

Procesamiento de Imágenes

Fecha: 22-Julio-2020
Inicio: 1:00 pm

Segmentación de una Imagen

La segmentación es el campo del procesamiento de imágenes que divide una imagen digital en varias partes (grupos de píxeles).

Esto permite simplificar la representación de una imagen. Ejemplos:

- Obtener una imagen binaria
- Obtener los bordes de una imagen.

En este tema veremos un conjunto de algoritmos para determinar la mejor representación de una imagen usando 2 o 3 tonos, de grises.

Una manera de extraer objetos o partes de una imagen es seleccionar un umbral T , donde:

- $T \in \{0, 1, 2, \dots, 255\}$ si la imagen es de 8 bits
- $T \in [0, 1]$ si los píxeles de la imagen están normalizada.

Para esto, consideremos la siguiente segmentación. Sea A una imagen a escala de grises. Entonces la imagen segmentada B se obtiene de la siguiente manera:

$$B(i,j) = \begin{cases} a & \text{si } A(i,j) > T \\ b & \text{si } A(i,j) \leq T. \end{cases}$$

donde $a, b \in \{0, 1, 2, \dots, 255\}$ si A es de 8 bits o $a, b \in [0, 1]$ si A es una imagen normalizada.

Caso Particular:

$$B(i,j) = \begin{cases} 1 & \text{si } A(i,j) > T \\ 0 & \text{si } A(i,j) \leq T \end{cases}$$

Si A es normalizado

$$\text{ó } B(i,j) = \begin{cases} 255 & \text{si } A(i,j) > T \\ 0 & \text{si } A(i,j) \leq T \end{cases}$$

Si A es de 8 Bits

En general, uno puede hacer una segmentación de más de 2 partes. Ejemplo

$$B(i,j) = \begin{cases} a & \text{si } A(i,j) > T_2 \\ b & \text{si } T_1 \leq A(i,j) \leq T_2 \\ c & \text{si } A(i,j) \leq T_1 \end{cases}$$

En este caso tendríamos dos umbrales: $T_1 \leq T_2$.

La función del ruido en el procesamiento de imágenes usando umbrales

Si una imagen tiene una cantidad definida de escala de grises, el ruido en esa imagen modifica completamente este comportamiento. Este análisis se puede ver más claro usando el histograma de la imagen.

Algoritmo Para Detectar el Umbral Óptimo

① Umbral Global Básico: Este algoritmo se utiliza cuando los valores de los píxeles son separados en 2 grupos bien definidos. Este algoritmo se comporta de la siguiente manera:

Paso 1: Seleccionar un valor inicial de umbral $T^{(0)}$.

Paso 2: Para $k = 0, 1, 2, 3, \dots$

Paso 3: Usando $T^{(k)}$ dividir la imagen en 2 bloques:

Bloque 1: Intensidad $> T^{(k)}$

Bloque 2: Intensidad $\leq T^{(k)}$

Paso 4: Calcular el promedio de intensidad de la imagen original usando cada bloque. Sea m_1 el promedio usando el bloque 1, y sea m_2 el promedio usando el bloque 2.

Paso 5: Actualizar el valor del umbral: $T^{(k+1)} = \frac{1}{2} (m_1 + m_2)$

Paso 6: Si $|T^{(k+1)} - T^{(k)}| \geq \text{tol}$, detenerse. Sino $k = k+1$ y repetir Pasos 3-6.