

PAUTA PRUEBA N°2

Nombre:_____ ROL:_____ Firma:_____ Fecha: 05-12-2007

1.- [5%] Indique los pasos de un algoritmo genético básico.

Paso 0: se crea una población aleatoria de soluciones
Paso 1: se evalúa cada solución y se asigna una puntuación
Paso 2: Selección natural: se escoge las soluciones con mejor puntuación
Paso 3: Mutación y crossover: se realizan mutaciones y entrecruces en la población seleccionada para reconstruir las soluciones faltantes
Paso 4: Se vuelve al paso 1 hasta que se haya encontrado la solución deseada.

2.- [5%] ¿Cómo se "entrena" un Perceptrón?, Indique la regla de actualización de pesos.

Paso 0: Se inicializan los pesos y el umbral de forma aleatoria
Paso 1: Se selecciona un ejemplo X del conjunto de entrenamiento
Paso 2: Si la salida Y es errónea se modifican los pesos $w_i=w_i+(-1) \cdot Y \cdot x_i$ (en la dirección contraria a la salida Y)
Paso 3: Si no se ha cumplido el criterio de finalización, entonces volver al paso 1

3.- [5%] ¿Qué son los descriptores de Fourier? ¿Para qué sirven?

Corresponden a los coeficientes de la transformada de Fourier Unidimensional de la función de borde x,y de un objeto, donde cada punto del borde x,y se considera como un número complejo $p=x+jy$
Si se consideran sólo los primeros coeficientes de la transformada ("pasa bajos") se está considerando los aspectos principales de la forma lo que facilita la comparación y reconocimiento de patrones

4.- [5%] ¿Qué son los códigos de cadena? ¿Cómo se normalizan en longitud?

Corresponden a una codificación del ángulo entre píxeles vecinos de un borde. Para comparar códigos de cadena de objetos de diferente tamaño es posible normalizar la longitud del código tomando la "moda estadística" de un cierto número de elementos del código.

5.- [5%] Calcule la Entropía de la secuencia:
255, 255, 255, 255, 254, 253, 254, 253

$P(255)=4/8$
 $P(254)=2/8$
 $P(253)=2/8$
Entropía= $-4/8\log_2(4/8)-2/8\log_2(2/8)-2/8\log_2(2/8)=.5+.5+.5=1.5$

$$H = - \sum_{i=1}^I P(a_i) \log_2 P(a_i)$$

Ayuda: $\log_2(1/8)=-3$ $\log_2(2/8)=-2$ $\log_2(4/8)=-1$

6.- [5%] Indique la diferencia entre las técnicas de compresión basadas en "eliminación de píxeles redundantes" y las basadas en "eliminación de redundancia visual"

a) Píxeles redundantes: cuando existen semejanzas o correlaciones entre píxeles vecinos es posible codificar un conjunto de píxeles en base a un subconjunto de ellos
b) en cambio la redundancia visual consiste en reducir o eliminar la información menos sensible o imperceptible a la visión humana.

7.- [5%] Indique la diferencia entre un filtro de color con distancia de Mahalanobis y uno con distancia Euclídiana. ¿Cuál es mejor? ¿Por qué?

El filtro con distancia euclidiana genera una esfera en el espacio de colores entorno al color que se desea filtrar. En cambio si se utiliza la distancia de Mahalanobis se genera una vecindad de forma elíptica entorno al color objetivo, y cuyo eje principal está en línea con el color del matiz o color predominante. Es mejor el filtro de Mahalanobis. Por ejemplo si se desea filtrar colores rojizos el filtro será más amplio en la dirección del eje rojo del espacio de colores. Esto genera mejores resultados.

8.- [5%] ¿Por qué la transformada Hough es robusta incluso en imágenes mal binarizadas?

