VISIÓN ARTIFICIAL:PRÁCTICA 3

segmentación, descriptores de imágenes & SIFT

Marc Ferrer & BOGDAN MARCUT

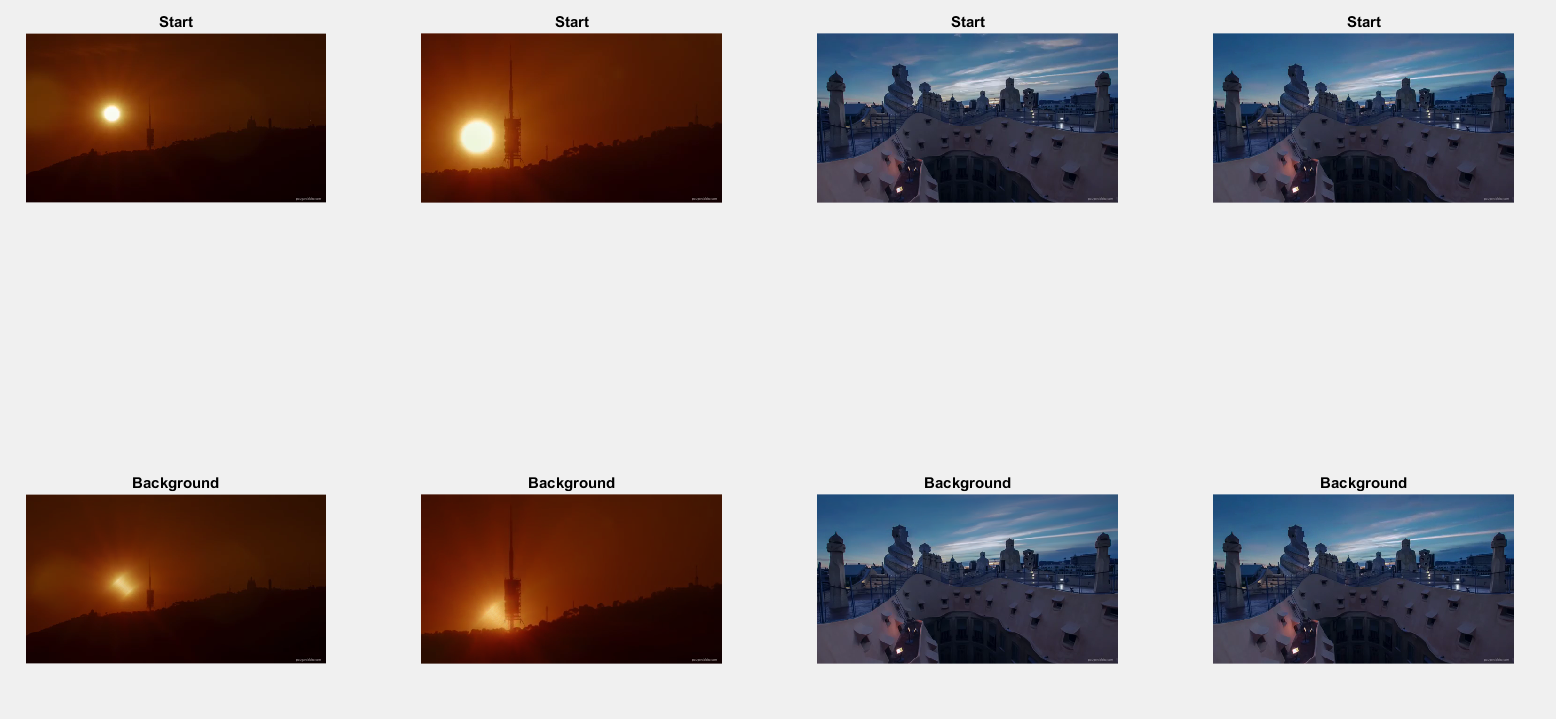
2016

# 3.1 Métodos de “background substraction”

**a) Encontrar donde se acaba una escena y comienza otra (estos frames se denominan *shots* del vídeo). ¿Qué medida de las imágenes se puede utilizar para distinguir las escenas?**

