Lab 3 - Image Processing EEE330

Image transform

Report is due in **14 days** from the date of running this lab

Introduction:

This assignment focuses on the image transform, including five tasks. In these tasks, the students will design algorithms to transform images, and these tasks help students understand image transform deeply. The assessment includes the programming code and the report. The programming code should be run successfully, and the results should be correct. Moreover, the code quality will also be considered, such as efficiency, comments, robustness. The report should answer all questions in the tasks, and explain them clearly. It is recommended to add some conclusions about the whole lab.

Objectives:

To test the student learning outcomes A, B, C and D.

Download:

Download the file *Lab3-material.rar* from the ICE, which includes `fence.jpg`, lenna_ds420.bmp, lenna_ds440.bmp, and lenna512.bmp.

Matlab functions:

The following are some *Matlab* functions that might be used in this session:

```
fft2, ifft2, fftshift, imresize

rand, randn, hold on, meshgrid, mesh
```

Hint: read the help about each of the above functions and any other function you might use. Some Matlab functions have a section describing the *Algorithm(s)* they use; it is worth reading this section.

Tasks:

1. Task 1: (**20 masks**)

   (a) Please plot the vector *x = [1:256]* in Matlab. (**2 masks**)

(b) Evaluate the DFT and DCT of the vector x, after that force a percentage of high frequency components of the transformed vector to zero, and see the impact of this on the quality of reconstruct line (the whole processing is equivalent to filtering the data *x* with an ideal LPF). (**8 masks**)

(c) Display the reconstructed line and compute the PSNR for the following percentage of zeroed high frequencies (the percentage: is the amount of zeroed high frequencies with respect to the total number of coefficients). Comment on the subjective quality of the reconstructed line. (**10 masks**)

PSNR	*32/256*	*64/256*	*96/256*	*128/256*	*160/256*	*192/256*	*224/256*
DFT							
DCT							

2. Task 2: (**25 masks**)

(a) Run the following code in Matlab, what do you observe? Does the image have periodicity? (**4 masks**)

```
width = 256; N=width/16; n=[1:width]-1; x = 1+cos(2*pi*n/N);
im = ones(width, 1)*x;
```

(b) Write a program to display the log of *magnitude* image of the 2D-DFT for the above image, what do you observe? Then use the command *fftshift,* and display the result again, what do you observe now? (**8 masks**)

(c) Change *N* in the above task (a) to N=width/**8**, then rerun the above two tasks of (a) and (b), compare the results with the original results; (**6 masks**)

(d) Change *x* in the above task (a) to x = 1+cos(2*pi*n/N)+ **cos(4*pi*n/N)**, then rerun the above two tasks of (a) and (b), compare the results with the original results; (**7 masks**)

3. Task 3: (**15 masks**)

Download the image `fence.jpg` from ICE and copy it on your PC. This image should look like the one in Figure 1:

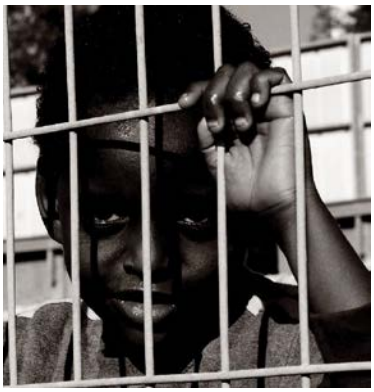


Figure 1: Fence example

Design a filter in the frequency domain which tries to eliminate the “fence” from the image. Describe in your report the designed filter(s) and justify your choices. Please comment the subjective quality.

4. Task 4: (**20 masks**)

Zero-padding technique could be used to resize or scale an image.

To scale-up an image using zero-padding technique you need to add zeroes in the frequency domain at the high frequency band, then you inverse the transform of the zero-padded matrix to generate a scale-up image. A pictorial representation of this process is given in the following:

