Universidad Nacional Autónoma de México
Facultad de Ingeniería

Procesamiento Digital de Imágenes
Práctica 5 - Interpolación

Integrantes:
Cabrera Sánchez Manuel Salvador
Torres Ruiz Mateo Alberto
Velázquez Sánchez José Antonio

1. Objetivos:

- Dada una imagen con un número de muestras igual a NxM, donde N y M son el ancho y el largo de la imagen respectivamente:
- Encontrar las imágenes interpoladas usando interpoladores de orden cero, lineal y cúbico.
- Interpolar la imagen en el dominio de la frecuencia.

2. Introducción

El problema de construir una función continua a partir de datos discretos es inevitable cuando estos datos deben ser manipulados de cierta manera que se necesita información no incluida explícitamente. Para resolver este problema, el esquema más utilizado es la interpolación que consiste en construir una función que aproxime de la manera más perfecta a la función original desconocida en los puntos de la medición.

La interpolación de imágenes es una operación muy importante usada en imagenología médica, procesamiento de imágenes y graficación por computadora. Existe una gran variedad de métodos de interpolación reportados en la literatura [1][2]. La interpolación de imágenes es necesaria en una gran variedad de situaciones como las que se mencionan a continuación:

1 .
Representarimágenesovolúmenesaunniveldeseadodediscretizaciónmodificandoparaellolatasademuestreo de los pixeles o voxels (voxel, elemento de volumen equivalente a pixel en 2D).

2. Cambiar la orientación de alguna rejilla de discretización.

3. Combinar la información sobre un mismo objeto desde múltiples modalidades en una sola imagen (fusión de imágenes). Por ejemplo, una resonancia magnética (MRI) y una Tomografía por Emisión de Positrones (PET) del cerebro de un paciente.

4. Cambio de rejilla de discretización, por ejemplo de polares a rectangulares.

Probablemente el uso más común de la interpolación sea la mencionada en el punto número 1, donde por lo regular se desean analizar ciertos detalles de una imagen a una escala y otros detalles a una escala más fina (zoom).

Algunas transformaciones pueden involucrar un cambio de coordenadas, por ejemplo, la función de conversión de coordenadas polares, adquiridas a través de un transductor de ultrasonido, a coordenadas cartesianas necesario para la visualización de la imagen en un monitor. En general, casi cualquier transformación geométrica sobre una imagen o un volumen necesita que se efectué una interpolación.

La calidad de la imagen o volumen obtenidos dependerá del proceso utilizado para realizar la interpolación así como también del trabajo necesario para que una computadora lo ejecute en un tiempo razonable.

3. Desarrollo

Para todos los puntos siguientes y con la finalidad de poder observar el desempeño de los distintos interpoladores usar una imagen nítida de baja resolución. Se recomienda usar imágenes desde 128x128 hasta 256x256 pixeles como máximo y la imagen pentagon que se encuentra disponible en la sección de imágenes del curso.
Para el desarrollo de esta práctica, utilizamos la imagen llamada "lena.jpg" de 256x256 pixeles en escala de grises y obtenida de internet.

1. Sobremuestro espacial:
- Obtenga el sobremuestreo de la imagen original insertando ceros entre los pixeles de la misma con factores T $\uparrow$=2×2 y T $\uparrow$=4×4.
- Obtenga la magnitud del espectro de la DFT (abs) de cada una de las imágenes sobremuestreadas y de la imagen original. Despliegue los resultados en una misma figura para efectos de comparación (en Matlab se puede usar el comando subplot). Recuerde usar fftshift para centrar los espectros y una función de escalamiento para el despliegue, ejemplo: ImFDespliegue=log(1.0 + ImF), donde ImF es la DFT de la imagen con sobremuestreo.

- **Explique el efecto del sobremuestreo espacial en el espectro de la DFT.**
El efecto que observamos en el espectro de la DFT de las imágenes, causado por el sobremuestreo espacial que realizamos al insertar ceros entre los pixeles de la matriz original, es que aumenta el número de puntos del espectro donde podemos observar los cambios en la intensidad de gris. Es decir, es como si alejáramos a la DFT original y observáramos que se repite en todo el dominio.

- **¿Qué sucedería si se intercalaran ceros entre los valores del espectro de la DFT?**
Probablemente al intercalar otros ceros entre los valores del espectro, podríamos cambiar de posición los puntos mencionados anteriormente, que es donde se nota la diferencia en las tonalidades de gris de la imagen.

