

Práctica 2, ejercicios prácticos sobre procesamiento de imágenes digitales

1 Objetivos

El objetivo de estos ejercicios es poder utilizar funciones relacionadas tanto al tratamiento de imágenes binarias como de colores.

Se proponen ejercicios que abarcan:

- Operaciones por puntos en imágenes.
- Separación de colores por umbrales.
- Extracción de características geométricas.

2 Ejercicios

2.1 Revisión de bibliografía sobre operaciones en píxeles.

Revisión de bibliografía “Fundamentals of image processing”. Desarrollar los ejemplos del libro y probarlos en Matlab. Todos los ejercicios de adición, sustracción. Multiplicación, división. Bajar las imágenes que se utilizan en el libro.

Punto 3.2 “Operations upon pixels”. Punto 3.2.2 “Operaciones lógicas”, NOT, OR, XOR y funciones. Punto 3.2.3 “Thresholding”, función `im2bw`.

Puntos 3.4.2 “Adaptive thresholding”. Probar el método y comparar las imágenes de los resultados obtenidos.

2.2 Segmentación de colores

Usar el método de thresholding con dos umbrales “inicio y final” para separar regiones de colores específicos. En la imagen de la lapicera y el cuchillo “lapizcuchillo.jpg”. La aplicación presente en Matlab “App Thresholding”, permite realizar pruebas en forma visual y generar un código fuente para aplicar en Matlab. Se recomienda utilizar el espacio de colores CIELAB, por ser el espacio de colores que representa la similitud de colores de forma parecida al ojo humano.

- a) Encontrar una secuencia que permite aislar solamente el cuchillo, separar solamente el cuchillo.
- b.- Encontrar una secuencia de instrucciones que permita separar solamente la tapa del bolígrafo azul.

2.3 Tratamiento de características geométricas

Con la instrucción `regionprops` y usando la imagen `011.jpg_soM4.jpg`, la cual es una máscara binaria de defectos, desarrollar una secuencia de Matlab que permita obtener los siguientes datos:

- a) Contar cuantos elementos no conectados se encuentran en la imagen.
- b) Desplegar por cada elemento el area total de pixeles que ocupan.
- c) Por cada defecto encontrado, obtener las coordenadas de la región de pixeles y recortar dicho fragmento desde la imagen en colores RGB. Almacenar el resultado en archivos separados por manchas.

Información adicional y códigos fuentes pueden ser descargados desde:

<http://otroblogdetecnologias.blogspot.com/2016/11/extraccion-de-caracteristicas-en.html>

http://otroblogdetecnologias.blogspot.com/2016/11/extraccion-de-caracteristicas-en_24.html