

# Prueba II

## “Procesamiento Digital de Imágenes”

Nombre:

Firma:

Fecha: 11/nov/2005

1
1
1
1

1.- Escriba el pseudo código para implementar la operación morfológica “dilatación”, con el siguiente elemento estructurante :

función Dilatacion( parámetro A ) retorna B

H=alto(A);

W=ancho(A);

K=[1 1 1 1];

3 pts

PARA f desde 1 hasta (H-4)

PARA c desde 1 hasta W

suma=0;

PARA ff desde 1 hasta 4

suma=suma+K(ff)\*A(f+ff,c);

3 pts

FIN ff;

SI suma>=1 B(f,c)=255, SINO B(f,c)=0;

FIN c;

FIN f;

4 pts

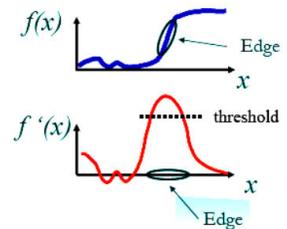
RETORNA B;

FIN;

Se debe tener precaución al definir los límites de los ciclos.

La operación dilatación exige que a lo menos uno de los píxeles del elemento estructurante coincida con uno de los píxeles de la imagen.

2.- Los algoritmos de detección de bordes, se fundamentan en la binarización de la derivada de la función de intensidad de una imagen. Sin embargo, al definir un umbral, se pierden bordes de menor intensidad y los bordes de mayor intensidad poseen diversos grosores. ¿Cómo se puede detectar bordes ideales (delgados) sin definir un umbral?



5 pts

La derivada de la función de intensidad representa la presencia de bordes en una imagen. Un borde se entiende como un cambio notorio en la función de intensidad. Dicho cambio puede ser gradual o instantáneo. En general en imágenes naturales el cambio es gradual. Como se muestra en la figura el mayor borde genera un “peak” en la derivada de la función. Si se define un umbral, el “grosor” del borde será proporcional al ancho de dicho “peak”. Una forma alternativa es simplemente tomar sólo los puntos “máximos locales” de la derivada, es decir considerar como borde, justamente el punto de inflexión de la función de intensidad. Para robustecer el algoritmo, se puede filtrar la función derivada para potenciar los cambios notorios de intensidad y eliminar los cambios menores.

Estos máximos o mínimos locales de la derivada, se pueden obtener directamente de los cruces por cero de la segunda derivada de la función de intensidad.

3 pts

Para destacar los bordes más relevantes, se puede considerar como intensidad del borde el valor de la primera derivada, justo en los cruces por cero de la segunda derivada.

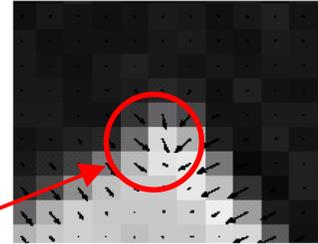
2 pts

3.- Explique conceptualmente como se pueden detectar esquinas basándose en el gradiente de una imagen.

3 pts

El gradiente de una imagen corresponde a la intensidad de la derivada y su dirección. Esto se representa gráficamente en la figura por pequeñas flechas o vectores que representan la intensidad de la derivada y apuntan en la dirección de cambio máximo de la intensidad.

Una esquina entonces corresponde a un subconjunto de los puntos de un borde. Los puntos de un borde son aquellos que tienen mayores valores en el gradiente. Pero las esquinas se caracterizan por un abrupto cambio en la dirección del gradiente



7 pts

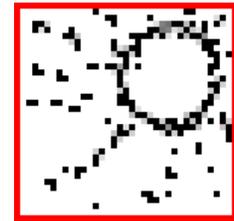
4.- Para detectar un objeto circular, explique conceptualmente el uso de la transformada Hough.

- Escriba los pasos a seguir y dibuje cualitativamente el plano de la transformada resultante.
- Comente sobre la robustez del algoritmo

a)

La transformada Hough, aplicada a la detección del círculo presente en la figura, se puede explicar en los siguiente pasos:

5 pts



1.- Cada par de puntos de la figura son posibles candidatos a pertenecer al perímetro de un círculo cuyo origen está en el centro de ambos puntos. Es decir se puede suponer que ambos puntos son diametralmente opuestos en un círculo dado.

2.- El círculo supuesto por cada par de puntos de la imagen, se puede representar por la distancia del centro al origen de un plano y por el ángulo desde el origen al centro del círculo. Lo que corresponde a la representación polar del centro del círculo.

3.- El plano Ángulo-Distancia se puede representar como un “manto” en el cual existirá un máximo local por cada círculo presente en la imagen. Este máximo corresponde al aporte unitario de cada par de puntos que determina un círculo con un mismo centro aproximado.

