ShanghaiTech University

CS270 Digital Image Processing

Spring 2022

Assignment 2

Due: 23:59, May. 13, 2022

1. This homework has 15 points in total. Please submit your homework (report in .pdf format and your codes) to blackboard with both subject and file named like **DIP2021-ID-Name-hw1.zip**. The zip file should contain three things: a folder named **'code'**, a folder named **'images'**, and your report named as **Report-ID-Name-hw1.pdf**. Name the filename of your code **'q1a.m'**(for (a) part of question one) so we can easily match your answer to the question. Please write the report carefully since it contributes the main part of your score.
2. This is an individual homework. While you may discuss the ideas and algorithms, at NO time may you read, possess, or submit the solution code of anyone else (including people not taking this course), or allow anyone else to read or possess your source code. We will detect plagiarism using automated tools and any violations will result in a zero score for this assignment.
3. The report should be written in **ENGLISH**.

Problem 1: Frequency Domain Filtering(5 pts)

You are given a matrix **F_data.mat** which contains the Fourier Transform data of an image. Please finish the following tasks using this data.

(a) Implement a 2D inverse DFT function by yourself to transform the data from frequency domain to spatial domain. Compare the results of your function and Matlab function *ifft2* by *norm(X,2)*. Please note that the top left corner denotes the zero frequency.

(b) As you can see, the image transformed by (a) contains many stripes, mainly because the value of **One** frequency has been changed. You need to find a way to remove the stripes and show the image.

(c) Now you finally get the original image, but the contrast is too low, you are asked to enhance the contrast by using homomorphic filter. You may need to try several different sets of parameters to get the best result.

The following items should be included in your **report**.

(a) Inverse DFT algorithm; your output; the L_2 distance to *ifft2* output.(2 pts)

(b) The coordinates of the spatial frequency you chose; the modified image.(1.5 pts)

(c) The parameters you chose; the output images.(1.5 pts)

Problem 2: CT Reconstruction(4 pts)

You are given a sinogram data named **scan.mat** and **sinogram.tif** is the corresponding image after normalization. You need to implement the FBP algorithm by using both band-limited Ramp filter and Hamming windowed Ramp filter. Then perform a level 3 wavelet decomposition of any output image using haar wavelet.

The following items should be included in your **report**.

(a) Explain briefly how you implemented the algorithm.(2 pts)

(b) Two output images; Which one is better? You can answer this question by comparing them with the original image named **img.mat**(1 pts)

(c) The level 3 decomposition like fig1.(1 pts)

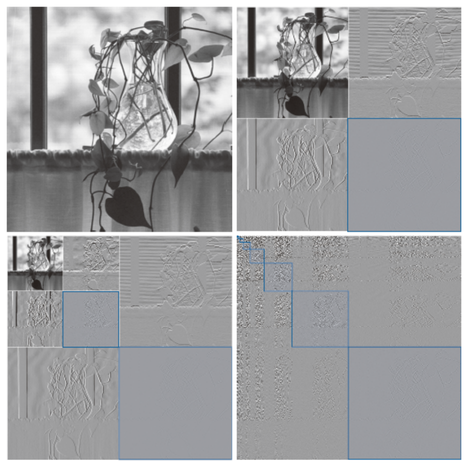

Figure 1: Example

Problem 3: Color Space [1](6 pts)

In this question, you need to follow the steps below to adjust the contrast of the picture. Note that library functions such as color space conversion and convolution operations are not allowed in this question. Some basic input and output library functions such as **imread** and **imwrite** are allowed to use.

(a) **RGB to HSV**

In general, color images are represented in RGB color space. HSV space is closer to human perception in which the (H) refers to the spectral composition of color, saturation (S) defines the purity of colors and (V) refers the brightness of a color or just the luminance value of the color. The RGB values of an image are converted into HSV values using the following equations

$$H = \begin{cases} H_1, & if\ B \le G \\ 360 - H_1, & if\ B > G \end{cases}$$

where

$$
\begin{aligned}
H_1 &= cos^{-1}\left(\frac{0.5[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right) \\
S &= \frac{max(R,G,B)-min(R,G,B)}{max(R,G,B)} \\
V &= \frac{max(R,G,B)}{255}
\end{aligned}
\tag{1}
$$

The range of hue value in the equation above is from 0 to 360° whereas saturation and purity values varies from 0 to 1. Show the result image.