2. Interpolación espacial. Interpole las imágenes con sobremuestreo obtenidas en el inciso anterior (con factores T $\uparrow$= 2×2 y T $\uparrow$= 4×4) usando interpoladores:
- De orden cero
- Lineal
- Cúbico

- **Para cada caso y factor de interpolación despliegue con alguna función de acercamiento (zoom) una mis- ma región seleccionada de las imágenes interpoladas. Compare y explique los resultados de los distintos interpoladores en una misma figura.**
Al comparar los distintos interpoladores para una imagen sobremuestreada con el mismo orden, notamos que el interpolado de orden cero hace que la imagen se vea más "pixeleada" que el interpolado lineal, debido a que el lineal realiza un promedio de los pixeles adyacentes y esto hace que la diferencia en niveles de gris entre pixeles no se note tanto.
En el caso de la interpolación lineal para la imagen sobremuestreada con factor 4x4, observamos que la imagen se ve oscura como si aún incluyera los ceros en la matriz, a pesar de ya haber aplicado el filtro.

- **Obtenga la magnitud del espectro de la DFT (abs) de cada una de las imágenes interpoladas. Despliegue los espectros en una misma figura para compararlos. Explique el efecto de los distintos tipos de interpolación en los espectros de las DFTs.**
  Al comparar la magnitud de los espectros de las DFT correspondientes, nos damos cuenta de que el interpolado lineal hace que las diferencias en los niveles de gris no sean tan notorias y, por lo tanto, la transición de tonos claros a oscuros en el espectro se ve más difuminada. Es decir, no se notan tanto las divisiones como en los espectros de las DFT interpoladas con orden cero.

3. Interpolación en frecuencia:
   - Obtenga la DFT de la imagen original y despliegue su magnitud.
   - Centre el espectro de la DFT (fftshift) y agregue ceros alrededor del mismo hasta completar y obtener dos espectros cuyas dimensiones correspondan a las dimensiones de la imagen original multiplicadas por los factores de interpolación $T \uparrow= 2 \times 2$ y $T \uparrow= 4 \times 4$ respectivamente.
   - Despliegue las magnitudes de las DFTs (abs) con ceros alrededor. Use una función de escalamiento para el despliegue.

   - **Compare en una misma figura las magnitudes de las DFTs (abs) de las imágenes interpoladas en el punto 2 con la magnitudes de las DFTs (abs) con ceros alrededor. Explique los resultados y diferencias.**
     Observamos, gracias a la manipulación anterior, imágenes con tamaño de orden 2x2 y 4x4, respectivamente, donde la DFT de la imagen original se haya en el cuadrante izquierdo superior. El resto de los pixeles en la imagen están en negro, debido a que rellenamos las matrices con ceros hacia la derecha y hacia abajo.

   - Calcule la inversa de la DFT (IDFT) de las DFTs con ceros alrededor.

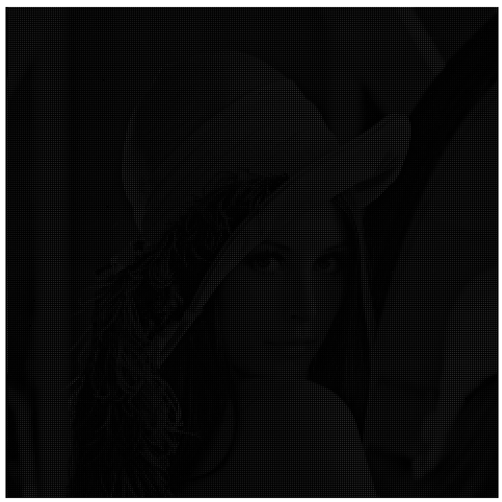
4. Compare en una misma figura los resultados de las imágenes interpoladas que se obtuvieron con los cuatro métodos.

4. Resultados

Imagen Original
(256 x 256)

Sobremuestreo de la imagen original:

Factor de 2x2



Factor de 4x4

Magnitud del espectro de la DFT:

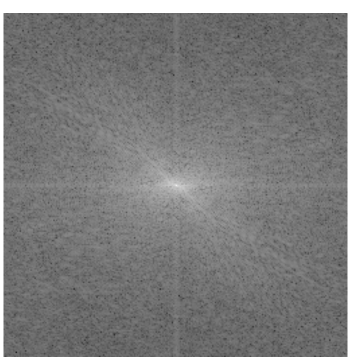
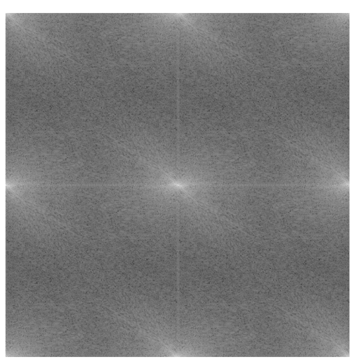
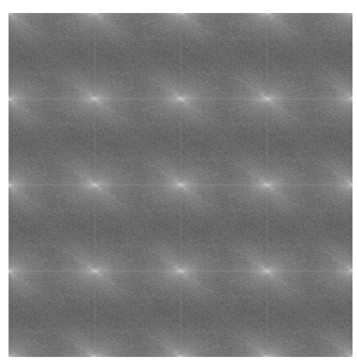


Imagen original



Factor de 2x2



Factor de 4x4

<u>Interpolación Espacial:</u>

Para el factor 2x2:

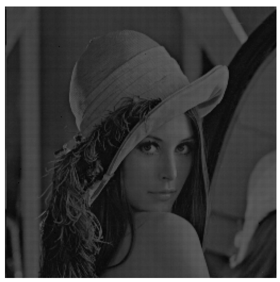
Interpolación de orden cero

Interpolación lineal

Interpolación cúbica

Para el factor 4x4:

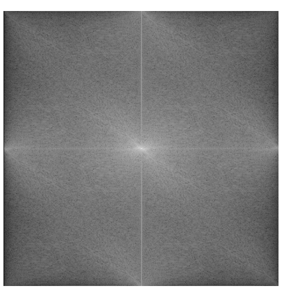
Interpolación de orden cero



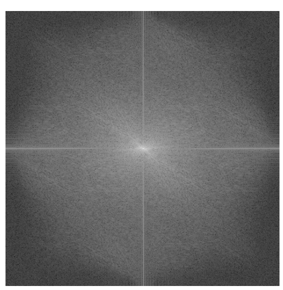
Interpolación lineal

Interpolación cúbica

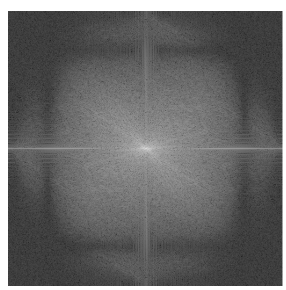
Magnitud del espectro de la DFT:



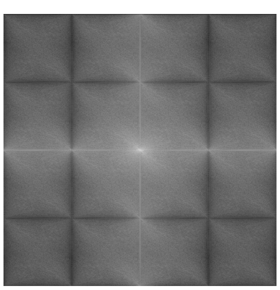
Factor 2x2
(interpolación cero)



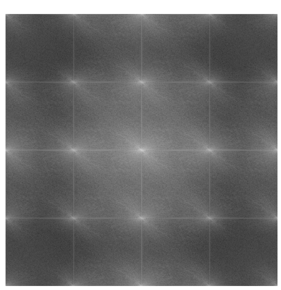
Factor 2x2
(interpolación lineal)



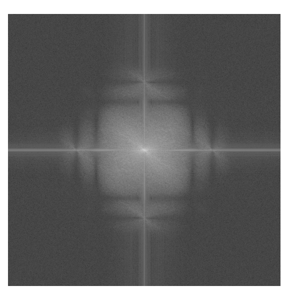
Factor 2x2
(interpolación cúbica)



Factor 4x4
(interpolación cero)



Factor 4x4
(interpolación lineal)



Factor 4x4
(interpolación cúbica)

Magnitud del espectro de la DFT:

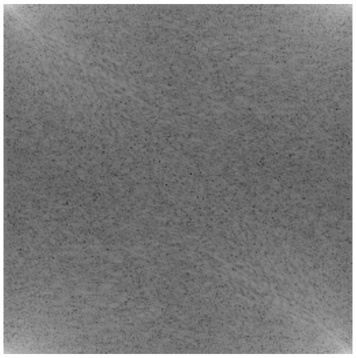
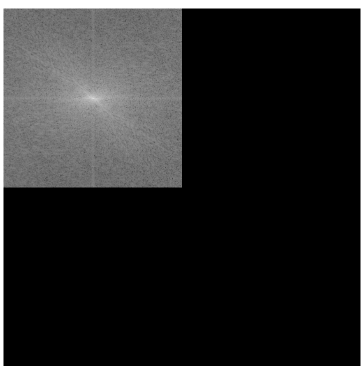
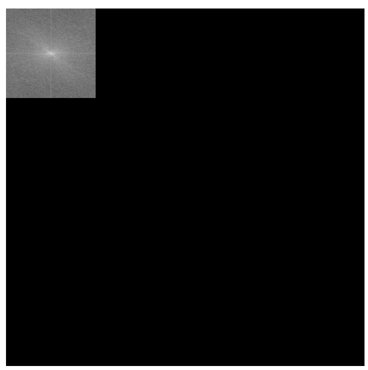