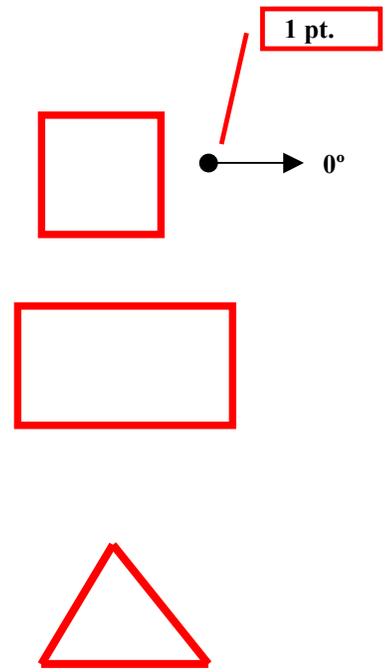
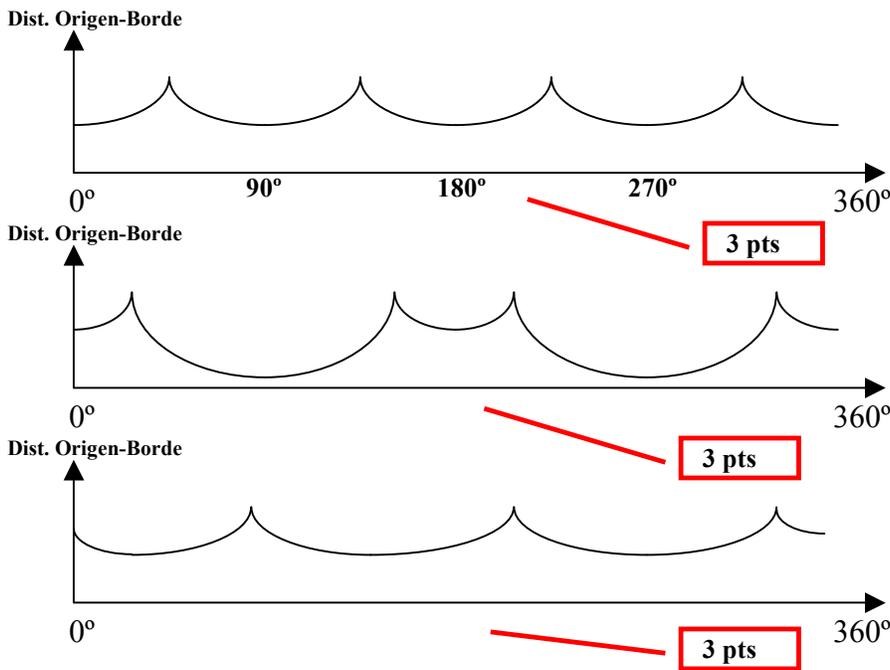
5 pts

b)

El algoritmo es muy robusto, siempre y cuando se controle el nivel de discretización del plano Angulo-Distancia, pues el aporte de diferentes pares de puntos define muchos círculos potenciales con centros relativamente cercanos. Para lograr un máximo en el plano de la transformada se debe disminuir la cuatización del plano de la transformada para agrupar todos los círculos muy cercanos en un sólo círculo representativo.

El algoritmo descrito no permite distinguir círculos concéntricos. Para ello se debe considerar un plano de la transformada que contemple el eje Ángulo, el eje Distancia al centro y el eje Radio del círculo.

5.- Dibuje cualitativamente la “firma” de un rectángulo, un cuadrado y un triángulo.



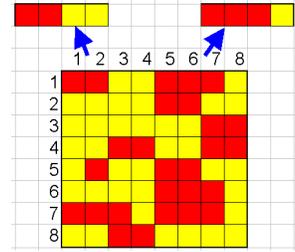
6.-“Los Algoritmos genéticos se entrenan en base a ejemplos” Comente.

La palabra “entrenamiento” no es la más indicada en el contexto de los algoritmos genéticos. Los algoritmos genéticos seleccionan los mejores elementos según alguna función de premio establecida. Esa función de premio puede ser justamente el error de la respuesta respecto de ciertos ejemplos, pero las operaciones que se ejecutan sobre los elementos (cromosomas) son:

- .- la mutación, la cual no corresponde a un aprendizaje;
- .- el cruzamiento, lo que busca la probabilidad de que los descendientes de dos elementos hereden buenas características;
- .- y la selección, la cual si depende de la función premio y en consecuencia de los posibles ejemplos.

5 pts

7.- Suponga la siguiente imagen binaria de 8x8 píxeles. Considere grupos de 4 píxeles consecutivos como se muestra en los dos ejemplos de la figura. Utilice método Huffman para codificar los 16 grupos (bloques de 4 píxeles) que posee la imagen. ¿Cuántos bits se deberán transmitir?