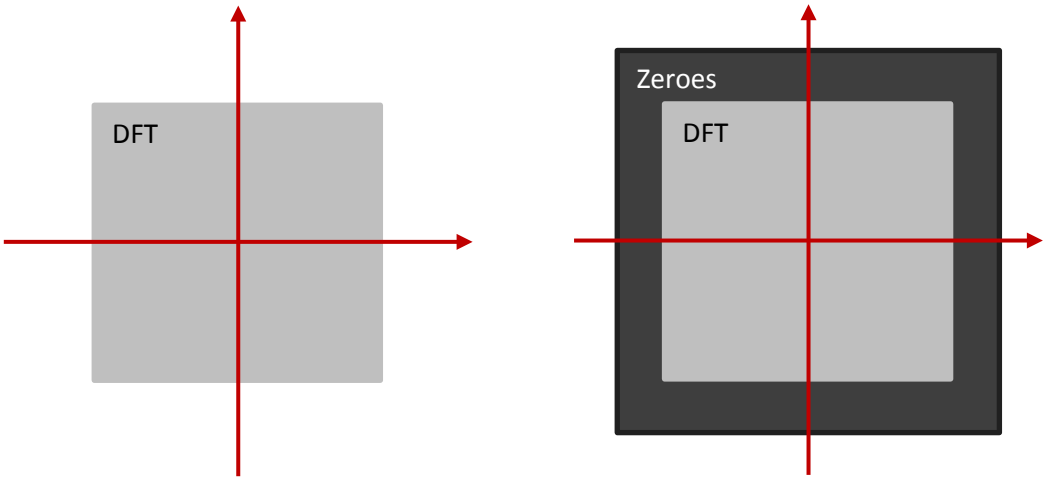


Figure 2: zero-padding example

The amount of zeroes you add determines the size of the output image you obtain after the inverse transform.

(a) Write a Matlab program to scale-up an image using the zero-padding technique. Your program should be flexible to handle different size requirement for the final image. Please note, that you need to have a normalization factor for the scaled image intensity, this normalization factor depends on the original image size and the final image size. (**8 masks**)

(b) Rescale the two images ***im420***`(lenna_ds420.bmp)`, *and* ***im440***`(lenna_ds440.bmp)` to become 512x512 pixels. Display and evaluate the PSNR of the two scaled images with respect to the reference image ***im*** and complete the following table: (**4 masks**)

	im420	im440
Zero-padding		
`Imresize`, *'nearest'*		
`Imresize`, *'lanczos3'*		

(c) Compare the obtained results with the one obtained by using Matlab command `imresize`, using the following kernels `'lanczos3'` and `'nearest'` method. (**8 masks**)

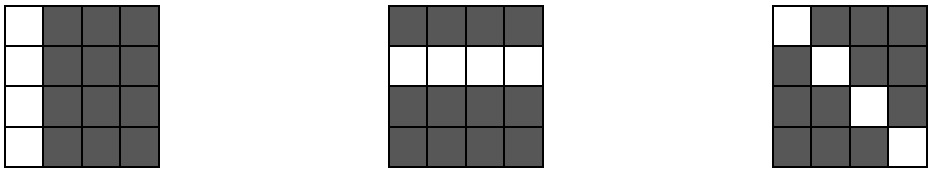
5. Task 5: (**20 masks**)

Functions could be written as linear combination of some bases function as following:

$f(t) = v_1 S_1(t) + v_2 S_2 + v_3 S_3 + ..$

(a) Write a function to generate the bases function for the 4 x 4 images of 2D-DCT? (**5 masks**)

(b) Write the following function to project the images shown in Fig 3 on the 4x4 2D-DCT bases `[im_DCT, DCT_bases] = projection_an_image_on_its_DCT_bases(im)` And what do you observe? (**8 masks**)



(i) (ii) (iii)

Figure 3: three 4x4 images

(c) Write a function which uses `im_DCT` and `DCT_bases` to inverse the DCT transform, did you get the original image ? (**7 masks**)

Lab Report

Write a report which should contain a **concise description** of your results and observations. Include listings of the Matlab scripts that you have written. Describe each of the images that you were asked to display. The format of report should be like this, which is repeated for all questions.

Question
(a) Answers;
(b) Figures if you have. Please add figure title;
(c) Concise code.

Submit the report electronically into ICE, and a hardcopy version into the collecting box beside the office EB310 (Hand written reports are not accepted).

Assignment set date is the 18th April 2018, and the due date is the 2nd May 2018.

Marking scheme

80%-100% Essentially complete and correct work.

60%-79% Shows understanding, but contains a small number of errors or gaps.

40%-59% Clear evidence of a serious attempt at the work, showing some understanding, but with important gaps.

20%-39% Scrappy work, bare evidence of understanding or significant work omitted.

<20% No understanding or little real attempt made.

This page last modified on 2018-04-17 12:55 PM