Image Sampling & Quantization

Aim:

The main objectives of this experiment are:

1. To gain a clearer understanding of Image sampling and quantization.
2. To learn how to simulate these operations using Scilab.

Apparatus:

1. A PC installed with Scilab – 5.5.0.
2. A digital image camera or web camera – for image sampling and quantization experiment.

Theory:

To create a digital image, we need to convert continuous sensed data into digital form. This involves two processes: *sampling* and *quantization*

1. Sampling:

To sample a continuous-time signal x(t) is to represent x(t) at a discrete number of points $t = nT_s$ where T_s is the sampling period. The sampling theorem states that a band-limited x(t) with a bandwidth Wcan be reconstructed from its sample values $x(n) = x(nT_s)$ if the sampling frequency $f_s = 1/T_s$ is greater than twice the bandwidth W of x(t). Otherwise, aliasing would result in x(t). The minimum sampling rate of $2f_s$ for an analog band-limited signal is called the Nyquist rate.

Sampling corresponds to a discretization of the space. That is, of the domain of the function, into $f : [1,\ldots,N] \times [1,\ldots,M] \longrightarrow \mathbb{R}^m$. Figure 1(a) shows a continuous image, f (x, y), that we want to convert to digital form. A continuous image f(x,y) is normally approximated by equally spaced samples arranged in the form of an N x M array where each element of the array is a discrete quantity. To convert it to digital form, we have to sample the function in both coordinates and in amplitude. An image may be continuous with respect to the x- and y-coordinates' and also in amplitude. Digitizing the coordinate values is called *sampling*, Digitizing the amplitude values is called *quantization.* Finer the sampling (i.e., the larger N and M) the better the approximation of the continuous image function f(x,y).

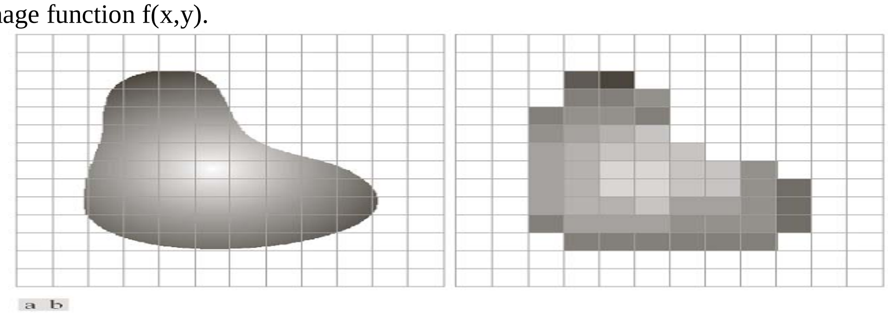


Figure 1: (a) Continuous image projected onto a sensor array..(b) Result of image sampling and quantization

Thus, the image can be seen as matrix,

$$f = \begin{bmatrix} f(1,1) & f(1,2) & \cdots & f(1,M) \\ f(2,1) & f(2,2) & \cdots & f(2,M) \\ \vdots & \vdots & \ddots & \vdots \\ f(N,1) & f(N,2) & \cdots & f(N,M) \end{bmatrix}.$$

The smallest element resulting from the discretization of the space is called a pixel (picture element). The number of bits required to store a digitised image is $b = M \times N \times k$. Where M & N are the number of rows and columns, respectively. The number of gray levels is an integer power of 2: $L = 2^k$ where k =1,2,…24. It is common practice to refer to the image as a "k-bit image"

Number of storage bits for various values of N and k.

N/k	1 (L = 2)	2 (L = 4)	3 (L = 8)	4 (L = 16)	5 (L = 32)	6 (L = 64)	7 (L = 128)	8 (L = 256)
32	1,024	2,048	3,072	4,096	5,120	6,144	7,168	8,192
64	4,096	8,192	12,288	16,384	20,480	24,576	28,672	32,768
128	16,384	32,768	49,152	65,536	81,920	98,304	114,688	131,072
256	65,536	131,072	196,608	262,144	327,680	393,216	458,752	524,288
512	262,144	524,288	786,432	1,048,576	1,310,720	1,572,864	1,835,008	2,097,152
1024	1,048,576	2,097,152	3,145,728	4,194,304	5,242,880	6,291,456	7,340,032	8,388,608
2048	4,194,304	8,388,608	12,582,912	16,777,216	20,971,520	25,165,824	29,369,128	33,554,432
4096	16,777,216	33,554,432	50,331,648	67,108,864	83,886,080	100,663,296	117,440,512	134,217,728
8192	67,108,864	134,217,728	201,326,592	268,435,456	335,544,320	402,653,184	469,762,048	536,870,912

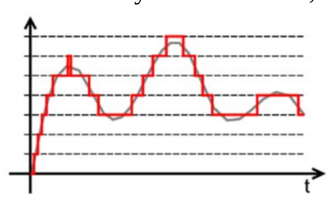
2. Quantization:

In order to process the sampled signal digitally, the sample values have to be quantized to a finite number of levels, and each value can then be represented by a string of bits. For example, if the signal is quantized to N different levels, then $\log_2(N)$ bits per sample are required.

