acuerdo al objeto al que pertenece.



Segmentación de imágenes

Segmentación bottom-up

- ▼ Técnicas basadas en regiones:
 - ▼ La segmentación se aplica sobre el espacio de imagen.
 - ▼ Crecimiento de regiones.
- ▼ Técnicas basadas en agrupaciones:
 - ▼ La segmentación se aplica sobre el espacio de características.
 - ▼ K-medias, *mean-shift*.

Técnicas basadas en regiones

- ▼ Crecimiento de regiones
- ▼ Uso de bordes en segmentación

Técnicas basadas en regiones

- ▼ Los métodos de segmentación basada en regiones tratan de particionar la imagen en regiones cuyos elementos presenten propiedades comunes. Estas propiedades pueden consistir en:
 - ▼ Valores de intensidad de la imagen original o valores calculados a partir de un cierto operador.
 - ▼ Patrones o texturas únicas en cada tipo de región.
 - ▼ Perfiles espectrales que proporcionan datos de imágenes multidimensionales.
- ▼ Las técnicas más elaboradas de segmentación utilizan una combinación de estas propiedades, mientras que los métodos más simples están restringidos a un conjunto mínimo de propiedades en función de los datos disponibles.

Crecimiento de regiones

- ▼ El crecimiento de regiones es una de las técnicas de segmentación bottom-up más populares:
 - ▼ Consiste en fusionar, de manera iterativa, pequeñas áreas (píxels), inicialmente independientes, de acuerdo con algún criterio de similitud.
 - ▼ Se parte de un píxel inicial, conocido como punto semilla, que constituye un punto representativo de una región. Este píxel se compara con sus vecinos.
 - ▼ La región crece a partir del punto semilla añadiendo los puntos conectados a él con características similares.
 - ▼ Cuando una región no puede seguir creciendo, se selecciona otro punto semilla y se repite el proceso para la nueva región.
 - ▼ El proceso continúa hasta que no existan más puntos semilla.

Crecimiento de regiones

Punto de partida: semillas
Distancia máxima entre características:2

7	8	8	9	7	7	60	61	61	62
9	9	7	7	8	9	60	60	61	62
9	9	8	8	9	8	61	60	61	62
8	8	7	7	7	8	61	61	61	61
9	7	8	8	9	60	61	61	60	60
9	7	8	8	62	61	62	62	61	60
8	8	9	60	61	61	62	61	60	61
7	8	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento: análisis de los vecinos
Conectividad 4

7	8	8	9	7	7	60	61	61	62
9	9	7	7	8	9	60	60	61	62
9	9	8	8	9	8	61	60	61	62
8	8	7	7	7	8	61	61	61	61
9	7	8	8	9	60	61	61	60	60
9	7	8	8	62	61	62	62	61	60
8	8	9	60	61	61	62	61	60	61
7	8	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento de regiones

Crecimiento: marcar los vecinos similares como pertenecientes a la misma región

7	8	8	9	7	7	60	61	61	62
9	9	7	7	8	9	60	60	61	62
9	9	8	7	9	8	61	60	61	62
8	8	7	7	7	8	61	61	61	61
9	7	9	7	9	60	61	61	60	60
9	7	8	8	62	61	62	62	61	60
8	8	9	60	61	61	62	61	60	61
7	8	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento: repetir el proceso para los vecinos que cumplan el criterio de similitud

7	8	8	9	7	7	60	61	61	62
9	9	7	7	8	9	60	60	61	62
9	9	8	7	9	8	61	60	61	62
8	8	7	7	7	8	61	61	61	61
9	7	9	7	9	60	61	61	60	60
9	7	8	8	62	61	62	62	61	60
8	8	9	60	61	61	62	61	60	61
7	8	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento de regiones

Crecimiento: parar cuando no se puedan marcar más píxeles

7	7	7	7	7	7	60	61	61	62
7	7	7	7	7	7	60	60	61	62
7	7	7	7	7	7	61	60	61	62
7	7	7	7	7	7	61	61	61	61
7	7	7	7	7	60	61	61	60	60
7	7	7	7	62	61	62	62	61	60
7	7	7	60	61	61	62	61	60	61
7	7	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento: repetir el proceso para la siguiente semilla

7	7	7	7	7	7	62	62	62	62
7	7	7	7	7	7	62	62	62	62
7	7	7	7	7	7	62	62	62	62
7	7	7	7	7	7	62	62	62	62
7	7	7	7	7	62	62	62	62	62
7	7	7	7	62	62	62	62	62	62
7	7	7	62	62	62	62	62	62	62
7	7	62	62	62	62	62	62	62	62
62	62	62	62	62	62	62	62	62	62
62	62	62	62	62	62	62	62	62	62

Crecimiento de regiones

- ▼ Uso de rango flotante en la comparación:
 - ▼ En el ejemplo anterior, un píxel se añade a la región si sus atributos son similares a los del punto semilla.
 - ▼ En ese caso se habla de comparación de **rango fijo**.
 - ▼ Otra posibilidad es utilizar una comparación de **rango flotante**.
 - ▼ En la comparación de **rango flotante**, cada píxel se compara con su vecino de referencia.
 - ▼ Proporciona mejores resultados cuando la variación de intensidad de una región es gradual.

Crecimiento de regiones

Crecimiento con rango fijo
Distancia máxima entre características: 2

10	11	10	10	10	12	60	61	61	62
10	9	8	8	8	10	60	60	61	62
11	9	8	8	9	10	61	60	61	62
10	8	7	7	8	10	61	61	61	61
10	8	8	8	10	60	61	61	60	60
10	8	8	10	62	61	62	62	61	60
10	9	10	60	61	61	62	61	60	61
11	10	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Resultado de la segmentación
a partir de la semilla

10	11	10	10	10	12	60	61	61	62
10	7	7	7	7	10	60	60	61	62
11	7	7	7	7	10	61	60	61	62
10	7	7	7	7	10	61	61	61	61
10	7	7	7	10	60	61	61	60	60
10	7	7	10	62	61	62	62	61	60
10	7	10	60	61	61	62	61	60	61
11	10	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

- Los puntos del contorno de la región no se clasifican correctamente.
- Si se aumenta la distancia máxima entre característica, los puntos se clasificarían correctamente, pero podrían incluirse algunos erróneamente.

Crecimiento de regiones

Crecimiento con rango flotante
Distancia máxima entre características: 2

10	11	10	10	10	12	60	61	61	62
10	9	8	8	8	10	60	60	61	62
11	9	8	8	9	10	61	60	61	62
10	8	7	7	8	10	61	61	61	61
10	8	8	8	10	60	61	61	60	60
10	8	8	10	62	61	62	62	61	60
10	9	10	60	61	61	62	61	60	61
11	10	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Resultado de la segmentación
a partir de la semilla

7	7	7	7	7	7	60	61	61	62
7	7	7	7	7	7	60	60	61	62
7	7	7	7	7	7	61	60	61	62
7	7	7	7	7	7	61	61	61	61
7	7	7	7	7	60	61	61	60	60
7	7	7	7	62	61	62	62	61	60
7	7	7	60	61	61	62	61	60	61
7	7	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento de regiones

- ▼ Crecimiento de regiones sin puntos semilla predefinidos:
 - ▼ Comenzar por una determinada posición de la imagen (esquina superior izquierda): considerar dicha posición como el primer punto de una nueva región
 - ▼ Hacer crecer la región a través de los vecinos de cada nuevo punto añadido.
 - ▼ Cuando una región no puede crecer más, se repite el proceso a partir del primer punto de la imagen que no pertenezca a ninguna región.

Crecimiento de regiones

Semilla: primer punto de imagen
Distancia máxima entre características:2

7	8	8	9	7	7	60	61	61	62
9	9	7	7	8	9	60	60	61	62
9	9	8	8	9	8	61	60	61	62
8	8	7	7	7	8	61	61	61	61
9	7	8	8	9	60	61	61	60	60
9	7	8	8	62	61	62	62	61	60
8	8	9	60	61	61	62	61	60	61
7	8	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento con conectividad 4

7	7	7	7	7	7	60	61	61	62
7	7	7	7	7	7	60	60	61	62
7	7	7	7	7	7	61	60	61	62
7	7	7	7	7	7	61	61	61	61
7	7	7	7	7	60	61	61	60	60
7	7	7	7	62	61	62	62	61	60
7	7	7	60	61	61	62	61	60	61
7	7	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento de regiones

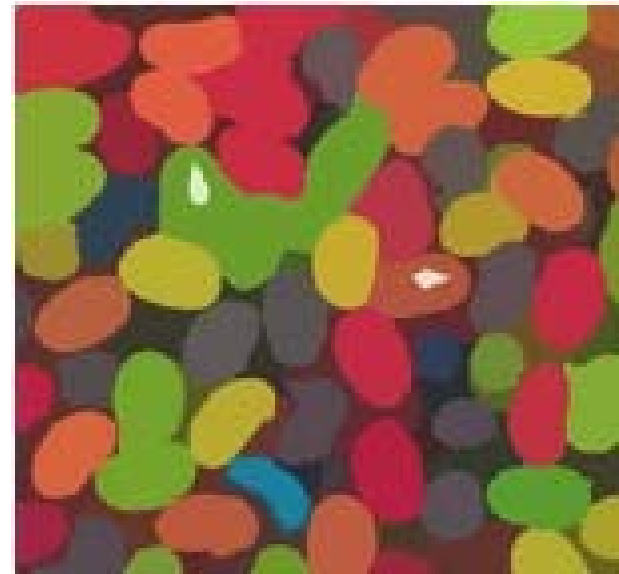
Siguiente semilla: primer punto sin clasificar