Imagen original
(sin Shift)

<u>Magnitud de las DFT con ceros:</u>



Orden 2x2



Orden 4x4

Transformada inversa de las DFT:

Orden 2x2

Orden 4x4

5.- Código

```
% Cargando imagen
clear
img = imread('lena.jpg');
imwrite(mat2gray(img,[0 255]),'lena.png');

%%%%% Ejercicio1 %%%%%

%Sobremuestreo 2x2
sobre2 = upsample(img,2);
sobre2 = rot90(sobre2);
sobre2 = upsample(sobre2,2);
sobre2 = rot90(sobre2,-1);
imwrite(mat2gray(sobre2),'1_sobre2.png');

%Sobremuestreo 4x4
sobre4 = upsample(img,4);
sobre4 = rot90(sobre4);
sobre4 = upsample(sobre4,4);
sobre4 = rot90(sobre4,-1);
imwrite(mat2gray(sobre4),'1_sobre4.png');

%Magnitud del espectro de DFT 2x2
sobre2_dft = fft2(sobre2);
sobre2_dft = abs(sobre2_dft);
sobre2_dft = log(sobre2_dft + 1);
sobre2_dft = fftshift(sobre2_dft);
imwrite(mat2gray(sobre2_dft),'1_sobre2dft.png');

%Magnitud del espectro de DFT 4x4
sobre4_dft = fft2(sobre4);
sobre4_dft = abs(sobre4_dft);
sobre4_dft = log(sobre4_dft + 1);
sobre4_dft = fftshift(sobre4_dft);
imwrite(mat2gray(sobre4_dft),'1_sobre4dft.png');

%Magnitud del espectro DFT original con
shift
img_dft = fft2(double(img));
img_dft = abs(img_dft);
img_dft = log(img_dft + 1);
img_dft_shift = fftshift(img_dft);
imwrite(mat2gray(img_dft_shift),'1_dft_shift.png'
);

%%%%% Ejercicio2 %%%%%

%Interpolacion de orden cero 2x2
filtro02 = ones(2,2);
inter_cero2 = imfilter(sobre2,filtro02);
imwrite(mat2gray(inter_cero2),'2_inter_cero2.png
');

%Interpolacion de orden cero 4x4
filtro04 = ones(4,4);
inter_cero4 = imfilter(sobre4,filtro04);
imwrite(mat2gray(inter_cero4),'2_inter_cero4.png
');

%Interpolacion lineal 2x2
filtro12 = [(1/4),(1/2),(1/4);
        (1/2),1,(1/2);
        (1/4),(1/2),(1/4)];
inter_lin2 = imfilter(sobre2,filtro12);
imwrite(mat2gray(inter_lin2),'2_inter_lin2.png');

%Interpolacion lineal 4x4
filtro14 = [(1/16),(1/8),(1/4),(1/8),(1/16);
        (1/8),(1/4),(1/2),(1/4),(1/8);
        (1/4),(1/2),1,(1/2),(1/4);
        (1/8),(1/4),(1/2),(1/4),(1/8);
        (1/16),(1/8),(1/4),(1/8),(1/16)];
inter_lin4 = imfilter(sobre4,filtro14);
imwrite(mat2gray(inter_lin4),'2_inter_lin4.png');

%Interpolacion cubica 2x2
filtro32 = fcubic(2);
inter_cub2 = imfilter(sobre2,filtro32);
imwrite(mat2gray(inter_cub2),'2_inter_cub2.png')
;

%Interpolacion cubica 4x4
filtro34 = fcubic(4);
inter_cub4 = imfilter(sobre4,filtro34);
imwrite(mat2gray(inter_cub4),'2_inter_cub4.png')
;

%Magnitud de los espectros de DFT 2x2
cero2_dft = fft2(inter_cero2);
cero2_dft = abs(cero2_dft);
cero2_dft = log(cero2_dft + 1);
cero2_dft = fftshift(cero2_dft);
imwrite(mat2gray(cero2_dft),’2_cero2dft.png');

lin2_dft = fft2(inter_lin2);
lin2_dft = abs(lin2_dft);
lin2_dft = log(lin2_dft + 1);
lin2_dft = fftshift(lin2_dft);
imwrite(mat2gray(lin2_dft),’2_lin2dft.png');
```

```
cub2_dft = fft2(inter_cub2);
cub2_dft = abs(cub2_dft);
cub2_dft = log(cub2_dft + 1);
cub2_dft = fftshift(cub2_dft);
imwrite(mat2gray(cub2_dft),'2_cub2dft.png');
```