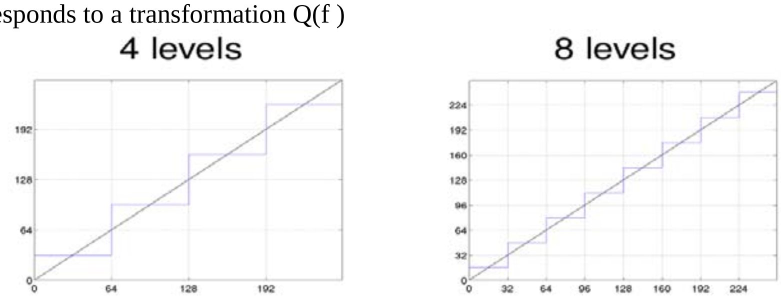
Notice that to quantize a sample value is to round it to the nearest point among a finite set of permissible values. Therefore, a distortion will inevitably occur. This is called quantization noise (or error).

Quantization can be classified as uniform and non-uniform. In the case of uniform quantization, the quantization regions are chosen to have equal length. However, in non-uniform quantization, regions of various lengths are allowed. Non-uniform quantization can be implemented through compression-expansion (or companding) of the signal, and this is commonly used (as in telephony) to maintain a uniform signal-to quantization noise ratio over the full dynamic range of the signal (refer to your textbook for more details).

Quantization corresponds to a discretization of the intensity values. That is, of the co-domain of the function.

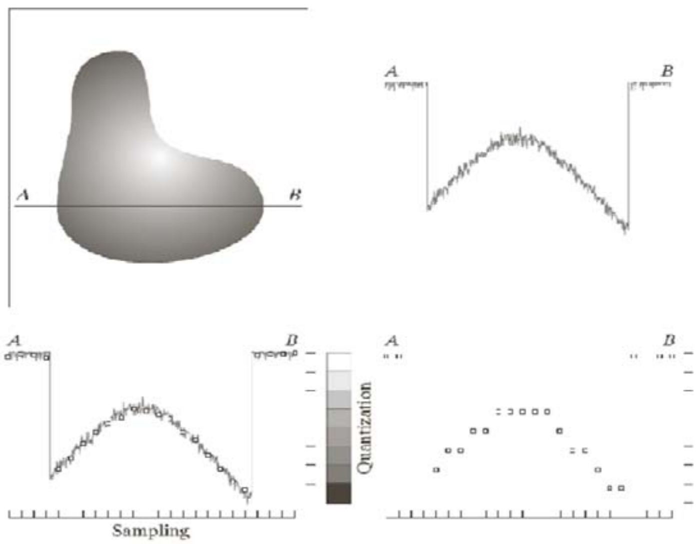


After sampling and quantization, we get
$f : [1, \ldots, N] \times [1, \ldots, M] \longrightarrow [0, \ldots, L]$.
Quantization corresponds to a transformation Q(f)



Typically, 256 levels (8 bits/pixel) suffice to represent the intensity. For color images, 256 levels are usually used for each color intensity.

The number of gray levels is an integer power of 2: $L = 2^k$ where $k = 1,2,\ldots 24$



a b
c d

FIGURE 2.16 Generating a digital image. (a) Continuous image. (b) A scan line from A to B in the continuous image, used to illustrate the concepts of sampling and quantization. (c) Sampling and quantization. (d) Digital scan line.

Procedure:

- Take the photograph of student using desktop camera.
- Convert the photograph into gray scale image using Scilab function “rgb2gray”.
- Vary the sampling and observe the effect.
- Vary the quantization levels then calculate the mean square error (MSE) of the new image compared with original image.