7	7	7	7	7	7	60	61	61	62
7	7	7	7	7	7	60	60	61	62
7	7	7	7	7	7	61	60	61	62
7	7	7	7	7	7	61	61	61	61
7	7	7	7	7	60	61	61	60	60
7	7	7	7	62	61	62	62	61	60
7	7	7	60	61	61	62	61	60	61
7	7	61	61	61	62	62	61	61	61
60	62	62	62	61	61	61	62	62	62
60	60	60	61	61	61	60	61	61	62

Crecimiento con conectividad 4

7	7	7	7	7	7	60	60	60	60
7	7	7	7	7	7	60	60	60	60
7	7	7	7	7	7	60	60	60	60
7	7	7	7	7	7	60	60	60	60
7	7	7	7	7	60	60	60	60	60
7	7	7	7	60	60	60	60	60	60
7	7	7	60	60	60	60	60	60	60
7	7	60	60	60	60	60	60	60	60
60	60	60	60	60	60	60	60	60	60
60	60	60	60	60	60	60	60	60	60

Crecimiento de regiones



Uso de bordes en segmentación

- ▼ La información de los bordes presentes en la imagen puede ser de gran utilidad en segmentación.
- ▼ Los bordes representan, en muchas ocasiones, cambios de intensidad entre regiones diferentes: **límites de regiones**
- ▼ El uso exclusivo de bordes en segmentación no proporciona buenos resultados:
 - ▼ El ruido puede detectarse como borde.
 - ▼ Es posible que la variación de intensidad entre regiones contiguas no sea lo suficientemente alta: *ausencia de borde*
- ▼ Las técnicas basadas en regiones tienden a cometer errores en los límites entre regiones.
- ▼ La información de bordes puede reducir los problemas asociados con el crecimiento de regiones.

Uso de bordes en segmentación

- ▼ La información de bordes puede integrarse en el método de crecimiento de regiones de varias maneras.
- ▼ Dos tipos de integración:
 - ▼ Integración *en línea*: la información de bordes se utiliza durante el proceso de crecimiento.
 - ▼ Criterio de control adicional.
 - ▼ Selección de puntos semilla.
 - ▼ Integración posterior: combina la imagen de regiones y la de bordes con el objetivo de obtener una segmentación más precisa.

Uso de bordes en segmentación

- ▼ Integración en línea: criterio de control adicional
 - ▼ Seleccionar un píxel que no pertenezca a ninguna región
 - ▼ **Criterio adicional: el píxel seleccionado no puede pertenecer a un borde**
 - ▼ Añadir los píxels vecinos que sean similares en base a un criterio de similitud
 - ▼ Continuar el crecimiento de la región a partir de los vecinos similares:
 - ▼ **Criterio adicional: si un píxel pertenece a un borde, el crecimiento de la región no continuará a partir de ese píxel.**

Uso de bordes en segmentación



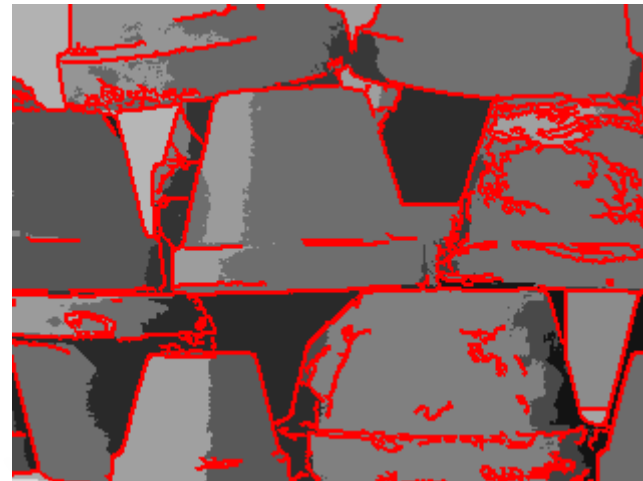
Uso de bordes en segmentación

- ▼ **Integración posterior:** reducir la sobresegmentación
 - ▼ Sobresegmentación: imagen excesivamente particionada.
 - ▼ La información de bordes puede ayudar a reducir este efecto.

Resultado de la segmentación



Bordes superpuestos (en rojo)



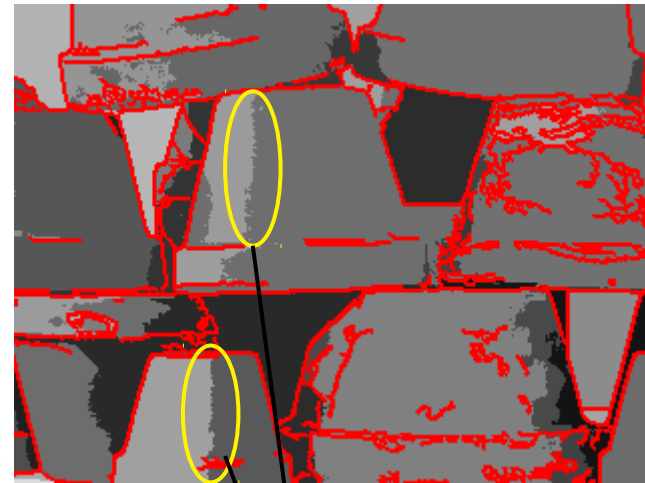
Uso de bordes en segmentación

- ▼ **Integración posterior:** reducir la sobresegmentación
 - ▼ Sobresegmentación: imagen excesivamente particionada.
 - ▼ La información de bordes puede ayudar a reducir este efecto.

Resultado de la segmentación



Bordes superpuestos (en rojo)



Frontera entre regiones compuesta por una mayoría de puntos no pertenecientes a un borde

Uso de bordes en segmentación

▼ Integración posterior: unión de regiones

- ▼ **Criterio de unión:** si la frontera de dos regiones adyacentes no se corresponde con puntos de borde (en un porcentaje elevado), dichas regiones pueden fusionarse. Deben considerarse regiones adyacentes por un número significativo de puntos frontera.

▼ Proceso:

- ▼ Por cada región, comprobar el criterio de unión con cada una de sus regiones adyacentes.
- ▼ Si dicho criterio se cumple, etiquetar los píxels de la región vecina como pertenecientes a la región actual.

Uso de bordes en segmentación

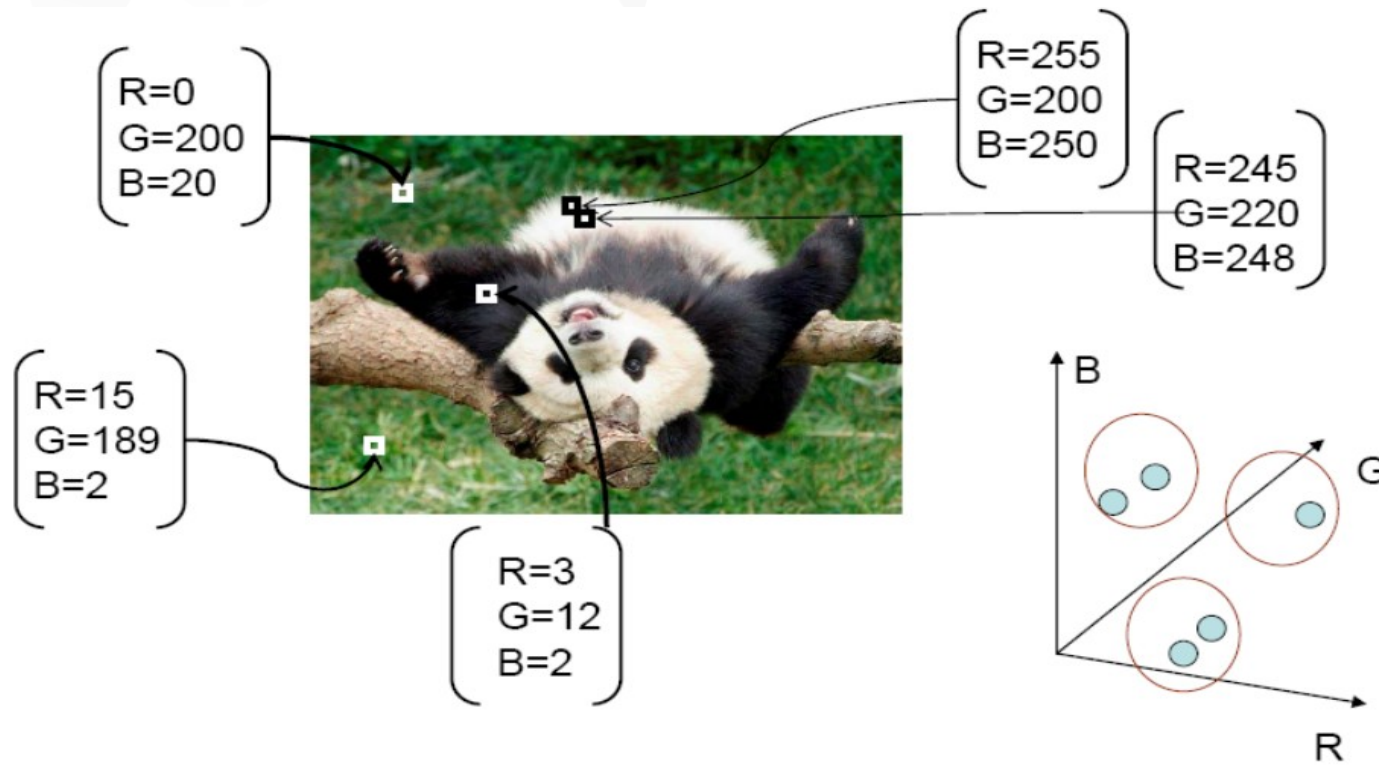


Técnicas basadas en agrupaciones

- ▼ K-medias
- ▼ Desplazamiento de la media (*mean-shift*)

