

EL-GY 6123 Image and Video Processing
Matlab Assignment 4

Name: Amitesh Kumar Sah
NYU ID: N19714360

Question 1

Write your own program or programs which can: a) Down sample an image by a factor of 2, without using a prefilter, and with using a filter you designed using MATLAB; b) Up-sample the previously down-sampled images by a factor of 2, using the bilinear interpolation and cubic interpolation methods, respectively. You should have a total of 4 interpolated images, with different combination of down-sampling and interpolation methods. Your program could either directly display on screen the processed images during program execution, or save the processed images as computer files for display after program execution. Run your program with the image Barbara. Comment on the quality of the down/up sampled images obtained with different methods. Note: you should not use the "resize" function in Matlab to do this

Matlab Program

```
clc;
clear all; close all;

% Reading an image
I=imread('barbara_gray.bmp');
% I=im2double(I);
[c,r]=size(I);
% Without using Pre-filter
% Downsampling by a factor of 2
% for i=1:c/2
%     for j=1:r/2
%         fd(i,j)=I(2*i,2*j);
%     end
% end

fd=I(1:2:r,1:2:c);
figure,
subplot(1,2,1),imshow(I),title('Original Image');
subplot(1,2,2),imshow(fd,[]),title('Downsample by 2 without using prefilter');

% With using Pre-filter and Downsampling by a factor of 2
%design a lowpass filter with cutoff at 1/2K and length N.
h=fir1(11, 1/2);
% Convolve the image with low pass filter
fp=conv2(I,h,'same');
gd=fp(1:2:r,1:2:c);
figure,
subplot(1,2,1),imshow(I),title('Original Image');
subplot(1,2,2),imshow(gd,[]),title('Downsample by 2 with using prefilter');

% Upsampling Using Bilinear Interpolation
% Without Prefilter while downsampling
[r1,c1]=size(fd);
B=zeros(2*r1,2*c1);
for m=1:r1-1
    for n=1:c1-1
        B(2*m,2*n)=fd(m,n);
```

```

B(2*m,2*n+1)=(fd(m,n)+fd(m,n+1))/2;
B(2*m+1,2*n)=(fd(m,n)+fd(m+1,n))/2;
B(2*m+1,2*n+1)=(fd(m,n)+fd(m,n+1)+fd(m+1,n)+fd(m+1,n+1))/4;
    end
end
figure,
subplot(1,2,1),imshow(fd,[],),title('Downsampled Without Prefiltering Image');
subplot(1,2,2),imshow(B,[],),title('Interpolation by 2 with using bilinear');

% Upsampling Using Bilinear Interpolation
% With Prefilter while downsampling
[r2,c2]=size(gd);
for m=1:r2-1
    for n=1:c2-1
        Bp(2*m,2*n)=gd(m,n);
        Bp(2*m,2*n+1)=(gd(m,n)+gd(m,n+1))/2;
        Bp(2*m+1,2*n)=(gd(m,n)+gd(m+1,n))/2;
        Bp(2*m+1,2*n+1)=(gd(m,n)+gd(m,n+1)+gd(m+1,n)+gd(m+1,n+1))/4;
    end
end
figure,
subplot(1,2,1),imshow(gd,[],),title('Downsampled With Prefiltering');
subplot(1,2,2),imshow(Bp,[],),title('Interpolation by 2 with using bilinear');

%Bicubic without prefiltering
for i=2:r2-2;
    for j=2:c2-2;
        Bc(2*i,2*j)=fd(i,j);
        Bc(2*i,2*j+1)=1/8*(-fd(i,j-1)+5*fd(i,j)+5*fd(i,j+1)-fd(i,j+2));
        Bc(2*i+1,2*j)=1/8*(-fd(i-1,j)+5*fd(i,j)+5*fd(i+1,j)-fd(i+2,j));
    end
end
figure,
subplot(1,2,1),imshow(fd,[],),title('Downsampled Without Prefiltering Image');
subplot(1,2,2),imshow(Bc,[],),title('Interpolation by 2 with using BiCubic');

%Bicubic with prefiltering
for i=2:r2-2;
    for j=2:c2-2;
        Bcp(2*i,2*j)=gd(i,j);
        Bcp(2*i,2*j+1)=1/8*(-gd(i,j-1)+5*gd(i,j)+5*gd(i,j+1)-gd(i,j+2));
        Bcp(2*i+1,2*j)=1/8*(-gd(i-1,j)+5*gd(i,j)+5*gd(i+1,j)-gd(i+2,j));
    end
end
figure,
subplot(1,2,1),imshow(gd,[],),title('Downsampled Without Prefiltering Image');
subplot(1,2,2),imshow(Bcp,[],),title('Interpolation by 2 with using BiCubic');

figure,
subplot(2,2,1),imshow(B,[],),title('Bilinear without prefiltering');
subplot(2,2,2),imshow(Bp,[],),title('Bilinear with prefiltering');
subplot(2,2,3),imshow(Bc,[],),title('BiCubic without prefiltering');
subplot(2,2,4),imshow(Bcp,[],),title('BiCubic with prefiltering');