Image Quantization

8 bits per pixel

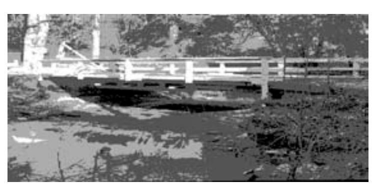
4 bits per pixel

2 bits per pixel

1 bit per pixel

Mean Square Error Calculation

```
93   81  85   85   113 . . .     . .   .
105  97  93   121  105 . . .     . .   .
89   85  113  93   85  . . .     . .   .
105  85  125  170  150 . . .     . .   .
81   93  113  158  146 . . .     . .   .
.                   .                  .
.                         .            .
. . . ..                         .     o
```



```
85  85  85  85   85 . . .        . .  .
85  85  85  85   85 . . .        . .  .
85  85  85  85   85 . . .        . .  .
85  85  85  170  170 . . .       . .  .
85  85  85  170  170 . . .       . .  .
.                    .
.                    .
. . . .     .        .   .            o
```

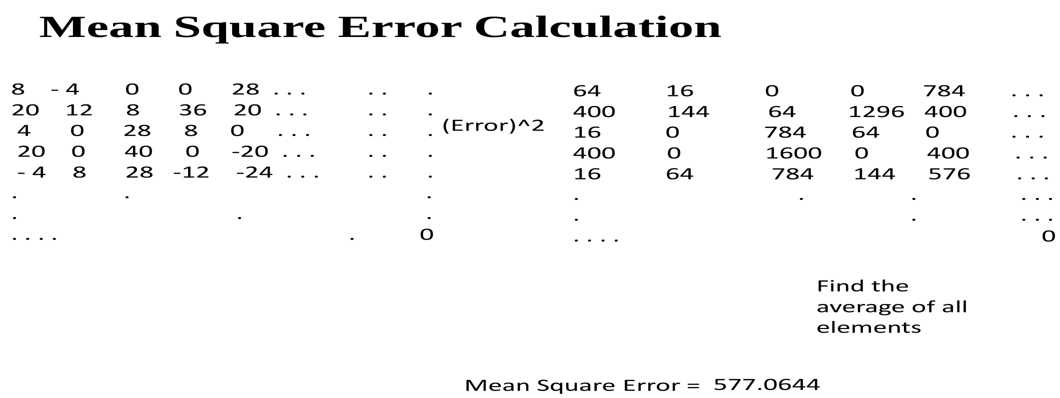


Image File Size Calculation

Example:

1. Image resolution = 512 x 512 pixels
2. Bits per pixel (bpp) = 8
3. Total bits in image = 512x512x8 = 2097152 bits
4. 2097152 bits = 256 KB
5. If bits per pixel = 4, then 512 x 512 x 4 = 128 KB

Bits per pixel (bpp)	Mean Square Error	File size
8	0	256 KB
4	22.6832	128 KB
2	577.0644	64 KB
1	3000	32 KB

Result and Calculation

1. Determine the file size for different quantization level and for different sampling rate.
2. Determine the MSE for different quantization level.

Assignment Questions

1. Capture and save your image in the system provided. In Scilab, *imread()* command reads the image you saved and saves it in an array named ***'image'***. Find the dimension (number of rows and columns) of this array using *size()* command in Scilab. How does this dimension relate to the length and width of the image?
2. Capture and save your image in the system provided. *rgb2gray()* command in Scilab saves the image in gray scale into a array ***image_gray***. Display the first 5 rows and columns of ***image_gray.*** Now, gently cover the camera in the computer provided with your thumb and save the image.

Again display the first 5 rows and columns of the array ***image_gray.*** What difference do you see between the values of former and the later output? Why this difference occurs?

3. Capture and save your image in the system provided. In the Scilab code provided, vary the value of ***nbits*** from 8, 4, 2 and 1. *disp(mtlb_var(image_vec - image_vec_reqd,1,2))* command gives the difference between the original image and the requantized image (quantization error). Tabulate this output for different values of ***nbits.*** What is your observation from the table? What happens to quantization error when ***nbits*** reduces?
4. Capture and save your image in the system provided. In the Scilab code provided, vary the value of ***ratio*** to any of {1/2, 1/4, 1/8, 1/16}. Now open the resized image. Is your image recognizable for ***ratio*** value of 1/16? Why it is not recognizable?

RECORD PATTERN FOR EXPERIMENT 6

AIM:

SOFTWARE USED:

THEORY:

Write the definations of

- Pixel
- Sampling (Specify the nyquist rate and its importance)
- Quantization
- Quantization error (or Mean Square error and specify the importance of mean square error).

PROCEDURE:

Write from the word document

EXPERIMENTAL RESULTS:

1.Tabular Column 1: nbitsVs error

2. Tabular Column 2 :nbitsVs file size

Write step by step calculations for nbits=8.Rest just tabulate the values.

RESULT:

The result should justify your aim.

ASSIGNMENT

- QUESTION
- COMMAND USED(in brackets specify the syntax)
- ANSWER YOU GOT
- JUSTIFICATION (Properly justify the answer with proper statements)