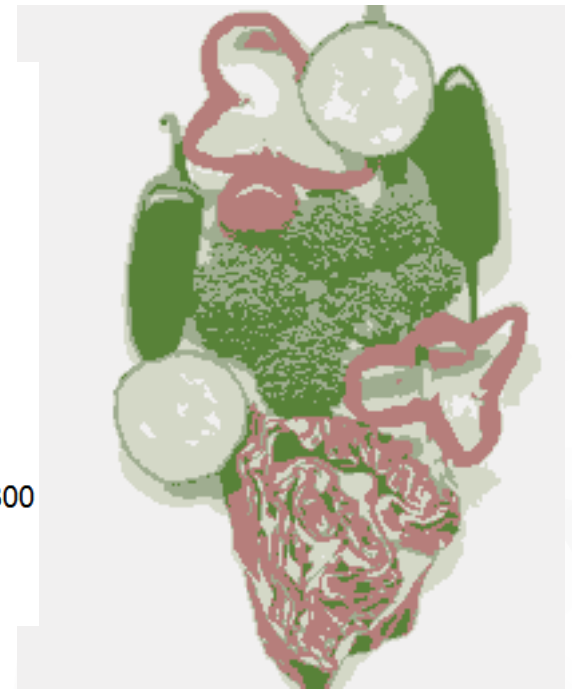
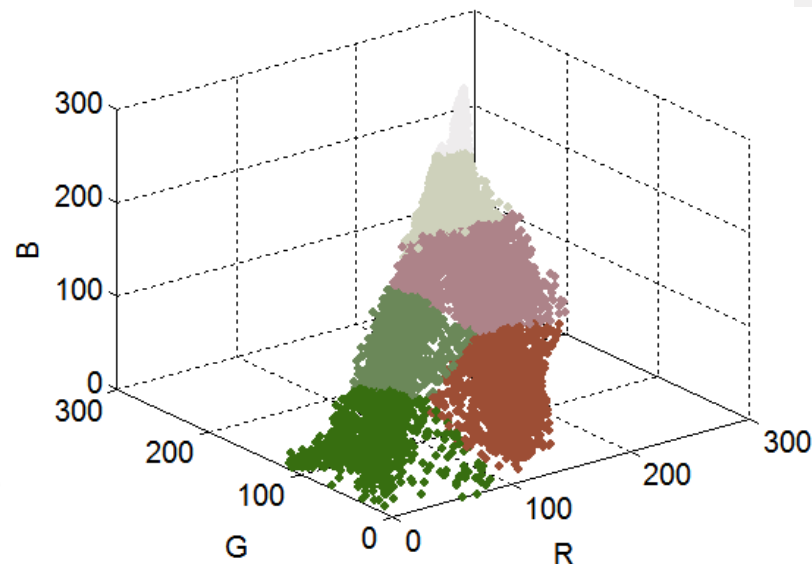
Técnicas basadas en agrupaciones

- ▼ Cada píxel es un punto en el espacio de características
- ▼ El objetivo es formar agrupaciones de puntos próximos en el espacio de características



Técnicas basadas en agrupaciones

- ▼ Características:
 - ▼ Intensidad (nivel de gris)
 - ▼ Color (RGB)

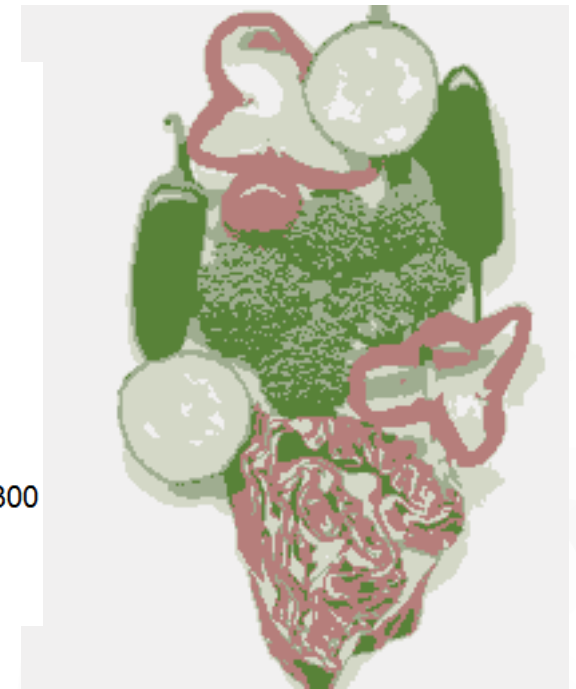
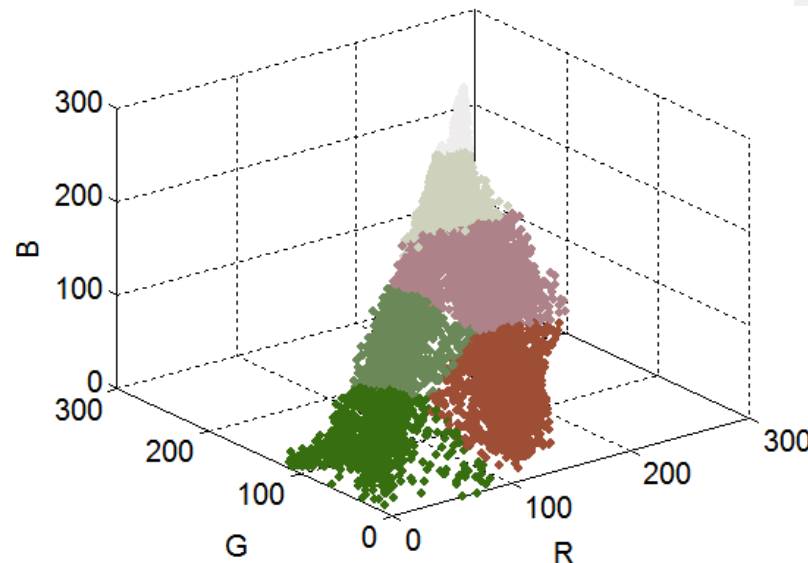


Técnicas basadas en agrupaciones

▼ Características:

- ▼ Intensidad (nivel de gris)
- ▼ Color (RGB)

Las agrupaciones no tienen coherencia espacial



Técnicas basadas en agrupaciones

▼ Características:

- ▼ Intensidad/color y posición (r, g, b, x, y)

Imagen original



Segmentación por color



Segmentación por color y posición



K-medias

- ▼ Suposición inicial: el número de agrupaciones (regiones) es conocido a priori.
- ▼ Cada agrupación (C_i) se caracteriza por su posición central (media de los puntos de la agrupación - m_i).

- ▼ Algoritmo:

- ▼ **Asignación**: asociar cada punto con la agrupación más próxima

$$C_i^{(t)} = \left\{ x_j : |x_j - m_i^{(t)}| \leq |x_j - m_l^{(t)}| \forall l = 1, \dots, k \right\}$$