```

Result

Original Image



Downsample by 2 without using prefilter



Original Image



Downsample by 2 with using prefilter



Downsampled Without Prefiltering Image



Interpolation by 2 with using bilinear



Downsampled With Prefiltering



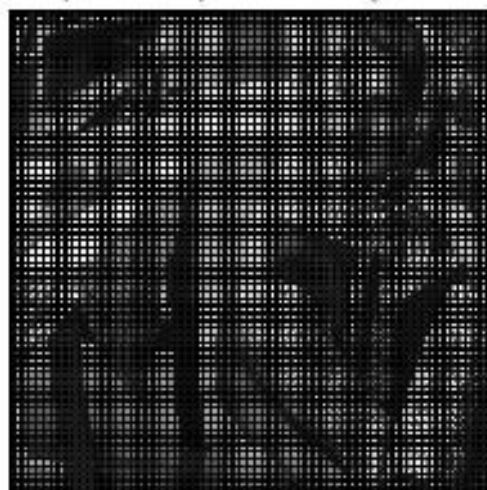
Interpolation by 2 with using bilinear



Downsampled Without Prefiltering Image



Interpolation by 2 with using BiCubic



Downsampled With Prefiltering Image



Interpolation by 2 with using BiCubic



Bilinear without prefiltering



Bilinear with prefiltering



BiCubic without prefiltering



BiCubic with prefiltering



Conclusion

By downsampling without prefiltering, there is more rough edges.

By downsampling with pre filtering, image is smoother.

Interpolation using Bilinear for downsampled image without prefiltering, reconstruction is not good,

Interpolation using Bilinear for downsampled image with prefiltering, reconstruction is good,

Interpolation using Biubicfor downsampled image without prefiltering, reconstruction is not good, and lose lot of information. Image is dark

Interpolation using Bicubic for downsampled image with prefiltering, reconstruction is average and its brighter.

Question 2

```
% Reading an image  
F=imread('fixed.jpg');  
M=imread('moving.jpg');
```

```

F1=rgb2gray(F);
M1=rgb2gray(M);

% figure,
% subplot(1,2,1),imshow(F1),title('FIXED Image');
% subplot(1,2,2),imshow(M1),title('MOVING Image');

% Select the corresponding feature point in these two images.
% cpselect(M1,F1);

% Affine Transformation represents a relation between two images.
one=ones(1,8);
A(:,1)=one';
A(:,2)=fixedPoints(:,1);
A(:,3)=fixedPoints(:,1);

x=movingPoints(:,1);
y=movingPoints(:,2);

% Here , K>N i.e. 8>3
% So we use least square solution
% a=inv((A'*A))*A'*x

tform=cp2tform(movingPoints,fixedPoints,'affine');
J=imtransform(F1,tform);

figure,
subplot(2,2,1),imshow(F1),title('FIXED Image');
subplot(2,2,2),imshow(M1),title('Moving Image');
subplot(2,2,3),imshow(J),title('Warping Image');

```

