

PRUEBA N°1 (20-10-2008)

La prueba tiene 120 pts, sin embargo se asignará nota 100 a quien logre 100 o más pts. En otras palabras puede "eliminar" preguntas por un equivalente a 20 puntos.

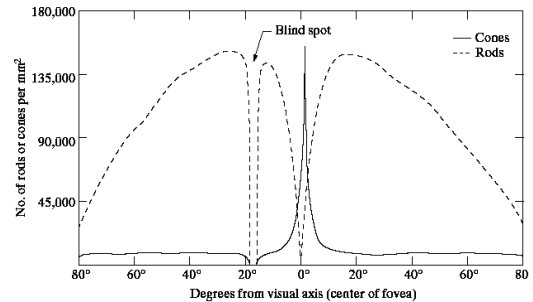
1.- [4 pts] A pesar que la cantidad de conos es menor a la cantidad de bastones... (ver gráfico) ¿por qué vemos claramente en colores? Comente en 3 líneas.

Respuesta 1

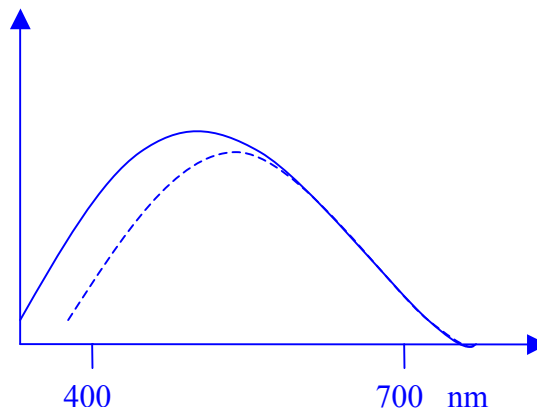
Vemos en colores no por la cantidad de los conos, sino por su especificidad respecto a la sensibilidad a longitudes de onda en diferentes rangos (R, G, y B)

Respuesta Alternativa

En otras palabras, para ver en colores se requieren 2 o más fotorreceptores sensibles a rangos de longitud de onda distintos.



2.- [4 pts] Suponga una especie animal que no posee capacidad de visión en colores, pero tiene una sensibilidad espectral superior en el rango cercano al ultravioleta y normal en el rango de luz visible. Presente un gráfico estimado de la sensibilidad espectral de sus bastones. (indique claramente el eje de los nanómetros).



3.- [4 pts] ¿Cuánto pesa (en megabyte) una imagen gris de 1024x1024, pero de alta resolución en frecuencia (65536 niveles = 2^{16})?

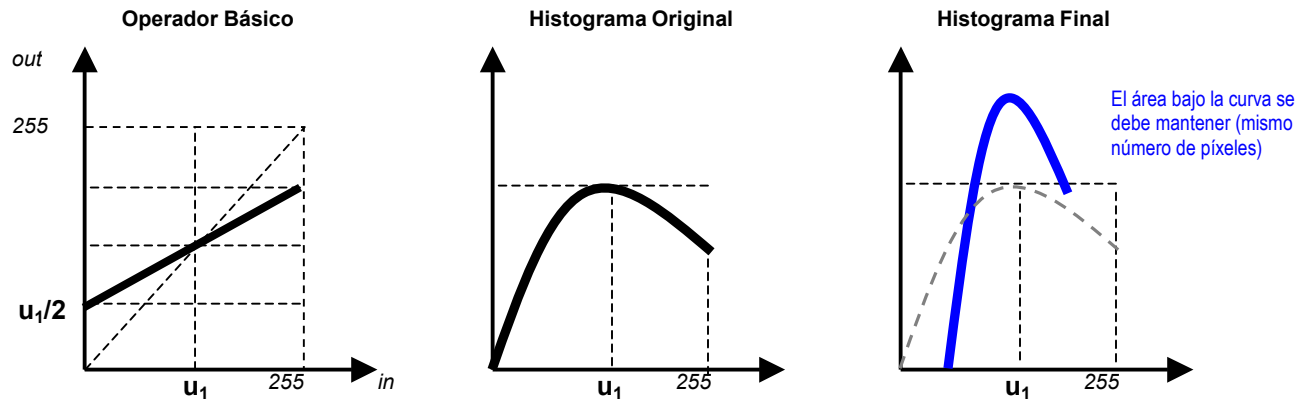
Una imagen de 1024x1024 contiene 1Mega píxeles. Dado que posee resolución en frecuencia (niveles de gris) de 16 bit, entonces cada píxel requiere de 2 bytes. Por lo tanto la imagen pesa 2 [Mega Bytes].