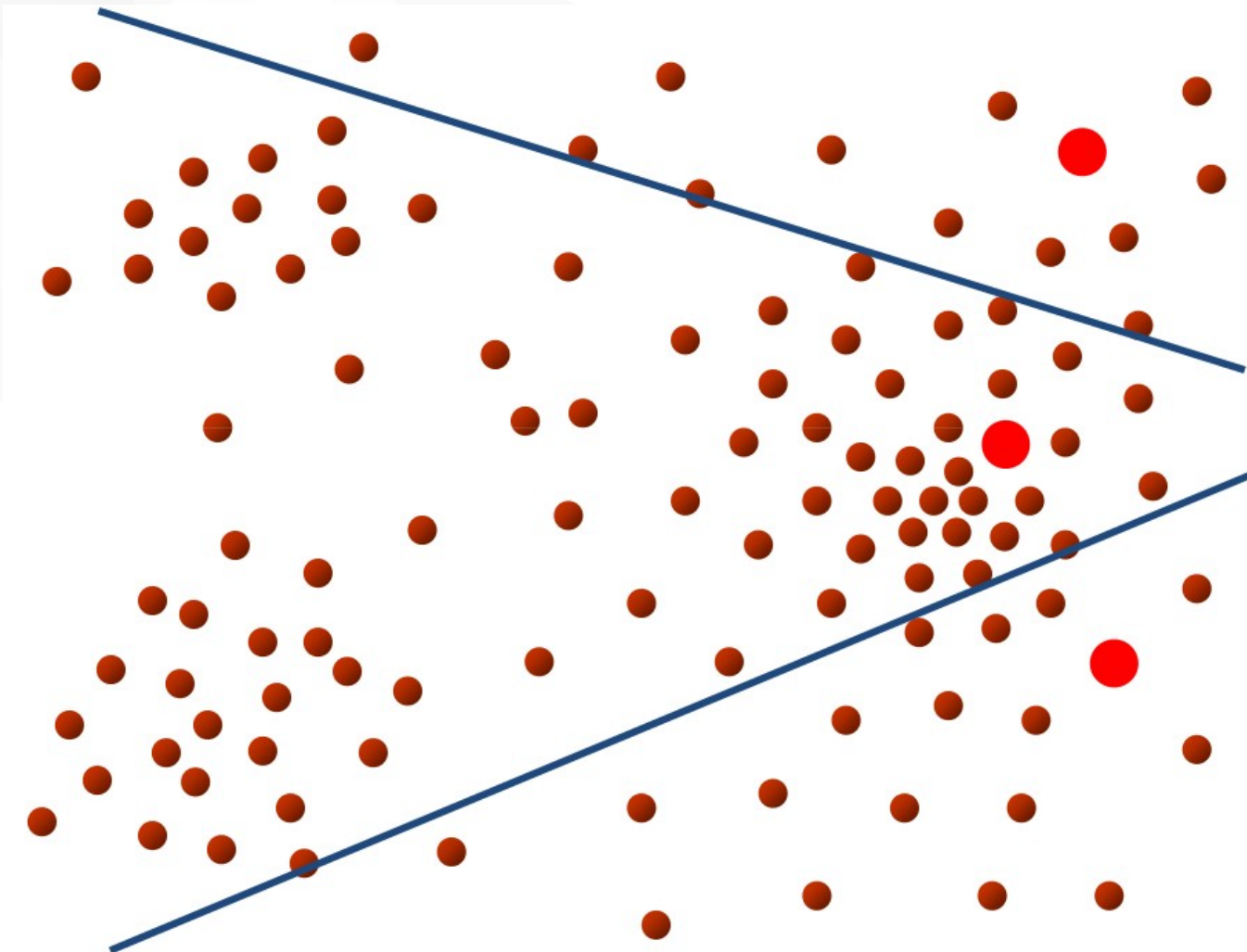
- ▼ **Actualización**: calcular los nuevos centros (medias) de cada agrupación

