

Berkeley Segmentation Data Set and Benchmarck 500 (BSDS500)

Laboratorio de Segmentación

Juliana Sánchez Posada
Universidad de los Andes
Visión Artificial

j.sanchez679@uniandes.edu.co

Abstract

*En este artículo se investigaron dos métodos sencillos de segmentación: K-means y Modelo de mezcla de Gausianas. En los dos se utilizó el color y la posición del píxel dentro de la imagen, como espacio de representación para las imágenes (R,G,B,x,y), de tal forma que no sólo los píxeles con colores semejantes si no cercanos entre sí, pertenecieran al mismo grupo; mejorando así, la coherencia espacial de los resultados obtenidos. En el artículo se realiza una introducción a los dos métodos, se presentan los resultados obtenidos al segmentar con los dos métodos las imágenes de la base de datos de **test** del 'Berkeley Segmentation Data Set' y se evalúan y comparan los resultados obteniendo la curva de precisión-cobertura.*

1. Introducción

La visión artificial, un subcampo de la inteligencia artificial, tiene como objetivo principal programar los computadores de tal forma que éstos logren entender e interpretar las imágenes tal y como un humano es capaz de hacerlo [1]. Uno de los objetivos secundarios que hace parte de este objetivo principal corresponde al proceso de segmentación, cuyo fin es particionar la imagen en regiones, las cuales tengan una apariencia visual razonablemente homogénea.

1.1. Segmentación

La segmentación hace referencia al proceso en el cual una imagen logra dividirse en sus diferentes partes constituyentes, separando y localizando objetos de interés. Los algoritmos de segmentación se basan en propiedades de la imagen como los son su color, textura o intensidad [3] [1]. Al final del proceso se obtiene una imagen en la cual cada uno de los píxeles de una región son similares en alguna o varias de estas características y en la que las regiones adyacentes son significativamente diferentes con respecto a la(s) misma(s) característica(s) [2].

La imagen segmentada provee una descripción compacta de la imagen original, la cual se puede utilizar en procesos más complejos y de mayor nivel, tales como: reconocimiento, clasificación de escenas, entre otros.

Un proceso de segmentación de una imagen inicia con la transformación de la imagen al espacio de representación (color, textura o intensidad) en la cual se desea analizar. Este espacio de representación es posteriormente agrupado ('clustered'), de tal forma que cada píxel recibe una asignación a una etiqueta que indica a qué grupo u objeto pertenece.

En algunos casos, ya que la información que recoge el espacio de representación no es compacta y coherente de manera espacial, la forma como se obtienen resultados más satisfactorios que incluyan esta información, es incorporando la información de la posición de cada uno de los píxeles (x,y), a los vectores que contienen la información del espacio de representación, para posteriormente realizar una nueva agrupación [2].

1.2. BSDS and BM500

Para poder validar y comparar los diferentes métodos de segmentación y de detección de bordes que se han creado alrededor del mundo en los últimos años, el grupo de Visión por Computador ('Computer Vision Group') de Berkeley en el 2001, creo una gran base de datos abierta de llamada Berkeley Segmentation Data Set and Benchmarck 500 (BSDS500) [1].

La base de datos incluye imágenes naturales, que han sido manualmente segmentadas. Las segmentaciones manuales, corresponden a la verdad terreno, de tal forma que los resultados de segmentación y detección de bordes que se obtengan por nuevos métodos o algoritmos puedan ser comparados no sólo entre sí, si no con las segmentaciones humanas.

El desempeño de los métodos se evalúa con la curva de precisión-cobertura y con la máxima medida F (ver Fig. 1) [3], y es con esta información que es posible comparar los diferentes métodos ya propuestos.

Las medidas de precisión y cobertura para una segmentación determinada se obtienen [3]:

$$Precision = \frac{VP}{VP + FP} \quad (1)$$

$$Cobertura = \frac{VP}{VP + FN} \quad (2)$$

dónde VP corresponde a la suma de todos los verdaderos positivos, FP a la suma de todos los falsos positivos y FN a la suma de todos los falsos negativos.