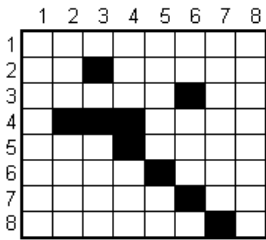
La transformada Hough realiza una especie de "encuesta" respecto de cuántos pares de píxeles pertenecerían a un cierto objeto (línea, círculo, etc) ubicado a cierta distancia del origen. Dado lo anterior no importa si los objetos están discontinuos o "pegados" unos con otros, lo importante es si existe la suficiente cantidad de pares de píxeles para destacar un cierto objeto respecto del ruido de fondo.

9.- [10%] Indique (en palabras) los pasos necesarios para implementar un algoritmo de "Firmas" o "Signaturas" sobre un objeto binarizado.

Paso 1: Obtener el centroide del objeto (etiquetarlo si es necesario para diferenciarlo de otros posibles objetos)
Paso 2: Posicionarse en el borde del objeto. Calcular y almacenar distancia y ángulo al centroide.
Paso 3: Marcar píxel del borde como "analizado".
Paso 4: Mientras existan píxeles del borde sin "analizar" avanzar al píxel vecino del borde y repetir paso 2.
FIN: Graficar arreglo ANGULO-DISTANCIA

10.- [20%] Escriba el pseudocódigo del siguiente algoritmo de compresión RLE (Run Length Encoding): i.- Codifica cada píxel con 1 byte. ii.- El número de veces que se repite un píxel se codifica con un número negativo (se utiliza como "separador") iii.- Los píxeles sin repetición consecutiva no se codifican. En la figura de ejemplo el resultado inicial sería:

(-10)(255)(0)(-10)(255)(0)(-3)(255)(-3)(0).....



[nf nc]=size(IM);	for x=2:L
A=zeros(nf*nc,1);	if A[x-1]==A[x]
x=0;	n=n+1;
for f=1:nf	else
for c=1:nc	if n>1, RLE=[RLE -1*n];end;
x=x+1;	RLE=[RLE A(x-1)];
A(x)=IM(f,c);	n=1;
end;	end;
end;	end;
L=x;	%Agrega el último
n=1;	if n>1, RLE=[RLE -1*n];end;
RLE=[];	RLE=[RLE A(x)];

11.- [15%] Programe la operación morfológica ERODE para un elemento estructurante 3x3

[nf nc]=size(A);	
K=ones(3);	Deja en "1" el centro f,c de cada iteración
B=A*0;	sólo si todos los píxeles del elemento
for f=1+1:nf-1	multiplicado por la imagen son igual a 1.
for c=1+1:nc-1	Es decir deja el "centro" si el elemento
s=0;	está completamente "contenido" en la
for ff=-1:1	imagen para cada iteración
for cc=-1:1	
s=s+K(ff+2,cc+2)*A(f+ff,c+cc)/255;	
end;	
end;	
if s=9, B(f,c)=255;	
else B(f,c)=0;	
end;	
end;	
end;	

12.- [15%] Suponga que los siguientes 24 números (byte) corresponden a los píxeles de una imagen. Determine la razón de compresión que se logra al aplicar la codificación Huffman

255	255	255	254	254	255	
255	254	254	254	253	254	
255	254	254	254	253	253	
255	255	254	254	253	252	

A=255=8

B=254=11

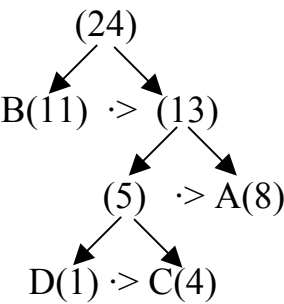
C=253=4

D=252=1

D(1) ·> C(4) ·> A(8) ·> B(11)

(5) ·> A(8) ·> B(11)
D(1) ·> C(4)

B(11) ·> (13)
(5) ·> A(8)
D(1) ·> C(4)



A=255=11

B=254=0

C=253=101

D=252=100

Sin compresión= 24 byte= 192 bits
Con compresión = 8*2+11*1+4*3+1*3 = 42 bits
Tabla = 4byte*2 = 64 bits (se puede mejorar)
Razón sin tabla= 192:42
Razón con tabla= 192:106