$$m_i^{(t+1)} = \frac{\sum_{x_j \in C_i^{(t)}} x_j}{|C_i^{(t)}|}$$

- ▼ Iterar hasta que las k-medias converjan: los centros no varíen

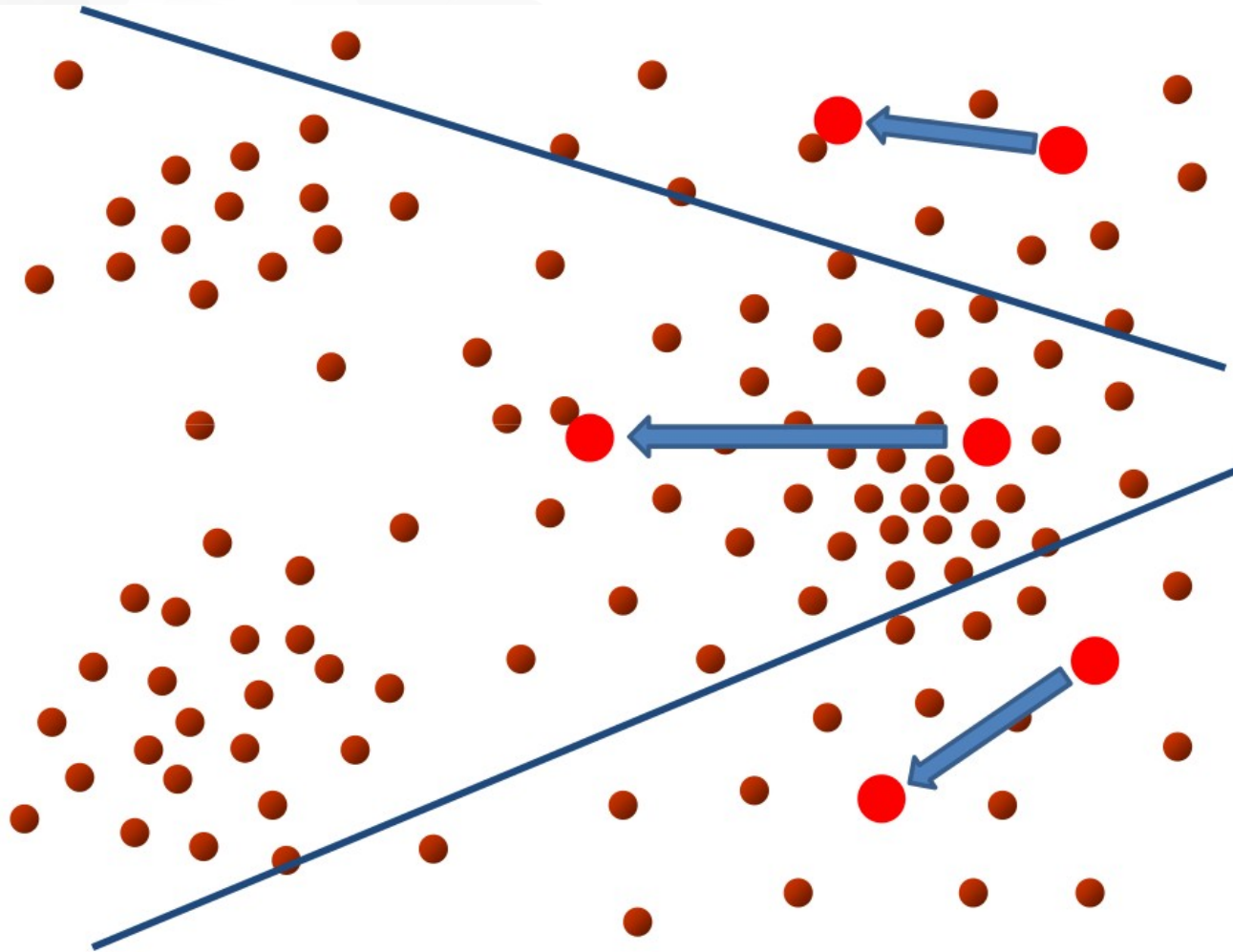
K-medias

Asignación de puntos a cada agrupación



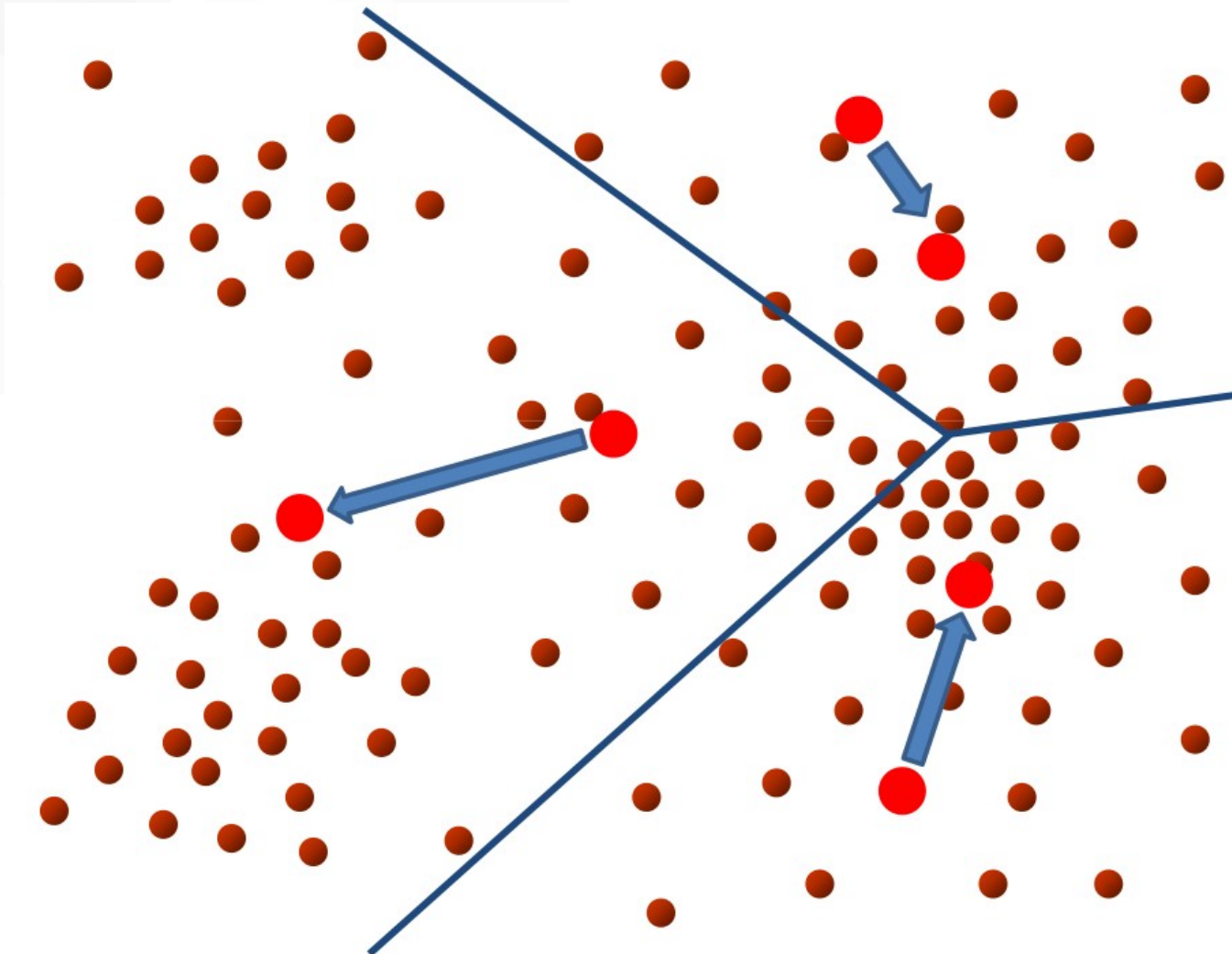
K-medias

Cálculo de la nueva media de cada agrupación



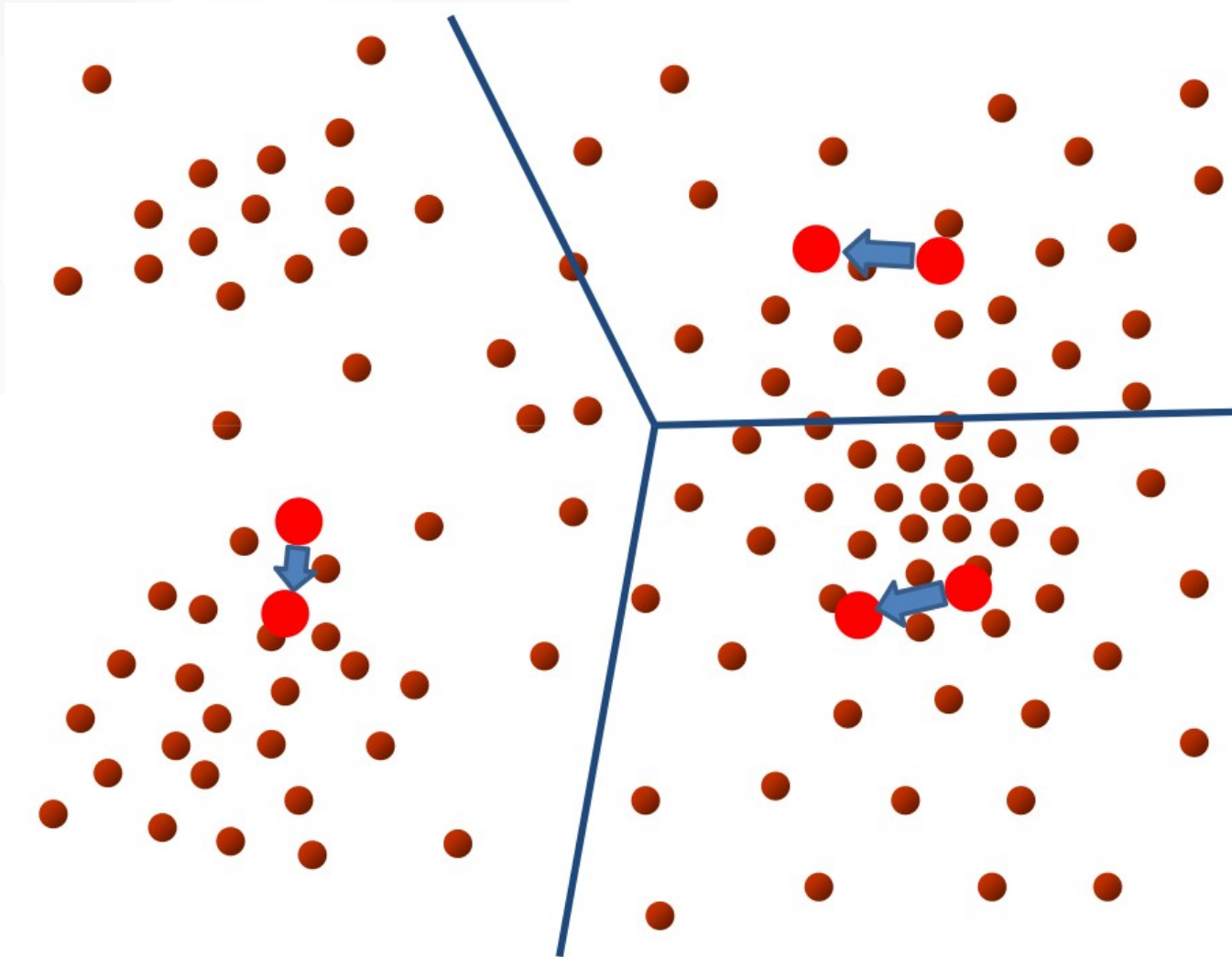
K-medias

Asignación de puntos y actualización de la media



K-medias

Asignación de puntos y actualización de la media



K-medias

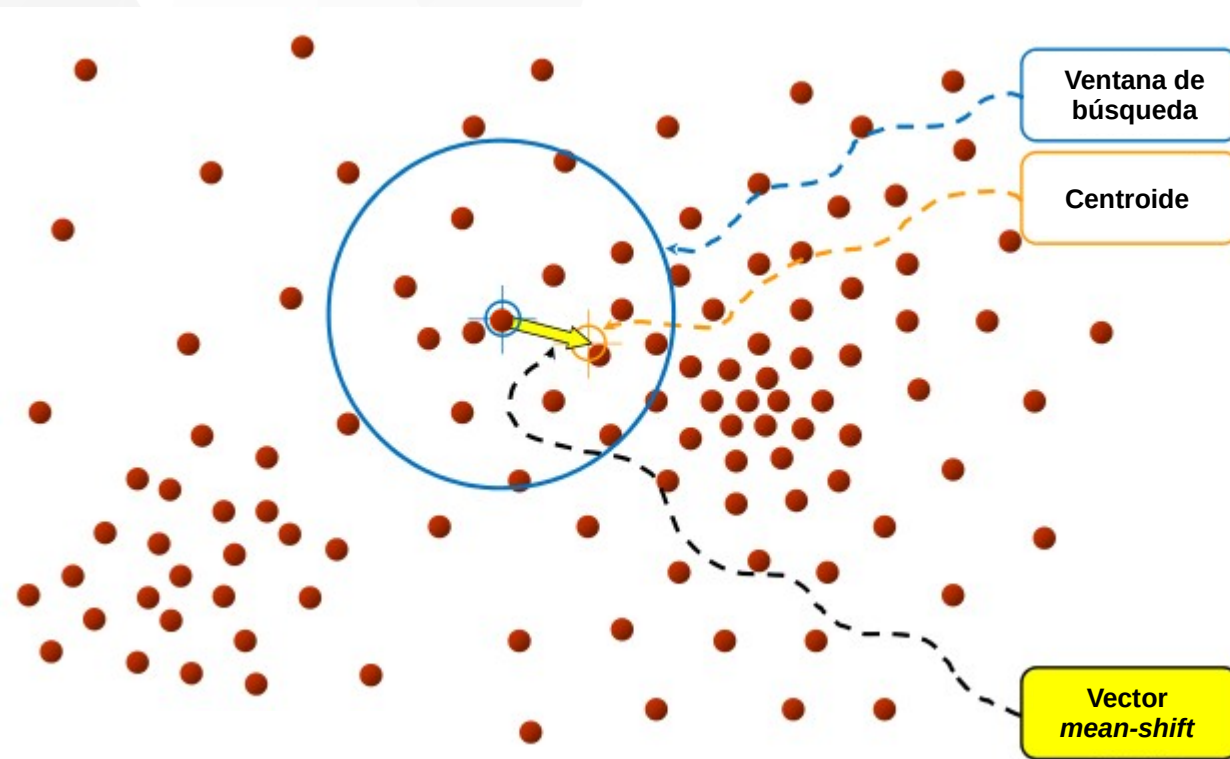
- ▼ Ventajas:
 - ▼ Método muy simple.
 - ▼ Converge a un mínimo local de la función de error.
- ▼ Inconvenientes
 - ▼ Es necesario establecer el número de agrupaciones.
 - ▼ Es sensible a la inicialización.
 - ▼ Es sensible a los *outliers* (puntos discrepantes).
 - ▼ Sólo estima agrupaciones esféricas.

Desplazamiento de la media (*mean-shift*)

- ▼ El método ***mean-shift*** busca los máximos locales de densidad (modas) del espacio de características.
- ▼ Idea principal: asociar cada punto con la zona de mayor densidad de puntos del espacio de características más próxima a él.
- ▼ ***Mean-shift* sobre un punto**:
 - ▼ Seleccionar una ventana de búsqueda (escala o ancho de banda) centrada en el punto.
 - ▼ Calcular la media de los puntos contenidos en dicha ventana (centroide).
 - ▼ Centrar la ventana de búsqueda en la nueva posición.
 - ▼ Repetir hasta alcanzar la convergencia (la media no se desplaza).