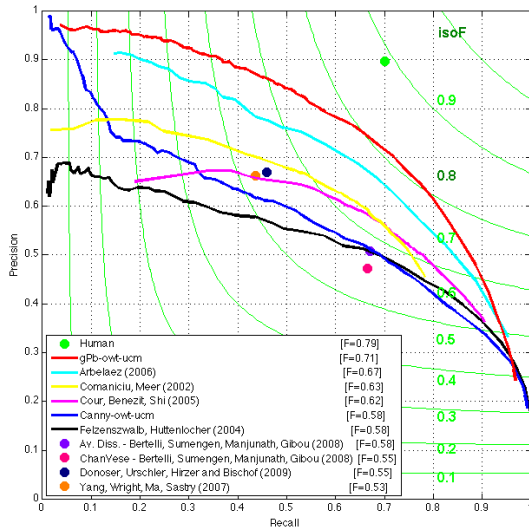


Figure 1: Curva de precisión-cobertura para diferentes algoritmos de segmentación en la BSDS.

Una interpretación más clara del significado de precisión y cobertura en el proceso de segmentación y de detección de bordes es:

- La precisión de una segmentación corresponde a la porción de píxeles de contorno que fueron identificados por la segmentación automática que corresponden a la verdad terreno.
- La cobertura corresponde a la porción de píxeles de contorno en la verdad terreno que fueron satisfactoriamente detectados por el algoritmo de segmentación.

Nota: En este laboratorio se utilizaron recursos del 'Berkeley Computer Vision Group', <http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html>

2. Métodos de segmentación por regiones implementados

2.1. K-means

K-means es un algoritmo de agrupación que mediante una técnica iterativa divide una imagen en k clusters. El algoritmo inicia escogiendo la posición de k centros de clusters (k dado por usuario). La forma como los escoge es aleatoria o basándose en algún modelo en particular, dentro de los que se encuentran Forgry, Random Partition, entre otros. El proceso iterativo consiste en encontrar la posición de los centros de los k clusters, tal que se minimice la distancia entre cada uno de los píxeles que se encuentran en el cluster con su centro. Diferentes distancias se pueden asociar a un espacio de $p - dimensiones$, y de acuerdo a ellas se calcula de manera diferente la posición asociada al centro de cada cluster. Algunas de las distancias comúnmente utilizadas son: 'euclidiana', 'city-block', y 'cosine'.

La forma como este algoritmo se aplica en la segmentación de una imagen consiste en transformar la imagen en el espacio de representación en el que ésta se quiere analizar (color, textura, intensidad o localización) o en una combinación de varios de ellos. Una vez transformada la imagen, se encuentra para cada píxel, se minimiza la distancia entre los factores correspondientes al espacio de representación y aquellos que se encuentran asociados al centro de cada cluster.

Algunas de las desventajas [3] que presenta este algoritmo son:

- requiere conocimiento previo del número de clusters k ,
- la solución aproximada depende en la inicialización de la posición de los clusters y puede converger a un mínimo local, lo que no corresponde a una solución óptima,
- prefiere clusters de tamaños semejantes y convexos,
- realiza una asignación fuerte sobre el cluster al que pertenece cada píxel.

Dado que el algoritmo es muy rápido y teniendo en cuenta que la calidad de la solución final depende del proceso de inicialización de los clusters, resulta común ejecutar el algoritmo varias veces para así quedarse con la solución que presente las mejores agrupaciones.

2.2. Modelo de mezcla de Gaussianas - GMM

GMM es un algoritmo de agrupación en el cual se pretende ajustar de manera óptima un grupo de puntos a n distribuciones gaussianas, dónde n es dado por el usuario. Para cumplir con este objetivo, el algoritmo estima de manera iterativa los parámetros óptimos para las $n - gaussianas$ (de

ahí el nombre: mezcla de gaussianas) de tal forma que éstas logren explicar los datos. Los clusters que se generan por éste método se forman representando la función de densidad de probabilidad de las variables observadas como una mezcla de densidades normales multivariadas. Semejante a k-means, se estima una distancia desde un punto a un cluster con la métrica de Mahalanobis.

