EELE 5110 Digital Image Processing
Lab. 3

Effects of Changes in Spatial Resolution and Quantization on Gray Level Images

1. Objectives

1. To study the effect of changing the number of gray levels on the quality of images.
2. To study the effect of changing spatial resolution on the quality of images, using two methods:
   a) Nearest neighbor interpolation.
   b) Bilinear interpolation.

2. Theory

I) Changing the number of gray levels:

It is has been realized (both subjectively and objectively) that the quality of a gray-level image is dramatically affected by its gray-level resolution. Other words, increasing the number of bits per pixel has a great effect in improving the quality of gray-level images. This is because that a higher number of gray levels would give a smooth transition along the details of the image and hence improving its quality to the human eye.

In this experiment, the gray level resolution of an image is to be reduced from 256 to 2, each time by a factor of 2 (equivalently, reducing the number of bits per pixel from 8 to 1). The change of the image quality and the ***false contouring*** effect are to be observed. In MATLAB, this can be performed by reducing the number of gray levels that are required to display the image using a smaller number of rows in the colormap matrix. All of this will be performed while keeping a constant spatial resolution for the image (256 X 256).

II) Changing the spatial resolution:

Changing the spatial resolution of a digital image, by zooming or shrinking, is an operation of great importance in a wide range of applications (i.e. in digital cameras, biomedical image processing and astronomical images). Simply, zooming and shrinking are the operations of oversampling and undersampling a digital image, respectively. Zooming a digital image requires two steps: the creation of new pixel locations, and assignment of gray levels to those new locations. The assignment of gray levels to the new pixel locations is an operation of great challenge. It can be performed using two approaches:

1) *Nearest Neighbor Interpolation:* each pixel in the zoomed image is assigned the gray level value of its closest pixel in the original image.
2) *Bilinear Interpolation:* the value of each pixel in the zoomed image is a weighted average of the gray level values of the pixels in the nearest 2-by-2 neighborhood, in the original image.

In this experiment, the spatial resolution of a gray level image is to be reduced from 256 to 16, each time by a factor of 2. The smallest discernible detail is to be measured subjectively. To compare these shrinking effects, the reduced images are to be resized again to the original size (256 X 256), then, the images quality and the ***checkerboard effects*** are to be observed. In MATLAB, this can be performed using the *imresize* function. This function accepts a factor greater than 1.0 for zooming and smaller than 1.0 for shrinking. Also, it accepts the required method of interpolation, either nearest neighbor or bilinear. All of this will be performed while keeping a constant gray level resolution for the image (256 gray levels or 8 bits per pixel).

3. Assignments

Note: In all experimental parts, you will use only one image: Fig2_19_a, found in the folder matlab6p5/work.

Part 1: (Reducing the Number of Gray Levels of an Image)
Apply the following MATLAB code:

```
% Reading the image and converting it to a gray-level image.
I=imread('Fig2_19_a.jpg');
I=rgb2gray(I);

% A 256 gray-level image:
[I256,map256]=gray2ind(I,256);

% A 128 gray-level image:
[I128,map128]=gray2ind(I,128);

% A 64 gray-level image:
[I64,map64]=gray2ind(I,64);

figure(1),subplot(221),subimage(I),title('I'),subplot(222),subimage(I256,m
ap256),title('I256'),subplot(223),subimage(I128,map128),title('I128'),subp
lot(224),subimage(I64,map64),title('I64'),pause

% A 32 gray-level image:
[I32,map32]=gray2ind(I,32);

% A 16 gray-level image:
[I16,map16]=gray2ind(I,16);

% A 8 gray-level image:
[I8,map8]=gray2ind(I,8);

figure(2),subplot(221),subimage(I),title('I'),subplot(222),subimage(I32,ma
p32),title('I32'),subplot(223),subimage(I16,map16),title('I16'),subplot(22
4),subimage(I8,map8),title('I8'),pause

% A 4 gray-level image:
[I4,map4]=gray2ind(I,4);

% A 2 gray-level image:
[I2,map2]=gray2ind(I,2);

figure(3),subplot(221),subimage(I),title('I'),subplot(222),subimage(I8,map
8),title('I8'),subplot(223),subimage(I4,map4),title('I4'),subplot(224),sub
image(I2,map2),title('I2'),pause
```

COMMENT on the results you have obtained in Part 1.

Part 2: (Reducing the Spatial Resolution of an Image)

Apply the following MATLAB code:

```
close all
clear all

% Reading the image and converting it to a gray-level image.
I=imread('Fig2_19_a.jpg');
I=rgb2gray(I);

% Reducing the Size of I using nearest neighbor interpolation
I128=imresize(I,0.5); imshow(I128),pause
I64=imresize(I,0.25);close,imshow(I64),pause
I32=imresize(I,0.125);close,imshow(I32),pause
I16=imresize(I,0.0625);close,imshow(I16),pause

% Resizing the Reduced Images to the Original Size (256 X 256) and Comapre
% them:
I16=imresize(I16,16);
I32=imresize(I32,8);
I64=imresize(I64,4);
I128=imresize(I128,2);

figure,subplot(121),subimage(I),title('I'),subplot(122),subimage(I128),tit
le('I128'),pause,close

figure,subplot(221),subimage(I),title('I'),subplot(222),subimage(I64),titl
e('I64'),
subplot(223),subimage(I32),title('I32'),subplot(224),subimage(I16),title('
I16'),pause

close all

% Reducing the Size of I using bilinear interpolation
I128_b=imresize(I,0.5,'bilinear');I128_b=imresize(I128_b,2,'bilinear');
I64_b=imresize(I,0.25,'bilinear');I64_b=imresize(I64_b,4,'bilinear');
I32_b=imresize(I,0.125,'bilinear');I32_b=imresize(I32_b,8,'bilinear');
I16_b=imresize(I,0.0625,'bilinear');I16_b=imresize(I16_b,16,'bilinear');

subplot(241),subimage(I128),axis off,title('I128'),
subplot(242),subimage(I64),axis off,title('I64'),
subplot(243),subimage(I32),axis off,title('I32'),
subplot(244),subimage(I16),axis off,title('I16'),
subplot(245),subimage(I128_b),axis off,title('I128_b'),
subplot(246),subimage(I64_b),axis off,title('I64_b'),
subplot(247),subimage(I32_b),axis off,title('I32_b'),
subplot(248),subimage(I16_b),axis off,title('I16_b'),
```

COMMENT on the results you have obtained in Part 2.