4.- [4 pts] a) Indique la diferencia entre la operación morfológica "open" y "dilatación". b) ¿Tienen el mismo efecto?

a) La operación OPEN corresponde a aplicar en secuencia una operación ERODE y DILATE. La operación DILATE permite dilatar los objetos binarizados en base a la aplicación de un AND respecto del centro de un elemento estructurante.

b) En términos simples la operación OPEN permite "separar" o "cortar" objetos que están levemente unidos. Esto ocurre pues al erosionar, los objetos se separan, pero al dilatar no se recupera completamente la erosión generada. Por el contrario el efecto de dilatar es "juntar" o "pegar" objetos cercanos, además del efecto obvio, que es dilatar los objetos binarizados.

5.- [4 pts] Dibuje (según las referencias indicadas), cómo sería el “Histograma Final”, luego de aplicar el “Operador Básico” a una imagen con un “Histograma Original” como el presentado en la figura.



6.- [15 pts] Escriba el pseudo-código del filtro denominado "Mediana 2.0", que consiste en asignar el valor de la mediana de la vecindad, sólo si la diferencia entre la máximo y el mínimo es mayor a K . De lo contrario se asigna el valor promedio de la vecindad.

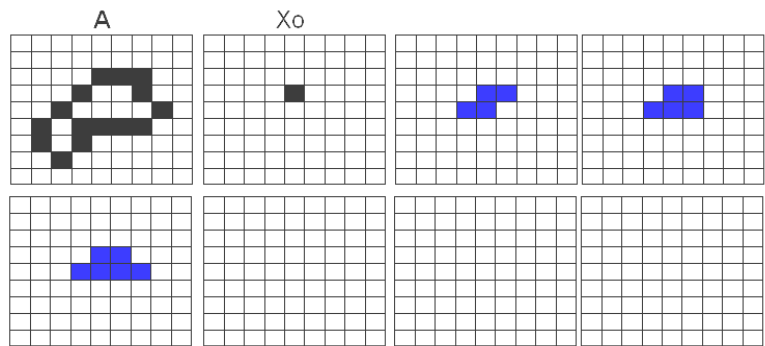
%Supone vecindad 3x3

```
[nf nc]=size(A)
v=zeros(1,9);
for f=1+1:nf-1
    for c=1+1:nc-1
        max=0;
        min=255;
        x=0;
        sum=0;
        for ff=-1:1
            for cc=-1:1
                if A(f+ff,c+cc)>max, max= A(f+ff,c+cc);end;
                if A(f+ff,c+cc)<min, min= A(f+ff,c+cc);end;
                x=x+1;
                v(x)=A(f+ff,c+cc);
            end;
        end;
        prom=sum/x;
        if (max-min)>K,
            %calcula mediana
            for x=1:8
                for y=x+1:9
                    if v(y)<v(x),
                        aux=v(x);
                        v(x)=v(y);
                        v(y)=aux;
                    end;
                end;
            end;
            A(f,c)=v(5); %Asigna mediana
        else
            A(f,c)=prom;
        end;
    end;
end;
```

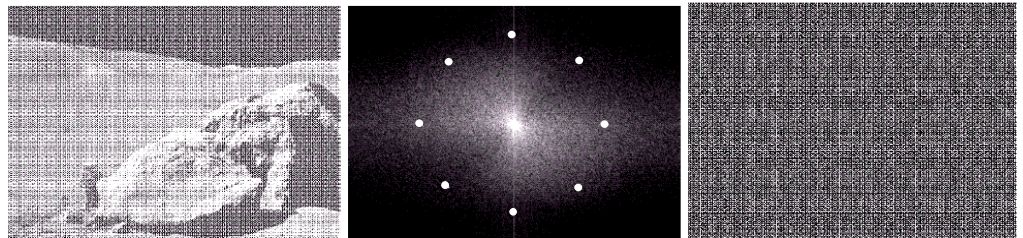
7.- [10 pts] La ecuación presenta una operación morfológica recursiva que corresponde a una dilatación de la imagen X_{k-1} y una intersección con el complemento de la imagen A. Dibuje todos los pasos de esta secuencia hasta que los cambios entre X_{k-1} y X_k se detengan.

Suponga $B=[0 \ 1 \ 0; 1 \ 1 \ 1; 0 \ 1 \ 0]$

$$X_k = (X_{k-1} \oplus B) \cap A^c$$



8.- [10 pts] En la figura se presentan 3 imágenes. La imagen A corresponde a una imagen de la NASA que posee un tipo de ruido con fuertes componentes sinusoidales. La imagen B corresponde al espectro de Fourier de la imagen A.



Programe un filtro “pasa banda” y aplíquelo sobre la imagen A para obtener de esta forma la imagen C.

Suponga que los 8 peaks de ruido sinusoidal están a una frecuencia Fr y la banda requerida para pasar dichos peaks tienen un ancho de “ dFr ”.