El proceso mediante el cual se generan los diferentes clusters es el siguiente:

1. Se estiman unos parámetros iniciales asociados a las n gaussianas.
2. Se estiman las responsabilidades de acuerdo a los parámetros asignados.
3. Se estiman unos nuevos parámetros dadas las responsabilidades.
4. Se itera hasta que converja.

Semejante a k-means, la forma como este algoritmo se aplica en la segmentación de una imagen consiste en transformar la imagen en el espacio de representación en el que ésta se quiere analizar (color, textura, intensidad o localización) o en una combinación de varios de ellos. Una vez transformada, se encuentran de manera iterativa los parámetros asociados a las n gaussianas que se acomodan y expliquen de mejor manera los datos.

Algunas de las ventajas [3] que presenta este algoritmo son:

- permite que la distancia se adapte a la distribución de los datos,
- produce responsabilidades para cada cluster, es decir le asigna a cada píxel la responsabilidad de pertenecer a cada uno de los n clusters, realizando entonces una asignación suave.

Algunas de las desventajas [3] de GMM son:

- requiere conocimiento previo del número de clusters n ,
- la solución aproximada al igual que en k-means depende en la inicialización de la posición de los clusters y puede converger a un mínimo local, lo que no corresponde a una solución óptima.

3. Resultados y discusión de resultados

Las 200 imágenes que pertenecen a los datos de *test* del BSDS, se segmentaron utilizando los dos algoritmos previamente presentados. Para cada algoritmo se produjeron 2, 5, 8 y 10 clusters. Inicialmente se realizó una segmentación realizando la transformación de las imágenes sólo al espacio

de representación de color RGB. Sin embargo, se observó que en muchos casos se obtenían regiones que pertenecían a la misma categoría (cluster) pero que no se encontraban cercanas o adyacentes lo que demostraba poca coherencia espacial entre los objetos presentes en la imagen, razón por la cual se decidió realizar una nueva segmentación teniendo en cuenta la posición de cada uno de los píxeles en la imagen (XY). En la Figura 2 se puede ver la imagen original sin segmentar. En la Figura 3a se puede ver la imagen segmentada con K-means sólo con espacio de representación RGB y en la Figura ?? la imagen segmentada con K-means y con con espacio de representación RGB+XY. Semejante a la Figura 3, en la Figura 4 se puede ver la imagen segmentada con GMM y 10 clusters con los dos espacios de representación.

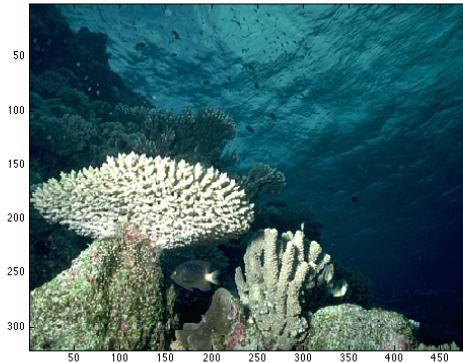
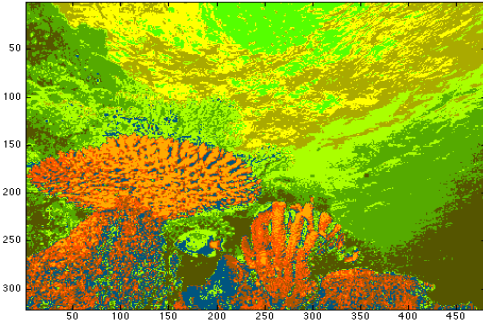


