

Práctica 1 (Opción 1)



**Estudio de medidas de distancia
invariantes a traslación y
rotación en espacios de color**

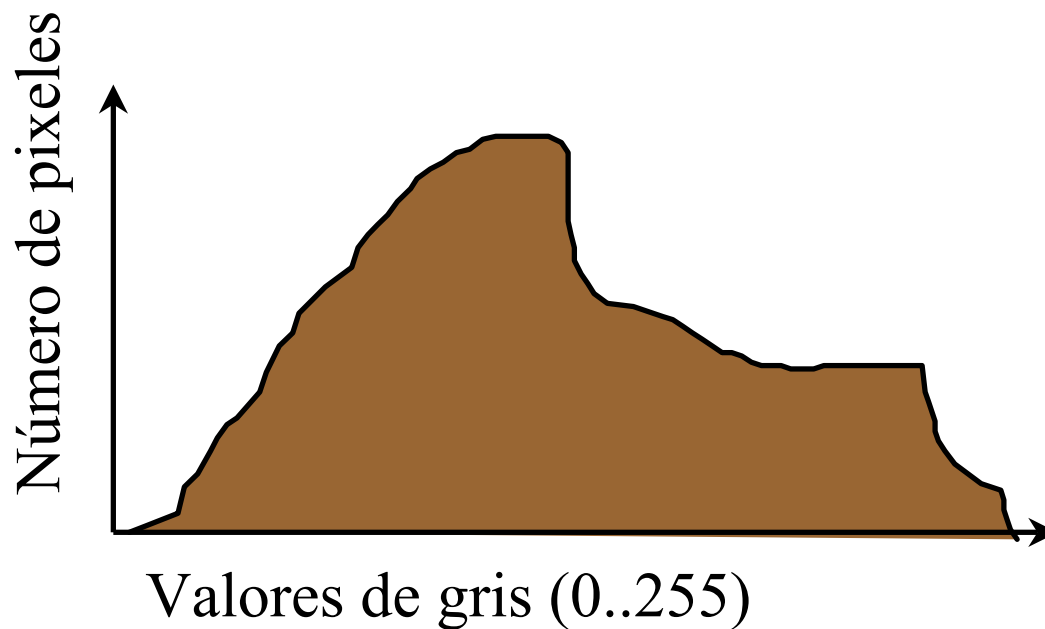
Parte I, Enunciado

- Escribir una función que calcule la distancia entre dos imágenes en color



- El espacio de color se debe poder elegir al menos entre dos opciones (rgb y hsv)
- La medida de distancia debe ser invariante a rotaciones y traslaciones (ej: histograma)

Detalles: Histograma de una Imagen en gris



Detalles: Histograma de una Imagen Color

- $\text{Hist}(r,g,b)$ es el área de la imagen (en píxeles) que es del color (r,g,b) . Función "hist_rgb"
- Invariante respecto a traslación y rotación (mide áreas)
- Este histograma es una matriz de tres dimensiones. El eje X corresponde a los valores de rojo, el Y al verde y el Z al azul; en general, al primer, segundo y tercer canal de color.
- Memoria ocupada por el histograma (tipo double)
 - $256*256*256*8=124\text{MB}!!!$ (completo)
 - $10*10*10*8=8\text{KB}$ (con 10 divisiones por eje)
- Reducir la memoria ocupada por el histograma
 - El histograma depende de la imagen y la precisión
 - precisión: número de valores diferentes considerados para cada canal (primario)

Parte I, Método



- Cargar dos imágenes diferentes de tetra-bricks (tcol1.bmp a tcol14.bmp)
- Calcular el histograma de color con un número determinado de divisiones por cada eje (array_3D=hist_rgb(imagen,N))
- Calcular la distancia euclídea entre los 2 arrays
- Hacer el mismo proceso pero en el espacio de color HSV
- Detalles:
 - la funcion hist_rgb recibe imágenes de tipo uint8
 - debido a la carga computacional no se debe poner $N > 10$

Parte I, Material a entregar



■ Función "Compara_color"

- Entradas: "imagen1", "imagen2", "espacio_de_color", N
- Salidas: "distancia_euclídea"

Parte II, Enunciado

- Para un conjunto de tetra-bricks dado, analizar si existe alguna medida de distancia capaz de emparejar automáticamente los t-b rotados

Detalles: Distancias entre Histogramas (I)

■ Distancia L1

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} |h1(r, g, b) - h2(r, g, b)|$$

■ Distancia L2 (Euclídea)

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} (h1(r, g, b) - h2(r, g, b))^2$$

■ Distancia L2 ponderada

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} \omega(r, g, b) (h1(r, g, b) - h2(r, g, b))^2$$

$\omega(r, g, b)$: función de ponderación definida por el usuario

Detalles: Distancias entre Histogramas (II)

■ Máximo (Mínimo)

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} \max(\min)(h1(r, g, b), h2(r, g, b))$$

■ Distancia normalizada

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} \frac{|h1(r, g, b) - h2(r, g, b)|}{h1(r, g, b) + h2(r, g, b)}$$

■ Distancia alternativa

$$\sum_{r=0}^{N-1} \sum_{g=0}^{N-1} \sum_{b=0}^{N-1} \max \left(\frac{h1(r, g, b)}{h2(r, g, b)}, \frac{h2(r, g, b)}{h1(r, g, b)} \right)$$

Parte II, Método



- Realizar una tabla de distancias de todos los tetra-bricks entre si (14x14 resultados)
 - Se debe poder elegir la medida de distancia (al menos 3), el espacio de color (al menos 2) y el número de divisiones de los canales para calcular el histograma de color.
 - Los resultados se analizarán automáticamente indicando qué tetra-brick se emparejaría mal por estar “más cerca” a uno de otro tipo que al del mismo tipo rotado.

Material a Entregar (II)

■ Función “Matriz_de_confusion”

- Entradas: nombre del directorio con las imágenes de tetra-bricks, tipo_de_distancia, espacio_de_color, N
- Salidas: Matriz de confusión y número de tetra-bricks mal emparejados.

■ Responder a:

- ¿Qué influye para que el tetra-brick rotado no esté a distancia 0 de su pareja?, ¿cómo se puede minimizar este efecto?
- ¿Por qué la misma medida de distancia en distintos espacios da resultados tan diferentes?, ¿cómo podemos mejorar los resultados en cada espacio de color?