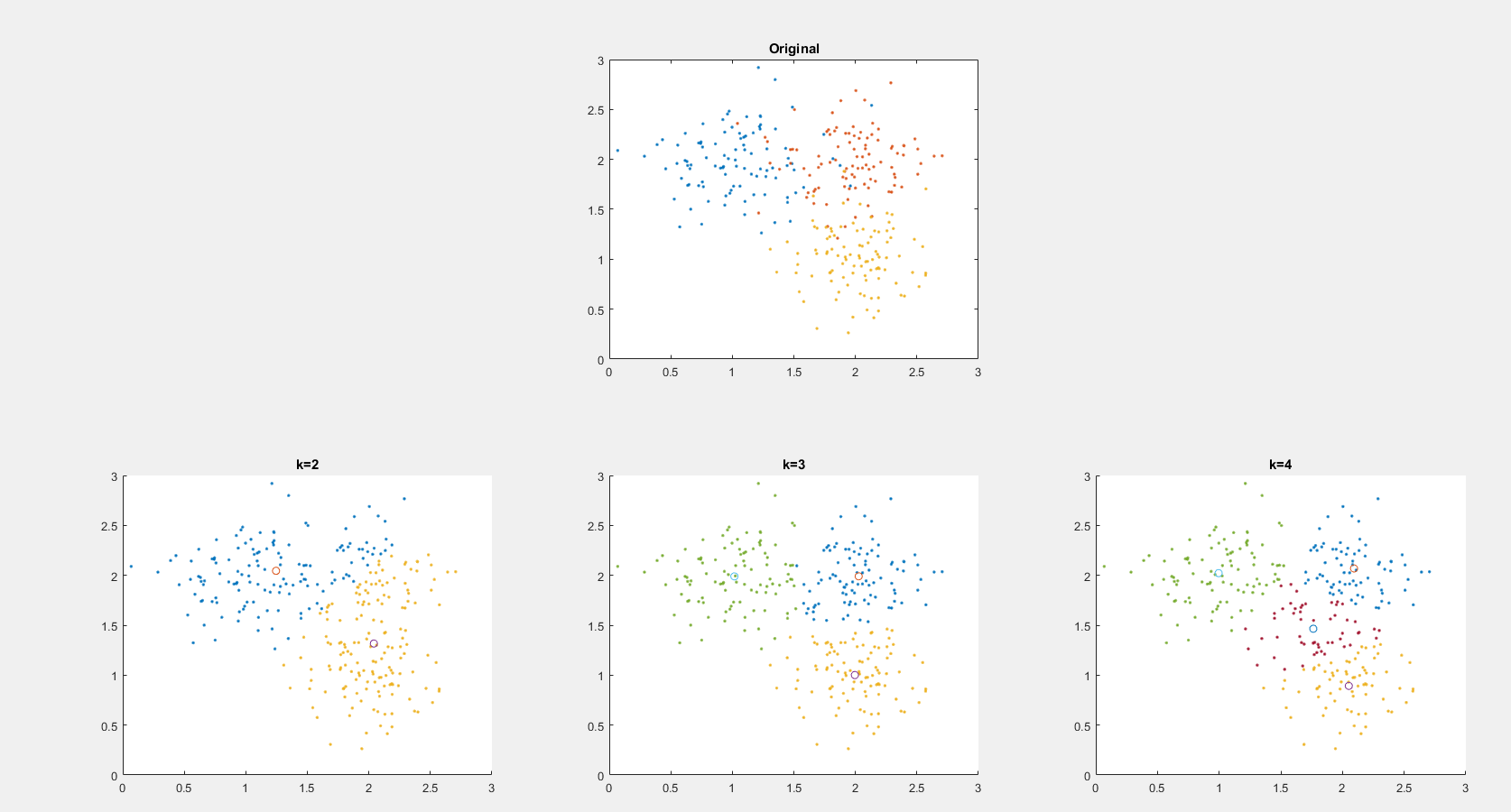
Para encontrar una medida que nos sirviera para distinguir las escenas hemos utilizado la función de Matlab “*imabsdiff*” que nos permite calcular la diferencia en valor absoluto como distancia de la cantidad de píxeles para un cierto umbral. Así pues, una vez tenemos este valor que nos permite distinguir las escenas aplicamos un filtro de la mediana des del *shot* actual hasta que este termine