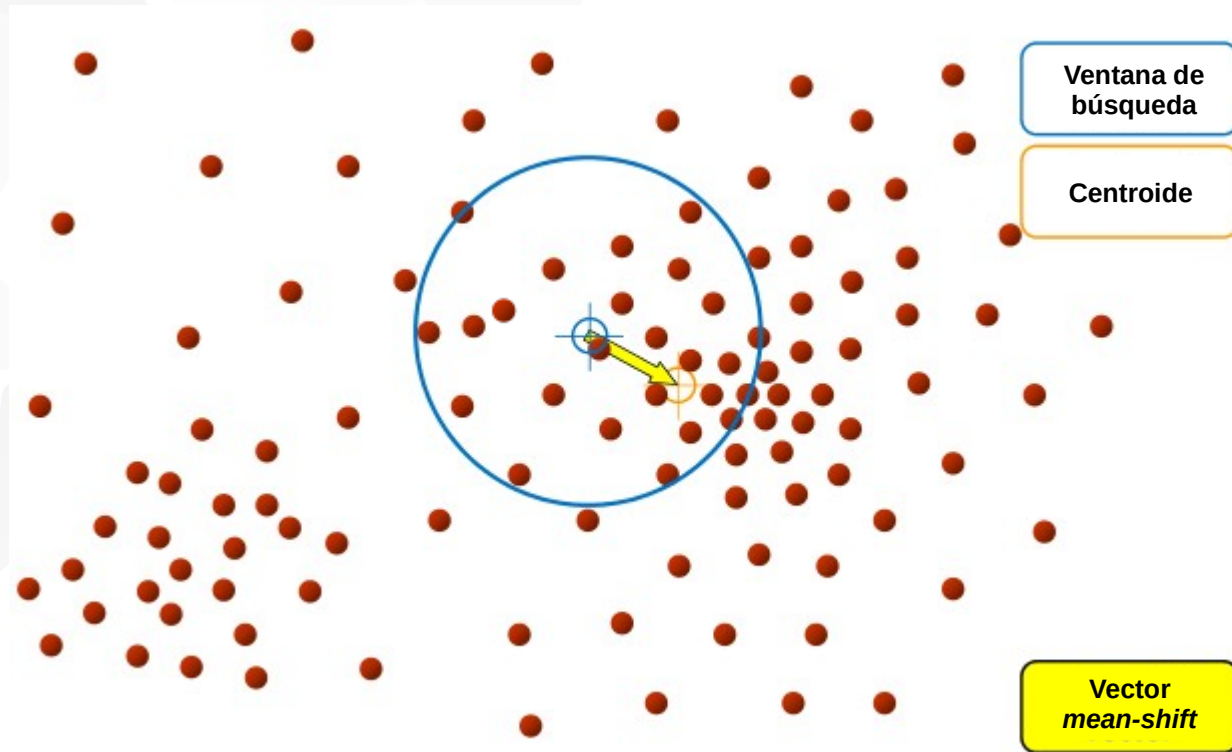
Desplazamiento de la media (*mean-shift*)

Mean-shift



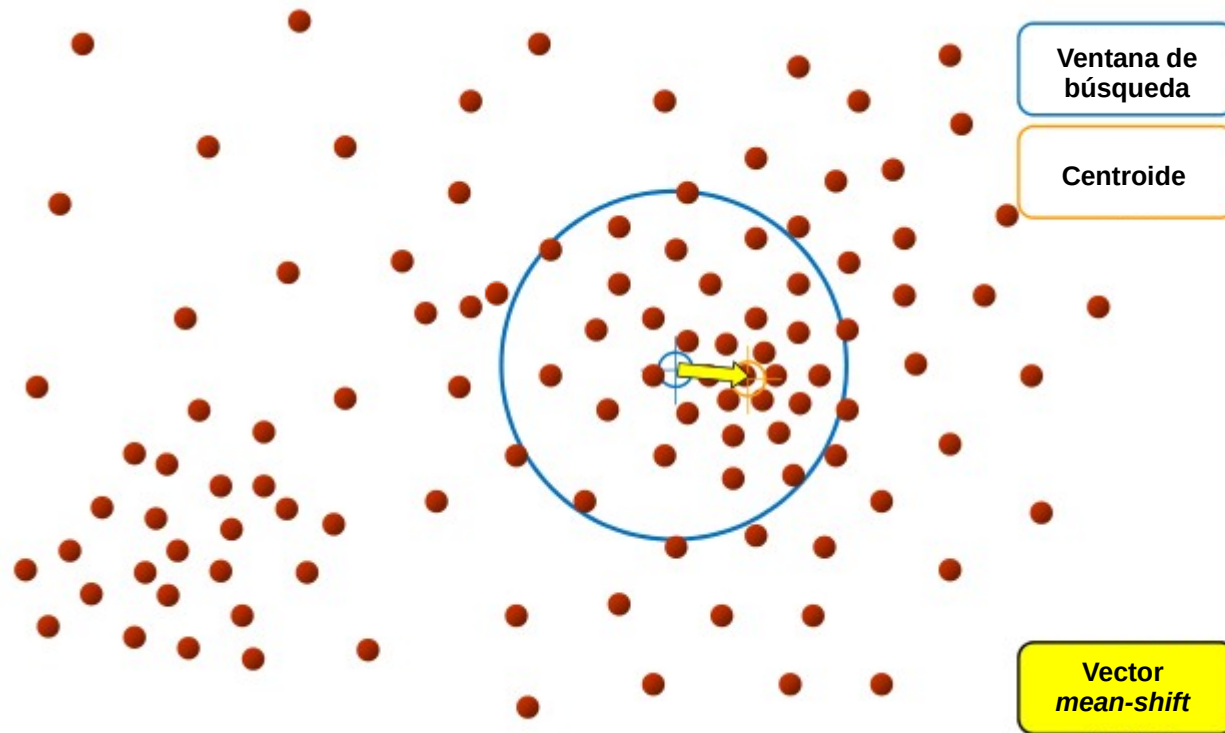
Desplazamiento de la media (*mean-shift*)

Mean-shift



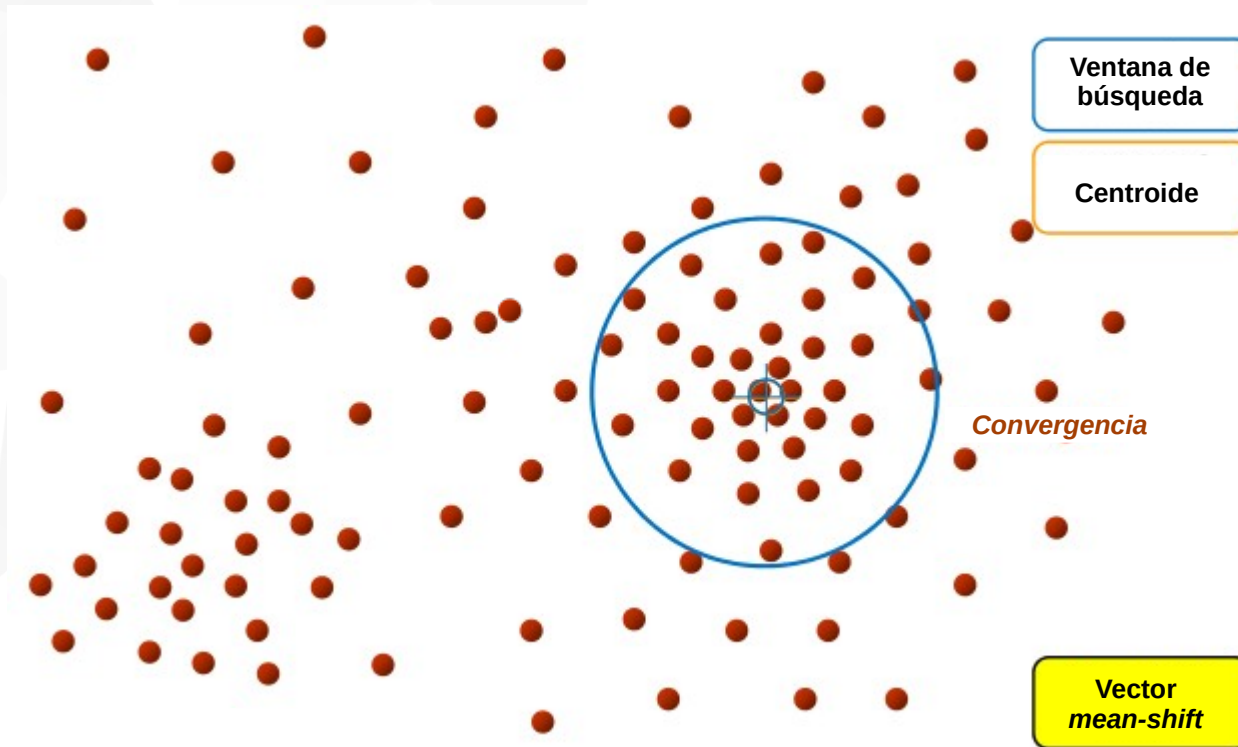
Desplazamiento de la media (*mean-shift*)

Mean-shift



Desplazamiento de la media (*mean-shift*)

