

1.- (3 puntos) Transformaciones de intensidad y transformaciones espaciales en procesamiento de imagen: describa en qué consisten y ponga un ejemplo de cada una de ellas.

2.- (3 puntos) Segmentación:

- a) Segmentación con conocimiento del dominio.
- b) Segmentación basada en el flujo óptico: fundamento y características relevantes.
- c) Segmentación por sustracción del fondo: fundamento y características relevantes.

3.- (4 puntos) Una cinta transportadora lleva cruces de distinto tipo. Para el control de calidad, una cámara toma imágenes en escala de grises de la cinta en las que se detectan cruces normales (enteras negras) y cruces con algún defecto (con zonas grises). El fondo es blanco y también aparecen otros artefactos (ruido) del mismo color que los defectos encontrados en las cruces. En la figura 1 se muestra un ejemplo.

Desarrolle un programa en pseudocódigo para realizar las siguientes operaciones:

- a) Localizar las cruces no defectuosas en la imagen
- b) Localizada la cruz, encontrar de manera automática un punto interior a la región central de la cruz, donde se cruzan sus brazos (zona x en figura 2).

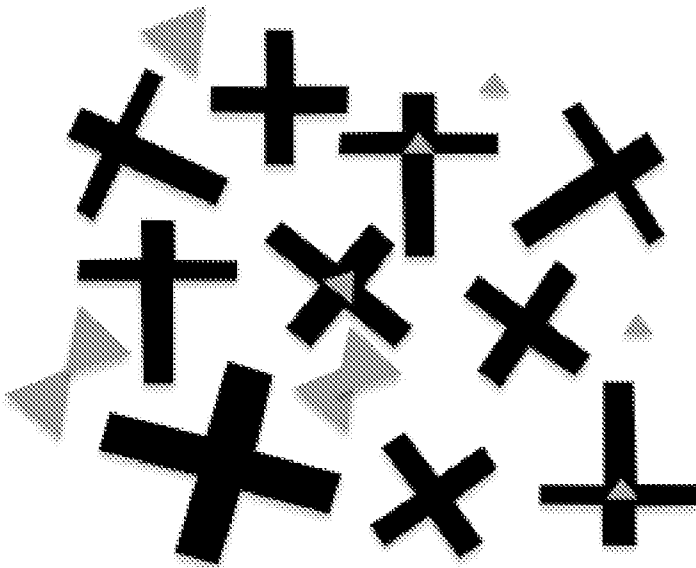


Figura 1: Imagen Io

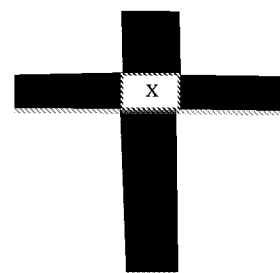


Figura 2: Zona central de la cruz

Para realizar el programa en pseudocódigo dispone de los siguientes operadores y estructuras de programación (continúa en la siguiente hoja de enunciado)

- a) Estructuras de control habituales (**for**, **if**, **while**, ...).

b) **Operaciones aritméticas y lógicas** sobre variables de tipo matriz, tanto en operación matricial como elemento a elemento.

c) Operadores de procesamiento de imagen:

- Sea **Ib** una imagen binaria
- Etiquetado: **L = bwlabel(Ib)** devuelve en **L** una imagen de las mismas dimensiones que **Ib** y con etiquetas distintas (1,2,3,...N) en cada uno de los N blobs independientes en conectividad 8 encontrados en **Ib**. Se supondrá que el fondo tiene la etiqueta "0".
- Rellenado de agujeros de los objetos: **Ib2 = bwfill(Ib)** Devuelve una imagen binaria de las mismas dimensiones que **Ib** y con los objetos de la imagen **Ib** y sus agujeros a "1".
- Mostrar imagen en pantalla: **imshow(I)**

d) Otros operadores:

[p] = find(X) encuentra los índices de **X** con valor distinto de cero.

[u] = unique(X) devuelve los valores de **X** sin repetición.

Paso de índice lineal a subíndice (fila, columna) y viceversa

[linearInd] = sub2ind(matrix2DSize, filaInd, colInd)

[filaInd, colInd] = ind2sub(siz, linearInd)

N = length(v) devuelve en **N** el número de elementos del vector **v**.

[f,c] = size(M) dimensiones en filas y columnas de la matriz **M**.

v = M(:) pasa una matriz N-dimensional **M** a vector unidimensional.

zeros(f,c) : genera una matriz de zeros de **f** filas y **c** columnas.

* Es muy probable que con los operadores descritos se pueda resolver el problema. Sin embargo, si considera que necesita más operadores, puede utilizarlos siempre que los justifique.

**** Atención: No basta con escribir el programa en pseudo-código. Debe comentarlo para justificar las decisiones tomadas.**