**b) Aplicar un algoritmo de background substraction (consultar el material**

**de teoría).**

# 3.2 Métodos de agrupación de datos numéricos

**a) La función gaussRandom (mu, sigma, numSamples), proporcionada con el enunciado, permite generar nubes de puntos con una distribución gaussiana con matriz de covarianza diagonal, utilizando como centro las coordenadas de mu. Genera tres nubes de 100 puntos con centros [1 2], [2 2] y [2 1]. En los tres casos utilizar una desviación estándar de 0.1 en todos los ejes. Visualiza los puntos generados (help: plot).**

**b) Utilizar el método kmeans para agrupar los datos anteriores. Visualizar en un mismo plot (help: subplot) una primera fila con los datos originales, y el resultado (incluyendo los centros) de las agrupaciones con 2, 3 y 4 centros respectivamente en la segunda fila del subplot, utilizando diferentes colores**.

**c) Comenta los resultados que has encontrado, valorando el número de agrupaciones que has encontrado en cada caso y la similitud con los datos iniciales.**

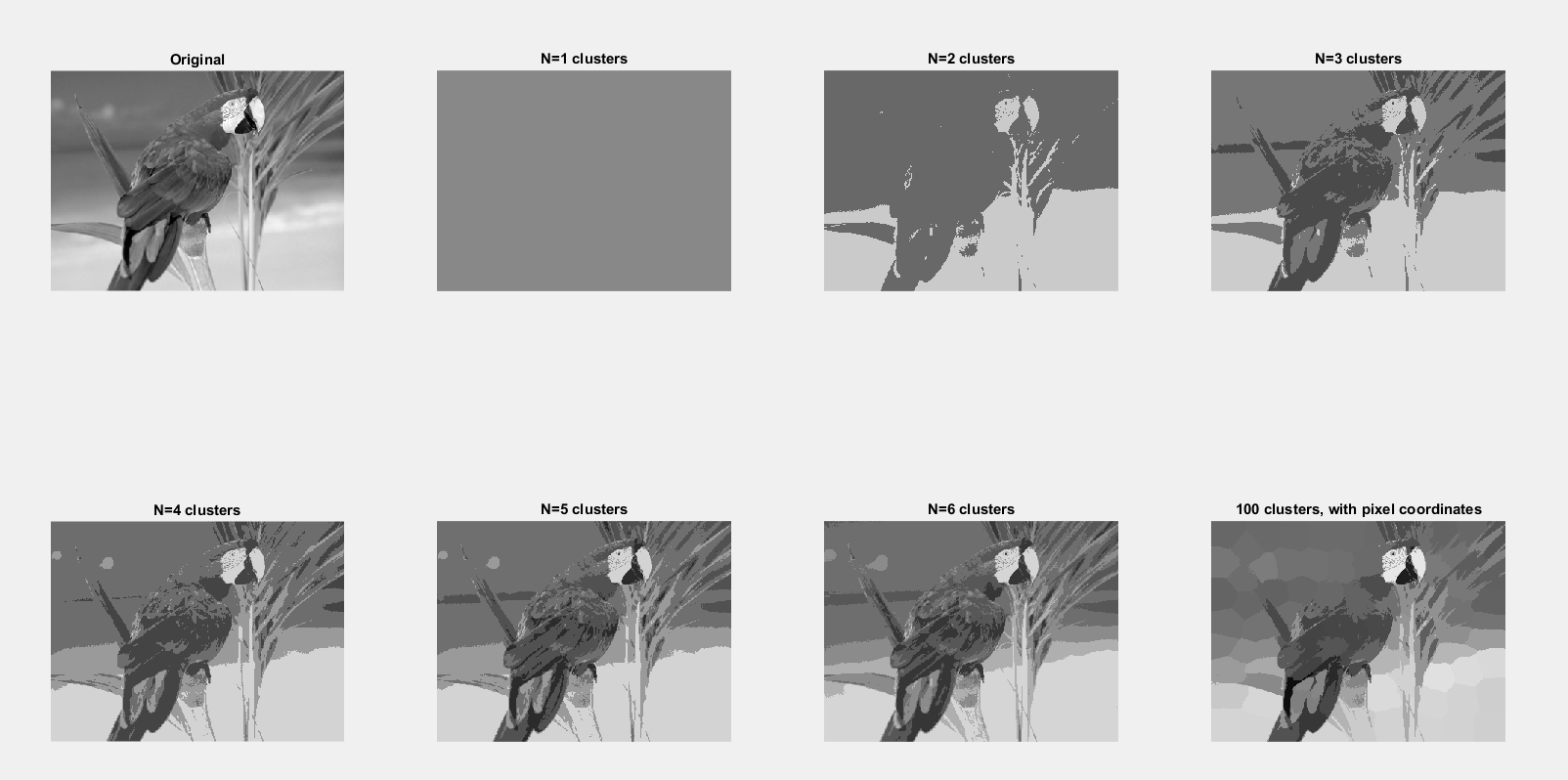
Podemos observar que el k-means da mejor resultado que el mean-shift, utilizando las nubes generadas por *gaussRandom*, ya que conocemos la cantidad de clúster que existen. Sino conociéramos la cantidad de clústeres, utilizar las nubes del mean-shift sería más eficiente que un k-means.