%Magnitud de los espectros de DFT 4x4

```
cero4_dft = fft2(inter_cero4);
cero4_dft = abs(cero4_dft);
cero4_dft = log(cero4_dft + 1);
cero4_dft = fftshift(cero4_dft);
imwrite(mat2gray(cero4_dft),'2_cero4dft.png');

lin4_dft = fft2(inter_lin4);
lin4_dft = abs(lin4_dft);
lin4_dft = log(lin4_dft + 1);
lin4_dft = fftshift(lin4_dft);
imwrite(mat2gray(lin4_dft),'2_lin4dft.png');

cub4_dft = fft2(inter_cub4);
cub4_dft = abs(cub4_dft);
cub4_dft = log(cub4_dft + 1);
cub4_dft = fftshift(cub4_dft);
imwrite(mat2gray(cub4_dft),'2_cub4dft.png');
```

%%%%% Ejercicio3 %%%%%

%Magnitud del espectro DFT sin shift

```
imwrite(mat2gray(img_dft), '3_dft.png');
img_dft_shift = fftshift(fft2(img));
```

%Sobremuestreo de 2x2

```
img_dft_pad2 = padarray(img_dft_shift,[256
256],0,'post');
imwrite(mat2gray(log(abs(img_dft_pad2) + 1)),
'3_dft_pad2.png');
```

%Sobremuestreo de 4x4

```
img_dft_pad4 = padarray(img_dft_shift,[768
768],0,'post');
imwrite(mat2gray(log(abs(img_dft_pad4) + 1)),
'3_dft_pad4.png');
```

%Transformada inversa 2x2

```
inversa2 = ifft2(img_dft_pad2);
inversa2 = abs(inversa2);
imwrite(mat2gray(inversa2), '3_inversa2.png');
```

%Transformada inversa 4x4

```
inversa4 = ifft2(img_dft_pad4);
inversa4 = abs(inversa4);
imwrite(mat2gray(inversa4), '3_inversa4.png');
```

%%%%% Archivo fcubic.m %%%%%

```
function filtro = fcubic(T)
    x = -2 : 1/T : 2;
    x = x(2:end-1);
    y = zeros(size(x));
    for i = 1:size(x,2)
        if abs(x(i)) < 1
            y(i) = 1-2*abs(x(i))^2 + abs(x(i))^3;
        else
            y(i) = 4 - 8*abs(x(i)) + 5*abs(x(i))^2 -
abs(x(i))^3;
        end
    end
    filtro = y'*y;
end
```

6. Conclusiones

- Con el desarrollo de esta práctica pudimos observar claramente el comportamiento de los filtros interpoladores sobre una imagen, que son utilizados para agrandar la imagen.
- Comprendimos que cada filtro interpolados funciona con coeficientes diferentes y, por lo tanto, realiza efectos diferentes sobre la imagen. Por ejemplo, el filtro de orden cero consiste en una matriz llena de unos, lo cual copiará directamente el valor de los pixeles adyacentes y eso hace que la diferencia de tonos de gris sea mucho más notorio en la imagen resultante. En cambio, un filtro de orden lineal realiza un promedio de los pixeles adyacentes para rellenar la matriz resultante, de forma que los cambios en los niveles de gris no resultan tan notorios.
- También entendimos que aumentar el orden del filtro interpolador a utilizar, provoca que la magnitud del espectro de la DFT de la imagen original pueda verse varias veces en el espectro de las imágenes resultantes.
- Conocimos la importancia de algunos nuevos comandos en Matlab, tal como UPSAMPLE el cual nosotros aprovechamos para sobremuestrear la imagen original con ceros en una dirección, posteriormente rotarla y volver a sobremuestrear para obtener la matriz completa.

Referencias

[1] E. Meijeringa, A Chronology of Interpolation: From Ancient Astronomy to Modern Signal and Image Processing, Proceedings of the IEEE, Vol. 90, No. 3, 2002.

[2] E. Maeland, On the Comparison of Interpolation Methods, IEEE Transactions on Medical Imaging, Vol. 1, No. 3, 1988.

[3] Gonzalez, R. C. , and Woods, P., Digital Image Processing, Addison Wesley, 2002.