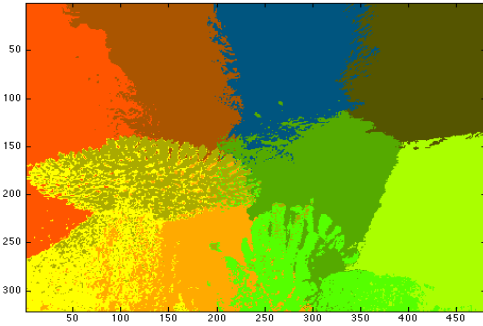
Figure 2: Imagen Original

Si se observan los resultados presentados por k-means en la Figura 3, inicialmente se podría pensar que la mejor segmentación corresponde a la que se presenta en la Figura 3a. Sin embargo, si se analiza mejor la imagen se puede ver que hay regiones que de acuerdo a la segmentación realizada por este algoritmo (k-means) con éste espacio de representación (RGB), no pertenecen a la misma categoría a pesar de presentar colores muy semejantes. Esto confirma el hecho de que el mejor espacio de representación es aquel que además de color tenga en cuenta la posición del píxel dentro de la imagen, pues de esta manera se descarta la idea de que los píxeles que pertenecen a una categoría además de presentar colores cercanos, se encuentren cercanos entre sí, obteniendo una mayor confianza en que estos sí pertenecen al mismo objeto.

Lo anterior a su vez ocurre con los resultados presentados por GMM, en los cuales es posible ver cómo cuando no se tiene en cuenta dentro del espacio de representación la posición X,Y del píxel en la imagen, el proceso de segmentación permite que zonas que no son adyacentes hagan



(a) Imagen segmentada con k-means, espacio de representación RGB

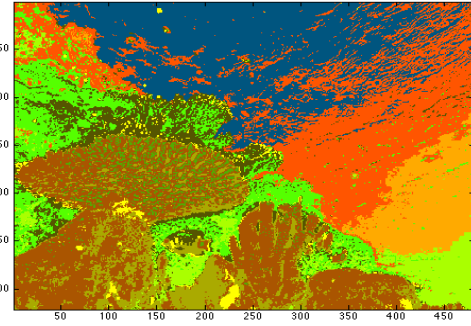


(b) Imagen segmentada con k-means, espacio de representación RGB+XY

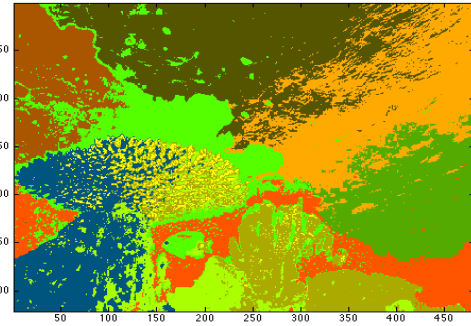
Figure 3: Imágenes segmentadas con k-means y 10 clusters para dos espacios de representación

parte de una misma categoría, lo cual en muchos casos no tiene sentido. Sin embargo, el sólo hecho de agregar la posición de cada píxel dentro del espacio de representación a segmentar, la segmentación resultante cambia, obteniendo resultados mucho mejores, con regiones cerradas y contiguas. Teniendo en cuenta estos resultados la base de datos de *test* de BSDS se segmentó haciendo uso del espacio de representación RGB+XY.

Ahora, si se comparan a simple vista los resultados obtenidos por los dos métodos de segmentación con el espacio de representación (RGB+XY) (Figura 3b y Figura 4b), es posible observar que el método que obtiene mejores resultados es GMM. Lo anterior ya que las desventajas mencionadas de k-means y la restricción incluida por la posición del píxel en la imagen hace que ésta muestre una segmentación inicialmente con regiones del mismo tamaño y rectangulares, no muy consecuente con las verdaderas regiones de la imagen. Algo diferente ocurre con la segmentación final que se obtiene con GMM, la cual logra separar de manera correcta los diferentes objetos de la imagen, mostrando regiones con formas semejantes a las de los ob-



(a) Imagen segmentada con espacio de representación RGB



(b) Imagen segmentada con espacio de representación RGB+XY

Figure 4: Imágenes segmentadas con GMM y 10 clusters para dos espacios de representación

jetos en la imagen.