# 3.3 Métodos de agrupación: segmentación en el espacio RGB

**a) Lee la imagen ‘loro.png’. Conviértela a escala de grises y aplica la segmentación con el *kmeans*. Prueba diferentes valores de *k* para encontrar la mejor segmentación.**

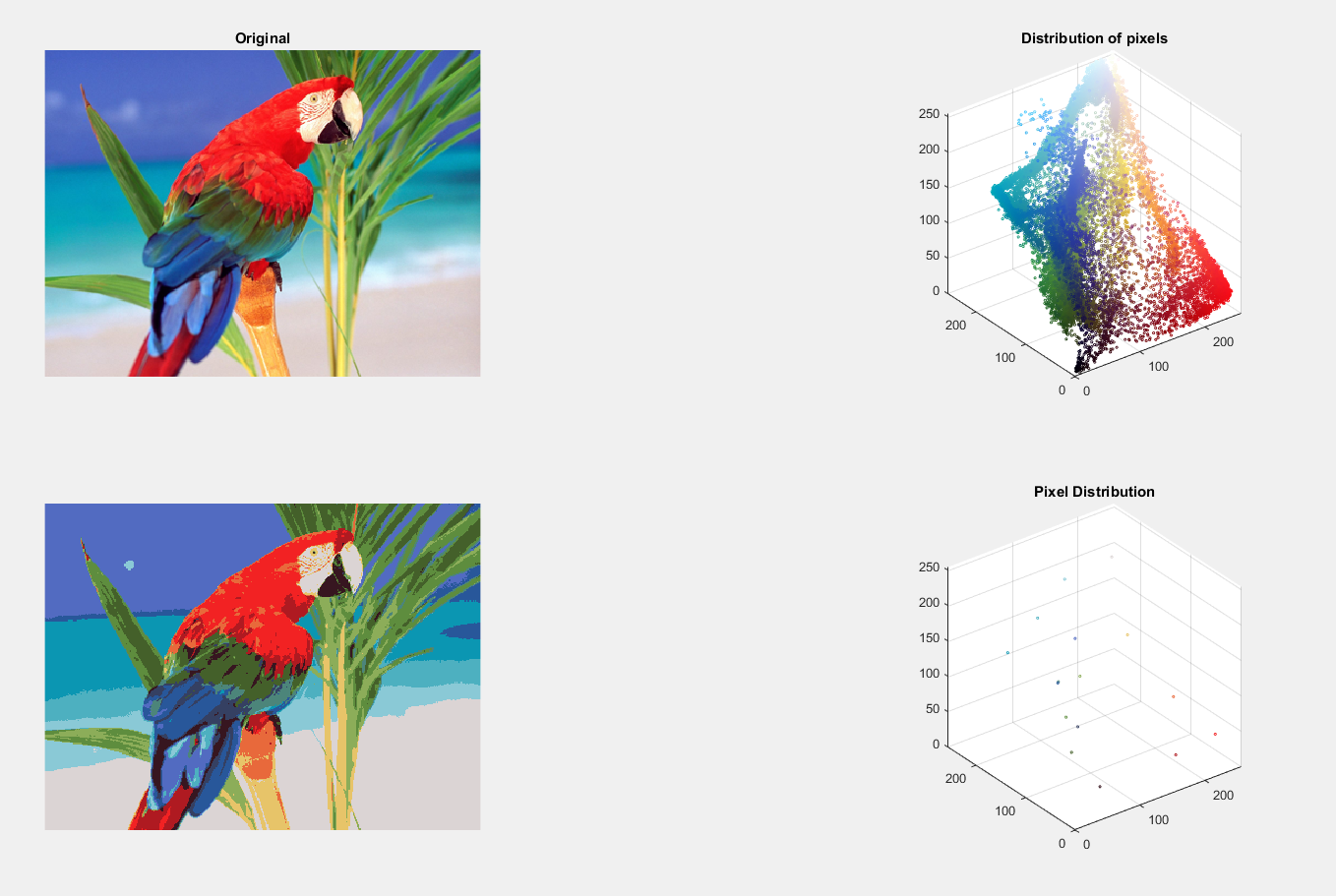
**b) Visualiza la imagen segmentada utilizando el nivel de gris promedio encontrado con el método de segmentación. ¿A qué corresponde?**