Puede utilizar las funciones pre-programadas de Matlab: *fft2*, *fftshift*, *ifft2* e *ifftshift*.

```
fc1=Fr-dFr/2;
fc2=Fr+dFr/2;
F=fft2(A);
F=fftshift(F);
[nf nc]=tamaño(F);
F2=zeros(nf,nc); %matriz nula
Para1 y desde 1 hasta nf
    Para2 x desde 1 hasta nc
        dist=(x-xc)*(x-xc)+(y-yc)*(y-yc);
        Si (dist>=fc1*fc1) y (dist<=fc2*fc2) entonces F2(y,x)=F(y,x);
    Fin2
Fin1
F2=ifftshift(F2);
D=real(ifft2(F2));
```

9.- [10 pts] Aplique el filtro laplaciano (básico: sin diagonales y con centro -4) y obtenga una imagen de borde. (Realice el procedimiento partiendo desde el pixel [2,2] hasta el pixel [4,4], para evitar operaciones de "padding")

5	5	7	7	7
0	5	5	7	7
0	0	5	7	7
0	0	0	5	5
0	0	0	0	5

-10	4	-2
10	-8	-4
0	10	-8

10.- [15 pts] Suponga una imagen de tamaño $N \times N$, y un filtro de tamaño 5×5 . La complejidad $O()$ de una operación de filtrado espacial mediante convolución es $O(N \cdot N \cdot 5 \cdot 5)$. La complejidad de la transformada rápida de Fourier (FFT) es $O(N \cdot N \cdot \log_2(N \cdot N))$. La complejidad de la transformada rápida de Fourier inversa (iFFT) es igual a la FFT. El proceso de filtrado en el plano de frecuencias se basa en una operación de FFT, luego una multiplicación de una máscara de tamaño $N \cdot N$, y finalmente una operación iFFT. Determine el valor de N a partir del cual es más rápido realizar "filtrado espacial" que "filtrado en frecuencia".

$$N \cdot N \cdot 5 \cdot 5 < N \cdot N \cdot \log_2(N \cdot N) + N \cdot N + N \cdot N \cdot \log_2(N \cdot N)$$

$$25 \cdot N^2 < 4 \cdot N^2 \cdot \log_2(N) + N^2$$

$$24 \cdot N^2 < 4 \cdot N^2 \cdot \log_2(N)$$

$$24 < 4 \cdot \log_2(N)$$

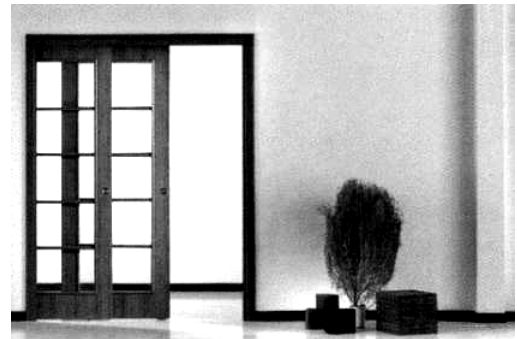
$$6 < \log_2(N)$$

$$2^6 < N$$

$N=64$ % para N mayor a 64 el filtrado espacial utilizando núcleo de 5×5 posee menor complejidad que el filtrado en frecuencia.

11.- [15 pts] En una aplicación de seguridad, se desea programar un algoritmo de procesamiento digital de imágenes, tal que active una alarma sólo en el caso que la puerta que se observa en la imagen se abra. Sin embargo la aplicación no debe activar la alarma si por el pasillo exterior circulan personas sin abrir la puerta.

Programe (en pseudo código) la aplicación completa. Suponga que la cámara está fija. La aplicación al iniciarse puede solicitar la interacción del usuario, por ejemplo puede solicitar que el usuario haga un click sobre la esquina superior izquierda e inferior derecha del marco de la puerta.