Los resultados de segmentación obtenidos con los dos métodos y los 2, 5 8 y 10 clusters para la imagen original presentada en la Figura 2, se presentan en las Figuras 6 y 7.

Algunos de los argumentos que se modificaron en los métodos ya establecidos en *MATLAB* para y GMM fueron:

- K-means: se corrieron 3 replicas para cada imagen, de tal forma que se pudiera escoger la mejor segmentación obtenida para la imagen, eliminando aquellas en las que se alcanzaba un mínimo local.
- GMM: Se estableció un valor de 0.0001 para la regularización del método de tal forma que la diagonal de la matriz de covarianza siempre fuera positiva.

La Tabla 1 muestra los resultados cuantitativos para los métodos implementados y los compara con el algoritmo de UCM. A su vez la Figura 5 muestra los resultados obtenidos para la curva precisión-cobertura. Los valores F obtenidos para los dos métodos muestran mejores resultados para GMM que para k-means, demostrando las limitaciones que

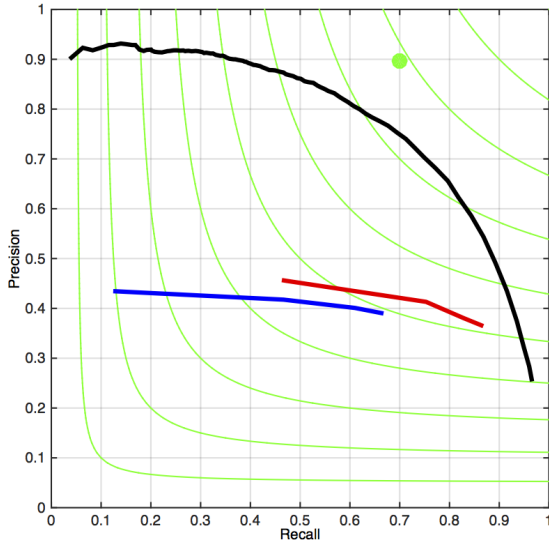


Figure 5: Curva de precisión-cobertura para k-means (azul) y GMM (rojo) y UCM (negro) al segmentar las imágenes de *test* del BSDS.

tiene este método por la asignación dura en la categoría que le asigna a cada píxel y por la definición misma de los clusters que hace que éstos tengan tamaños semejantes. A pesar de que los resultados obtenidos por k-means y GMM no son tan sobresalientes como los obtenidos por UCM, estos muestran una cobertura apreciable y significativa (siendo mejor para GMM que para k-means), mientras que mantienen una precisión casi estable (entre 38 y 45%).

Table 1: Resultados de las segmentación con K-means y GMM y su comparación con UCM.

Método	ODS: F	OIS:F	Área PR
K-means	0.49	0.51	0.23
GMM	0.53	0.58	0.17
UCM	0.73	0.76	0.73

La alta cobertura presentada especialmente por el método GMM implementado, indica entonces que una mayor cantidad de píxeles de contorno en la imagen, que eran reconocidos como contorno en la verdad terreno, fueron mejor identificados por el método GMM que por k-means. Ahora, la precisión casi constante en los dos métodos, indican que la porción de píxeles de contorno que fueron identificados por la segmentación como píxeles de contorno y que sí correspondían a contorno en la verdad

terreno se mantuvo constante para los dos métodos.

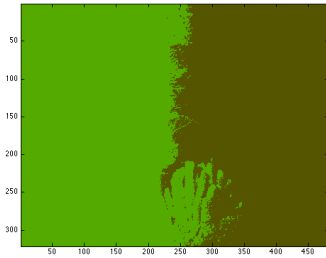
4. Conclusiones

Fue posible implementar dos métodos comunes de segmentación (k-means y GMM) y con ellos segmentar la base de datos de test del BSDS. Con los resultados obtenidos en la segmentación, fue posible realizar una comparación cuantitativa de los métodos implementados con el método conocido como UCM (carta de fronteras ultramétricas). A pesar de que los resultados obtenidos por los métodos implementados no fueron tan sobresalientes como UCM, fue posible observar las desventajas que presentan los dos métodos implementados, es especial k-means, y la dificultad existente en la correcta selección del espacio de representación requerido para lograr a partir de estos métodos sencillos segmentar imágenes complejas que contienen información de muchos objetos.