Mean-shift

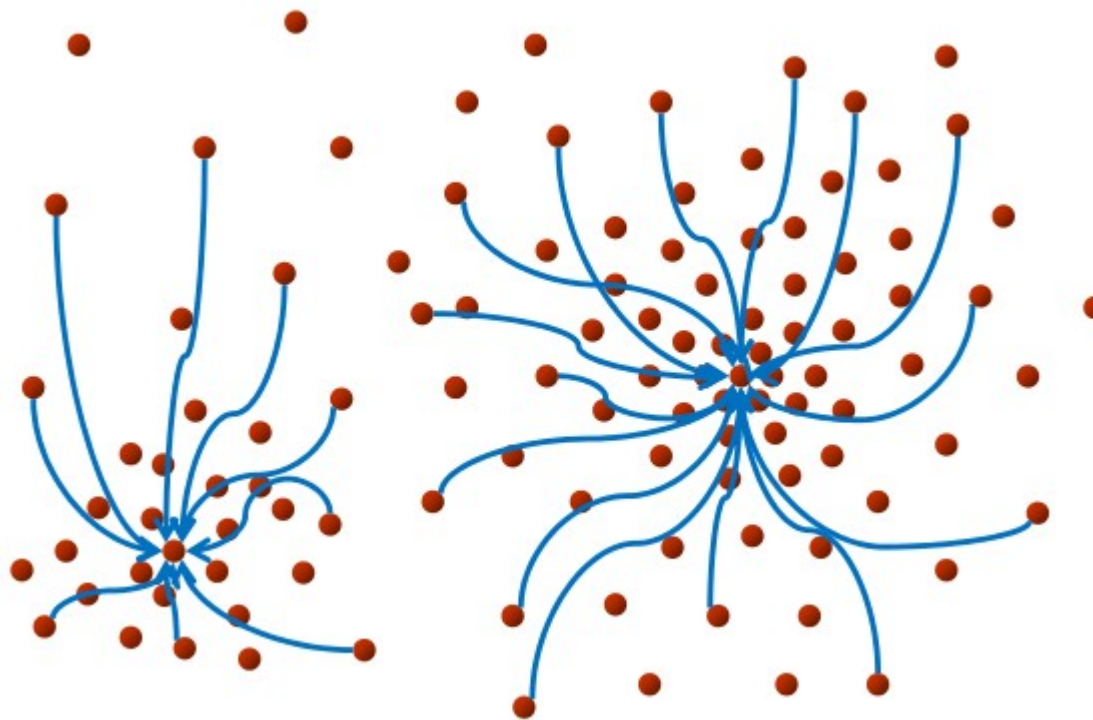


Desplazamiento de la media (*mean-shift*)

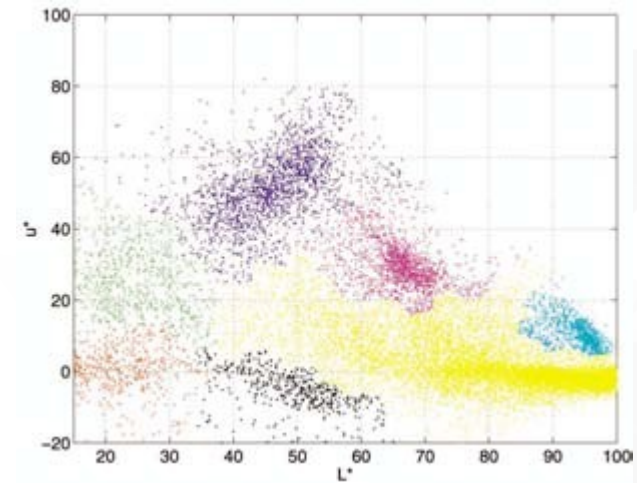
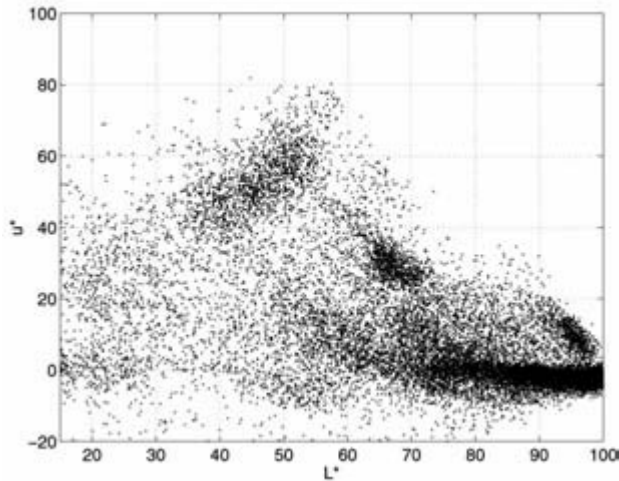
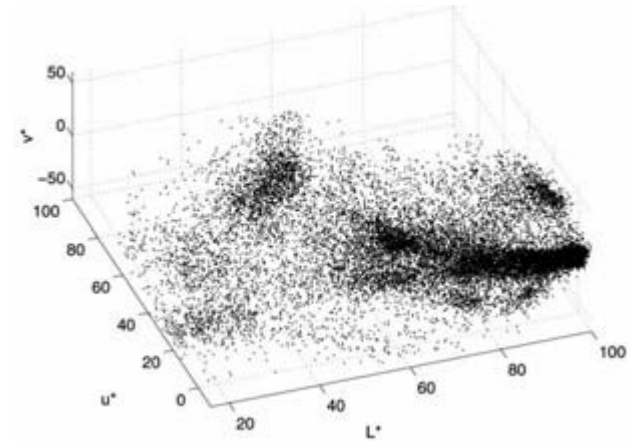
- ▼ La segmentación mediante este método consiste en agrupar puntos que converjan en posiciones similares:
 - ▼ Definir cada píxel como un punto en el espacio de características (color, posición, etc.).
 - ▼ Inicializar ventanas de búsqueda centradas en cada punto individual.
 - ▼ Ejecutar *mean-shift* en cada ventana hasta alcanzar la convergencia.
 - ▼ Agrupar los puntos que terminan en posiciones cercanas.
- ▼ Propiedades:
 - ▼ No es necesario especificar el número de agrupaciones.
 - ▼ Hay que definir el tamaño de la ventana de búsqueda.

Desplazamiento de la media (*mean-shift*)

Segmentación mediante *mean-shift*



Desplazamiento de la media (*mean-shift*)



Desplazamiento de la media (*mean-shift*)

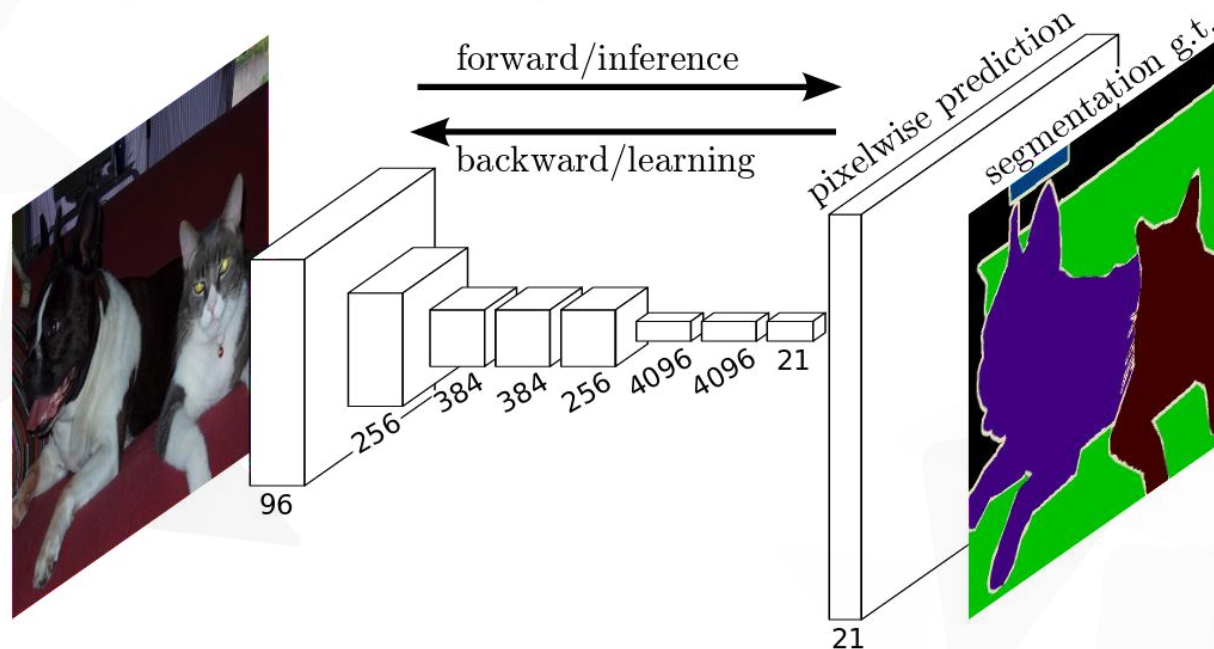
- ▼ Ventajas:
 - ▼ No supone agrupaciones esféricas.
 - ▼ Hay un único parámetro (el tamaño de la ventana).
 - ▼ Encuentra un número variable de regiones.
 - ▼ Es robusto a los *outliers*.
- ▼ Inconvenientes:
 - ▼ El resultado depende del ancho de banda.
 - ▼ Alto coste computacional.
 - ▼ No escala bien con las dimensiones del espacio de características.