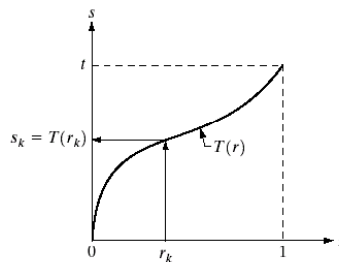
```
%PSEUDOLENGUAJE
A = CAPTURA_IMAGEN
ROI = REGION_Seleccionada_Por_Usuario
B = ROI(A)
MASK = BINARIZA(B)
MASK = NOT MASK
CICLO
    A = CAPTURA_NUEVA_IMAGEN
    B = ROI(A)
    ACTUAL = B .* MASK
    error = sum(sum(ACTUAL-ANTERIOR));
    if error>umbral, ALARMA!!!!
    ANTERIOR = ACTUAL
FIN CICLO
```

```
clear; colormap(gray(256)); A=double(imread('puerta.bmp'));
subplot(2,2,1);imshow(A, []);
disp('haga click sobre la esquina superior izquierda de la imagen'); [c1 f1]=ginput(1);
disp('haga click sobre la esquina superior izquierda de la imagen'); [c2 f2]=ginput(1);
c1=round(c1);c2=round(c2); f1=round(f1);f2=round(f2);

B=A(f1:f2,c1:c2);
[nf nc]=size(B);
MASK=B;
A2=double(imread('puerta2.bmp'));
subplot(2,2,2);imshow(A2, []);
umb=128;
for f=1:nf
    for c=1:nc
        if B(f,c)<umb, MASK(f,c)=1;else MASK(f,c)=0;end;
    end;
end;
subplot(2,2,3);imshow(MASK, []);
```

```
disp('presione control-c para terminar');
ANTERIOR=C; C=B;
sensibilidad=10; %(niveles de gris)
while 1==1
    %A2=AQUI SE DEBE CAPTUR NUEVA IMAGEN
    B = A2(f1:f2,c1:c2);
    for f=1:nf      %APLICA MASCARA
        for c=1:nc
            C(f,c)=B(f,c)*MASK(f,c);
        end;
    end;
    error=0;
    for f=1:nf      %CALCULA DIFERENCIAS
        for c=1:nc
            if abs(ANTERIOR(f,c)-C(f,c))>sensibilidad, error=error+1;end;
        end;
    end;
    subplot(2,2,4);imshow(C,[0 255]);
    if error/(nc*nf)>0.05, title('ALARMA!!!!');else title('Normal');end; %alarma si dif. > 5% de pixeles
    ANTERIOR = C;%guarda imagen actual para comparar en el siguiente ciclo
    pause;
end;
```

12.- [10 ptos] Programe el algoritmo clásico de "ecualización" de histogramas, cuya función de transformación T está dada por la siguiente fórmula:



$$T(r_k) = \sum_{j=1}^k p_r(r_j) = \sum_{j=1}^k \frac{n_j}{n}$$

k: es el número de niveles de gris
 p_r : es la probabilidad de un cierto nivel de gris
 r_j : es un nivel de gris "j" específico
 n_j : es el número de píxeles con nivel "j"
 n: es el total de píxeles

```
[nf nc]=size(IMAGEN) ;
H=zeros(1,256) ; %VECTOR PARA HISTOGRAMA
T=zeros(1,256) ; %VECTOR PARA TRANSFORMACION (ECUALIZACION)
```

```
%CALCULA HISTOGRAMA
for f=1:nf
    for c=1:nc
        H(IMAGEN(f,c)+1)=H(IMAGEN(f,c)+1)+1;
    end;
end;
```

```
%CALCULA TRANSFORMACION T
T(1)=H(1) ;
for i=2:256
    T(i)=T(i-1)+H(i) ;
end;
T=round(255*T/(nf*nc)) ;
```

```
%APLICA TRANSFORMACION T A LA IMAGEN
for f=1:nf
    for c=1:nc
        IMAGEN(f,c)=T(IMAGEN(f,c)+1) ;
    end;
end;
```

13.- [10 ptos] Dada la siguiente imagen de 4x8 píxeles (byte). Aplique el algoritmo de Huffman para recodificar la imagen y determine los bit por píxel (bpp) promedio resultante.

255	128	128	255
128	255	255	128
192	255	255	192
255	216	216	255
255	128	128	255
128	255	255	128
192	255	255	192
255	216	216	255

```
%Primero se calcula el histograma
"255"=16
"216"=4
"192"=4
"128"=8

*****
(4) 192 (4) 216 (8) 128 (16) 255
*****
      (8) (8) 128 (16) 255
(4) 192 (4) 216
*****
      (16) (16) 255
      (8) (8) 128
(4) 192 (4) 216
*****
      (32)
      (16) (16) 255
      (8) (8) 128
(4) 192 (4) 216
*****
%Asignando nuevos códigos de prefijo...
"255"=1
"128"=01
"216"=001
"192"=000
*****
%Cálculo de los nuevos bits por píxeles de la imagen(sin considerar la tabla)
"255"=16*1
"216"=4*3
"192"=4*3
"128"=8*2
Total=16+4*3+4*3+8*2=56 bit ==> 56/32 = 1.75 bpp
```

14.- [5 ptos] Para la misma imagen de la pregunta anterior, Calcule la Entropía:

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

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

```
*****
%Cálculo de la entropía
P("255")=16/32 = 1/2
P("216")=4/32=1/8
P("192")=4/32=1/8
P("128")=8/32=1/4

Entropía = H =
- P(255) log2(P(255)) - P(216) log2(P(216)) - P(192) log2(P(192)) - P(128) log2(P(128))
H= 1/2*1 + 1/8*3 + 1/8*3 + 1/4*2 = 1.75
*****
```