**Bloques:**

1100=A = 3

1110=B = 4

0000=C = 4

0011=D = 4

0100=E = 1

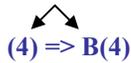
E(1) => A(3) => B(4) => C(4) => D(4)

(4) => B(4) => C(4) => D(4)



E(1) => A(3)

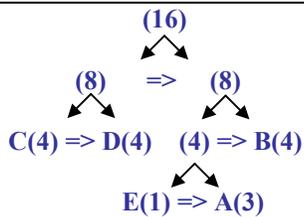
C(4) => D(4) => (8)



E(1) => A(3)

(8) => (8)  
C(4) => D(4) (4) => B(4)

E(1) => A(3)



5 pts

C=00

D=01

B=11

E=100

A=101

2 pts

A = 3 \* 3bits = 9 bits

B = 4 \* 2bits = 8 bits

C = 4 \* 2bits = 8 bits

D = 4 \* 2bits = 8 bits

E = 1 \* 3bits = 3 bits

3 pts

Total a transmitir = 36 bits de un total de 64 bits.

Tabla = (4+2) + (4+2) + (4+2) + (4+3) + (4+3) = 18+14 = 32

Se observa que en este caso la tabla más los datos suman 68 bits. Por lo tanto se puede afirmar que la codificación resulta eficiente sólo cuando existen algunos elementos que se repiten muchas veces

8.- En el caso de las imágenes, existen tres maneras de reducir el número de datos redundantes. Comente (1 línea c/u)

3 pts

En el caso de las imágenes, existen tres maneras de reducir el número de datos redundantes:

- eliminar código redundante: minimizar el uso de símbolos para representar la información (ej. Huffman)
- eliminar píxeles redundantes: codificar partes de la imagen o secuencias de píxeles, en base a píxeles vecinos (ej. RLC)
- eliminar redundancia visual: eliminar los datos que el ojo humano no es capaz de distinguir (ej. JPG)

3 pts

4 pts

9.- Se desea programar una red neuronal para “reconocer” objetos circulares de objetos cuadrados. Suponga imágenes de 32x32 píxeles. Suponga un objeto por imagen, relativamente centrado. ¿Qué tipo de red utilizaría? ¿Cuántos nodos posee la red en sus respectivas capas? ¿Es un problema lineal o no lineal?

3 pts

a) Perceptrón de una neurona.

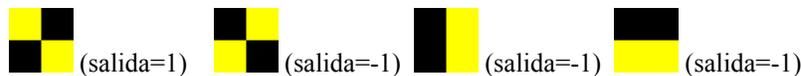
3 pts

b) Capa de entrada = 32x32=1024 nodos. Capa de salida= 1 neurona.

c) Es un problema lineal. Es decir las entradas son linealmente independientes. Pues no se puede construir un círculo en base a un cuadrado. En otras palabras existen píxeles que siempre difieren, como las esquinas. Por lo tanto, la red después del entrenamiento, probablemente potenciará los pesos de los nodos de entrada asignados a las esquinas. Además atenuará los pesos del centro, pues ambas clases de ejemplos poseen similitud en esos píxeles.

4 pts

10.- Programe un perceptrón (1 neurona), para “reconocer” líneas con pendiente 45° (como el patrón número 1). Realice las 2 primeras iteraciones (ej.1 y ej2), actualizando los pesos manualmente. Al inicio, suponga todos los pesos en 1.



Ej. 0110=1, 1001=-1

Ej. 0110 | 1 ==> 2+1>0 ==>1 ==> BIEN

Ej. 1001 |-1 ==> 2+1>0 ==>1 ==> MAL

ITERACION 1

$$\Delta w_i = d(x) x_i$$

$$w_1 = w_1 + \Delta w_1 = 1 + (-1) * 1 = 0$$

$$w_2 = w_2 + \Delta w_2 = 1 + (-1) * 0 = 1$$

$$w_3 = w_3 + \Delta w_3 = 1 + (-1) * 0 = 1$$

$$w_4 = w_4 + \Delta w_4 = 1 + (-1) * 1 = 0$$

$$w_0 = w_0 + \Delta w_0 = 1 + (-1) * 1 = 0$$

5 pts

Ej. 1010 |-1 ==> 1+0>0 ==>1 ==> MAL

ITERACION 2

$$w_1 = w_1 + \Delta w_1 = 0 + (-1) * 1 = -1$$

$$w_2 = w_2 + \Delta w_2 = 1 + (-1) * 0 = 1$$

$$w_3 = w_3 + \Delta w_3 = 1 + (-1) * 1 = 0$$

$$w_4 = w_4 + \Delta w_4 = 0 + (-1) * 0 = 0$$

$$w_0 = w_0 + \Delta w_0 = 0 + (-1) * 1 = -1$$

5 pts