Segmentación semántica

- ▼ Idea general
- ▼ DNNs en OpenCV

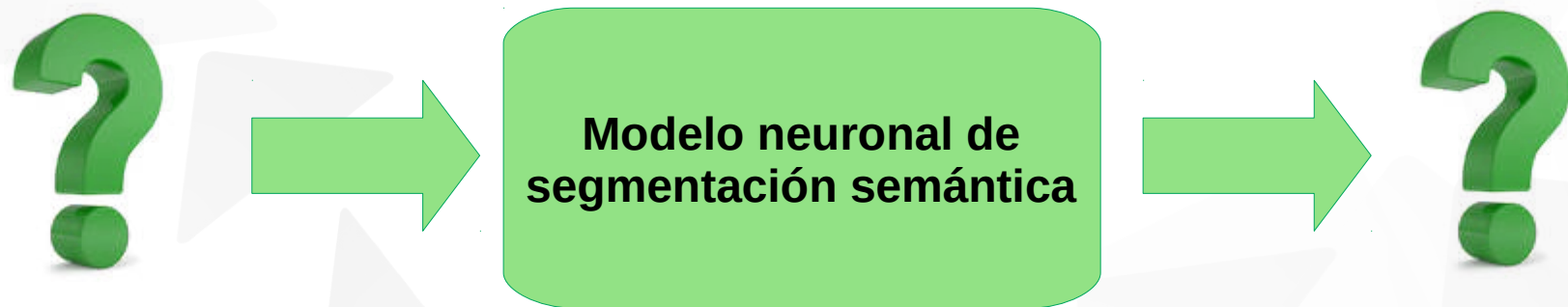
Segmentación semántica

- Basada en técnicas de *Deep Learning*
- A cada píxel se le asigna una etiqueta asociada con una categoría



Segmentación semántica

- ▼ ¿Cómo utilizar un modelo ya entrenado?
 - ▼ Normalización de la entrada
 - ▼ Interpretación de la salida



Segmentación semántica

- ▼ ¿Cómo utilizar un modelo ya entrenado?
 - ▼ **Normalización de la entrada**
 - ▼ Interpretación de la salida



Redimensión, resta de la media,
separación en canales, ...



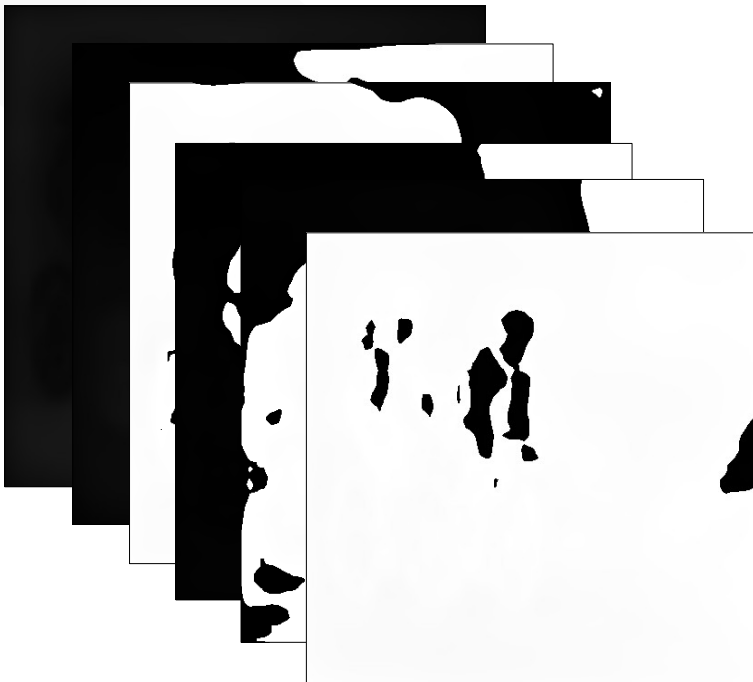
Segmentación semántica

▼ ¿Cómo utilizar un modelo ya entrenado?

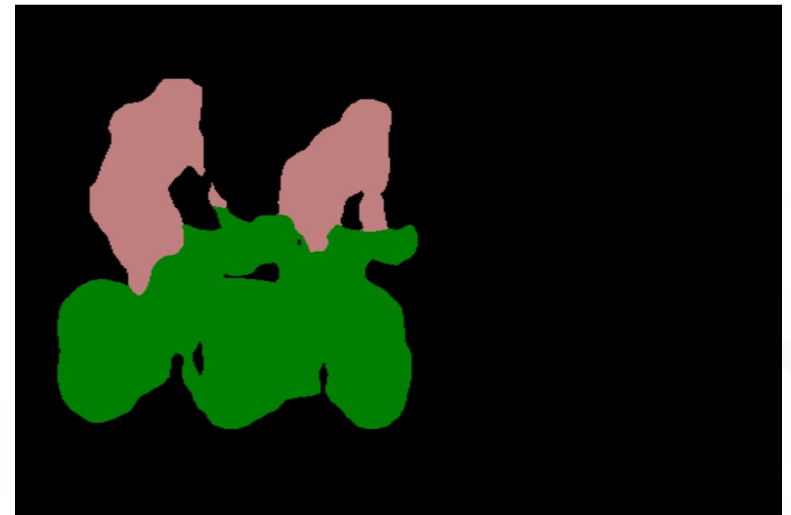
▼ Normalización de la entrada

▼ Interpretación de la salida

Un plano de imagen por categoría: en cada plano de imagen, el valor de un píxel representa la puntuación para esa categoría



Por cada píxel, escoger la categoría de máximo valor



Cada categoría tiene asociada un color para la visualización

Segmentación semántica

▼ Visualización del resultado



0,4



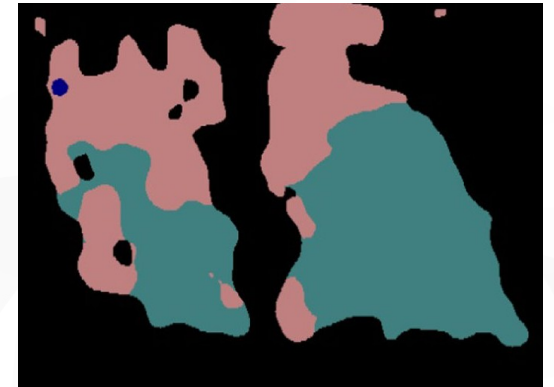
0,6

Suma ponderada de la imagen original y la obtenida por la red



Segmentación semántica

- ▼ Fortalezas:
 - ▼ Grandes resultados en contextos controlados.
- ▼ Limitaciones:
 - ▼ Requieren de una gran cantidad de datos etiquetados.
 - ▼ Los resultados son dependientes del conjunto de categorías utilizado durante el entrenamiento.



DNNs en OpenCV

▼ Módulo **dnn**

- ▼ Solo para inferencia. No puede usarse para entrenamiento.
- ▼ Permite cargar un modelo entrenado con distintos *frameworks*: Caffe, Tensorflow, Torch, ...
- ▼ Funciones para normalizar la entrada a la red.
- ▼ Clase *Net*: representa el modelo neuronal.
 - ▼ Métodos para asignar una entrada.
 - ▼ Métodos para inferir la salida.
- ▼ El resultado de la red se representa a través de un objeto de tipo *Mat*.

DNNs en OpenCV

- ▼ Funciones para cargar un modelo:
 - ▼ **cv::dnn::readNetFromCaffe** (const String &**prototxt**, const String &**caffeModel**=String())
 - ▼ **cv::dnn::readNetFromTensorflow** (const String &**model**, const String &**config**=String()).
 - ▼ **cv::dnn::readNetFromTorch** (const String &**model**, bool isBinary=true, bool evaluate=true)
 - ▼ Hay que especificar en los parámetros las rutas de los ficheros que contienen la configuración de la red y sus pesos.
 - ▼ En algunos frameworks (*Torch*) el modelo completo está en un único fichero.
 - ▼ Existen funciones genéricas para cargar un modelo desde cualquier framework.

DNNs en OpenCV

- ▼ La clase **Net**:
 - ▼ Todas las funciones anteriores devuelven un objeto de la clase **Net**.
 - ▼ Dicho objeto es la representación en memoria del modelo neuronal.
 - ▼ Métodos fundamentales:
 - ▼ void **setInput** (InputArray **blob**, ...): asigna a la entrada de la red los datos indicados en *blob*. El parámetro *blob* se representa a través de un objeto **Mat** que contiene los datos en el formato adecuado.
 - ▼ **Mat forward**(...): obtiene la salida de la red para la entrada especificada en *setInput*. Devuelve dicha salida en un objeto de tipo **Mat** con el formato concreto del modelo.

DNNs en OpenCV

- ▼ Normalización de la entrada:
 - ▼ void **cv::dnn::blobFromImage** (InputArray **image**, OutputArray **blob**, double **scalefactor**=1.0, const Size &**size**=Size(), const Scalar &**mean**=Scalar(), bool **swapRB**=false, bool **crop**=false, int **ddepth**=CV_32F)
 - ▼ Prepara la imagen indicada en *image* para adecuarla a la entrada de la red.
 - ▼ Redimensiona, intercambia canales, normaliza, ...
 - ▼ Resultado en *blob*: 4 dimensiones
 - ▼ 1ª dimensión: número de imágenes (1)
 - ▼ 2ª dimensión: número de canales (p.e. 3 en RGB)
 - ▼ 3ª dimensión: alto de la imagen procesada
 - ▼ 4ª dimensión: ancho de la imagen procesada

DNNs en OpenCV

- ▼ Interpretación de la salida:
 - ▼ Resultado de la llamada al método *forward*:
 - ▼ *Mat output = net.forward()*
 - ▼ Formato de *output* en segmentación semántica:
 - ▼ 4 dimensiones:
 - ▼ 1ª dimensión: número de imágenes (1)
 - ▼ 2ª dimensión: número de categorías
 - ▼ 3ª dimensión: alto de la imagen
 - ▼ 4ª dimensión: ancho de la imagen
 - ▼ Acceso al valor para una categoría (*cat*) de un píxel situado en una posición (*f*, *c*):
 - ▼ *int idx[4] = {0, cat, f, c}*
 - ▼ *float val = output.at<float>(idx)*