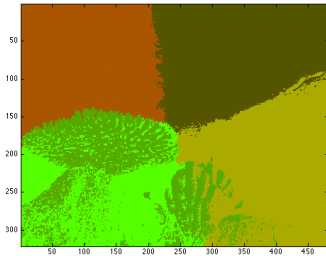
A pesar de que estos métodos se pueden mejorar, se ha visto en la literatura que éstos dos métodos no logran nunca obtener mejores resultados que métodos como UCM. Lo anterior ya que el mismo planteamiento de los métodos hace que estos sean poco flexibles, y que por lo mismo puedan llegar a ser buenos segmentando determinados tipos de imágenes pero que no lo sean en otros casos.

References

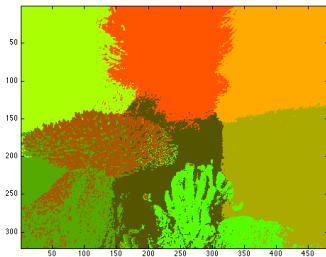
- [1] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. 33(5):898–916.
- [2] R. Huang, N. Sang, D. Luo, and Q. Tang. Image segmentation via coherent clustering in lab color space. 32(7):891–902.
- [3] Pablo Árbeláez. Presentaciones clase visión artificial.



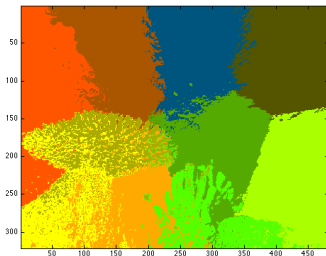
(a) Imagen segmentada con k-means, espacio de representación RGB+XY y 2 clusters



(b) Imagen segmentada con k-means, espacio de representación RGB+XY y 5 clusters

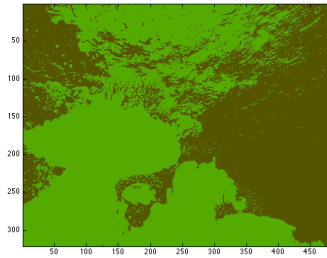


(c) Imagen segmentada con k-means, espacio de representación RGB+XY y 8 clusters

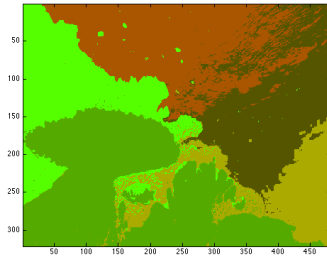


(d) Imagen segmentada con k-means, espacio de representación RGB+XY y 10 clusters

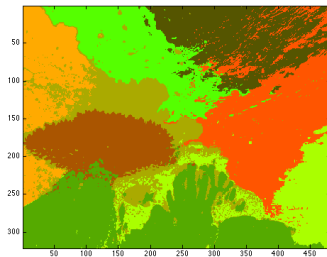
Figure 6: Imágenes segmentadas con k-means y espacio de representación RBG+XY



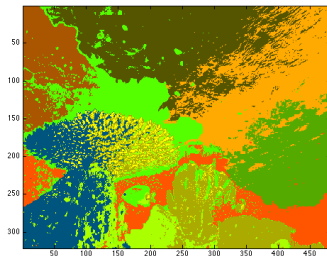
(a) Imagen segmentada con GMM, espacio de representación RGB+XY y 2 clusters



(b) Imagen segmentada con GMM, espacio de representación RGB+XY y 5 clusters



(c) Imagen segmentada con GMM, espacio de representación RGB+XY y 8 clusters



(d) Imagen segmentada con GMM, espacio de representación RGB+XY y 10 clusters

Figure 7: Imágenes segmentadas con GMM y espacio de representación RBG+XY