**c) Añade como características las coordenadas de los píxeles y comprueba si mejora el resultado de la segmentación.**

**d) Visualiza el resultado anterior en una figura junto con la distribución de sus colores (utiliza la función *plotPixelDistribution* facilitada con el enunciado).**

**e) A partir de la imagen de entrada, crea un matriz que contenga en cada fila la tripleta RGB de un píxel de la imagen. Tendrá tantas filas como píxeles haya en la imagen.**

**f) Utilizar el método *kmeans* para reducir el número de colores de la imagen a 16 colores diferentes.**

**g) Visualizar en una misma figura las imágenes del primer apartado, y la imagen con 16 colores junto con su distribución de colores (Figura 3).**

# 3.4 Extracción de descriptores

**a) ¿A qué corresponden las variables *f* y *d* que devuelve el método *vl\_sift*? ¿Qué tamaño tienen? ¿A qué corresponden sus valores?**

‘F’ es donde se encuentra el marco de las imágenes, donde se encuentran los puntos de interés, y ‘d’ es el descriptor del frame ‘f’.

**b) En este apartado se muestran los puntos característicos detectados con SIFT en una misma imagen con rotación. Compara los resultados obtenidos antes y después de hacer la rotación y comenta lo que ves. ¿Hay invariancia a rotación? ¿Qué significa la línea que aparece en el interior de los círculos?**

Vemos que al rotar la imagen también rotan los círculos, es decir, los puntos se conservan, aunque giremos la imagen. Eso significa que los puntos encontrados son invariables frente a las rotaciones.

La línea es la orientación con mayor valor del descriptor. Si hay más de una línea significa que hay orientaciones con el mismo valor.

**c) En este caso, se comparan los resultados con una misma imagen a distintas escalas. Compara los resultados obtenidos antes y después de hacer el reescalado y comenta lo que ves. ¿Hay invariancia a escala? ¿Qué significa el tamaño de los círculos que se muestran?**

La diferencia entre las imágenes es que hay menos círculos en las imágenes redimensionadas. Eso nos indica que no hay invariancia cuando escalamos las imágenes.

**d) En este apartado se genera una imagen sintética y se calcula el descriptor en dos puntos distintos de la imagen. Verás que se generan dos figuras con el mismo formato, pero con una pequeña diferencia en el cálculo del descriptor. Fíjate en los descriptores de la fila inferior del *subplot*. ¿Qué diferencia encuentras entre los mostrados en la primera figura y la segunda?**