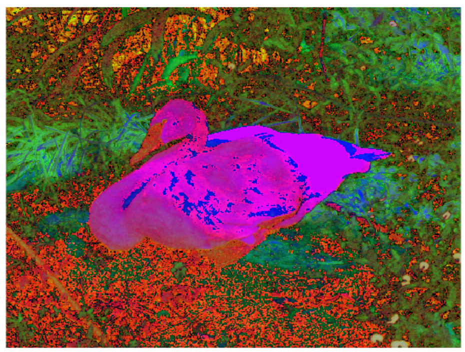

Figure 2: reference image

(b) **Block Overlapped Luminance Enhancement**

The V channel image given by (1) is subjected for luminance enhancement. Suppose V_{LE} the transferred value by applying nonlinear transfer function defined below.

$$
V_{LE} = \frac{V^{(0.75x+0.25)} + 0.4(1-x)(1-V) + V(1-x)}{2} \tag{2}
$$

The shape of the transfer function of (2) depends upon the parameter x, which is an image dependent parameter and is calculated by using the image histogram which is defined as

$$
x = \begin{cases} 0, & for\ L \leq 50 \\ \frac{L-50}{100}, & for\ 50 < L \leq 150 \\ 1, & for\ L > 150 \end{cases} \tag{3}
$$

where L is the intensity level corresponding to a cumulative distribution function (CDF) of 0.1, which means that when more than 90% of all the pixels have the intensity level higher than 150, x is 1. If 10% or more of all pixels have intensities lower than 50, x is 0. For all other cases, x=(L-50)/100.

(c) **Contrast Enhancement**

To get the luminance information of the surrounding pixels, 2-D Gaussian convolution is performed on the original V channel image of the input image in HSV space. Suppose that $G(x, y)$ denotes the 2-D Gaussian function and $V_F(x, y)$ denotes the 8 bit V channel image. The convolution can be expressed as

$$V_C(x, y) = V_F(x, y) \otimes G(x, y) \tag{4}$$

The convolution result V_C contains the luminance information from the surrounding pixels. The amount of contrast enhancement of the center pixel is now calculated by comparing the center pixel value of the original V_F image with the Gaussian convolution result. Following two equations defining the process of center-surround contrast enhancement:

$$V_{CE}(x, y) = V_{LE}(x, y)^{E(x,y)} \tag{5}$$

where $E(x, y)$ is defined as

$$E(x, y) = R(x, y)^p = [V_C(x, y)/V_F(x, y)]^p \tag{6}$$

where $V_{CE}(x, y)$ is the contrast enhanced V channel image, $R(x, y)$ is the ration between the Gaussian filtered and the original value component image. p is an image dependent parameter determined by

$$p = \begin{cases} 3 & for\ \delta \leq 3 \\ \dfrac{27 - 2\delta}{7} & for\ 3 < \delta < 10 \\ 1 & for\ \delta \geq 10 \end{cases} \tag{7}$$

δ denotes the global standard deviation of the input intensity image. Show the result image.

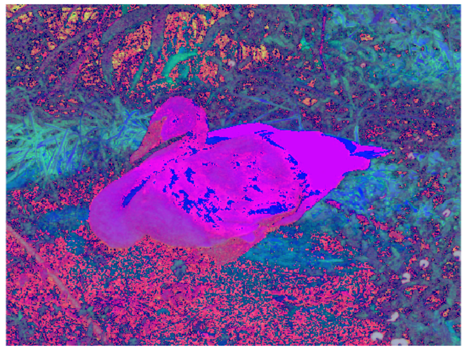

Figure 3: reference image

(d) **HSV to RGB**

When $0 \leq H < 360$, $0 \leq S \leq 1$ and $0 \leq V \leq 1$:

$$\begin{aligned} C &= V \times S \\ X &= C \times (1 - |(H/60^\circ) \ mod \ 2 - 1|) \\ m &= V - C \end{aligned} \tag{8}$$

$$(R^{'}, G^{'}, B^{'}) = \begin{cases} (C, X, 0), & 0^\circ \leq H < 60^\circ \\ (X, C, 0), & 60^\circ \leq H < 120^\circ \\ (0, C, X), & 120^\circ \leq H < 180^\circ \\ (0, X, C), & 180^\circ \leq H < 240^\circ \\ (X, 0, C), & 240^\circ \leq H < 300^\circ \\ (C, 0, X), & 300^\circ \leq H < 360^\circ \end{cases} \tag{9}$$

$$(R, G, B) = ((R^{'} + m)255, (G^{'} + m)255, (B^{'} + m)255) \tag{10}$$

After this, you will get the final result image.

Figure 4: reference image

You need to try it on **swan.jpg**, **Kiener.jpg** and any other picture.

The following items should be included in your **report**.

(a) You need to handle three images(swan, Kiener and your own), 2 pts for each.

(b) For each image, HSV image(0.5 pts), enhanced result in HSV space(1 pts) and RGB space(0.5 pts).

(bonus) Do you have any idea to improve this algorithm?(2 pts at most)

References

[1] D. Ghimire and J. Lee, "Nonlinear transfer function-based local approach for color image enhancement," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 2, p. 8, 2011.