Assignment 3

Brain Teasing in Frequency Domain!

Homeworks Guidelines and Policies

- ***What you must hand in.*** It is expected that the students submit an assignment report (HW3_[student_id].pdf) as well as required source codes (.m or .py) into an archive file (HW3_[student_id].zip).
- ***Pay attention to problem types.*** Some problems are required to be solved *by hand* (shown by the icon), and some need to be implemented (shown by the icon).
  Please don't use implementation tools when it is asked to solve the problem by hand, otherwise you'll be penalized and lose some points.
- ***Don't bother typing!*** You are free to solve by-hand problems on a paper and include picture of them in your report. Here, cleanness and readability are of high importance. Images should also have appropriate quality.
- ***Reports are critical.*** Your work will be evaluated mostly by the quality of your report. Don't forget to explain what you have done, and provide enough discussions when it's needed.
- ***Appearance matters!*** In each homework, 5 points (out of a possible 100) belongs to compactness, expressiveness and neatness of your report and codes.
- ***Python is also allowable.*** By default, we assume you implement your codes in MATLAB. If you're using Python, you have to use equivalent functions when it is asked to use specific MATLAB functions.
- ***Be neat and tidy!*** Your codes must be separated for each question, and for each part. For example, you have to create a separate .m file for part b. of question 3. Please name it like p3b.m.
- ***Use bonus points to improve your score.*** Problems with bonus points are marked by the icon. These problems usually include uncovered related topics or those that are only mentioned briefly in the class.
- ***Moodle access is essential.*** Make sure you have access to Moodle because that's where all assignments as well as course announcements are posted on. Homework submissions are also done through Moodle.

- ***Assignment Deadline.*** Please submit your work **before the end of May 30th**.
- ***Delay policy.*** During the semester, students are given 7 free late days which they can use them in their own ways. Afterwards there will be a 25% penalty for every late day, and no more than three late days will be accepted.
- ***Collaboration policy.*** We encourage students to work together, share their findings and utilize all the resources available. However you are not allowed to share codes/answers or use works from the past semesters. Violators will receive a zero for that particular problem.
- ***Any questions?*** If there is any question, please don't hesitate to contact me through **ali.the.special@gmail.com**. You may also find me in the pattern recognition and image processing lab, 3rd floor, CEIT building.

1. Fundamentals of Fourier Transform (I): DCT, DFT and Sampling Theorem (12 Pts.)

Keywords: *Frequency Domain, Fourier Transform, Inverse Fourier Transform, Discrete Fourier Transform, Discrete Cosine Transform, Sampling Theorem, Aliasing, Nyquist Sampling Rate*

The **Fourier Transform** is probably the most common transformation occurring in the nature. While its applications are mainly prominent in signal processing and differential equations, many other applications also make the Fourier transform and its variants universal elsewhere in almost all branches of science and engineering. No wonder it has such an enormous impact in the area of image processing and computer vision as well.

Trying to mathematically master **Fourier Transform** and related topics in the **Frequency Domain**, here we'll start with three topics: **Discrete Cosine Transform**, **Discrete Fourier Transform** and **Sampling Theorem**.

I. Discrete Cosine Transform

The definition of DCT is

$$F(u,v)=\frac{2C(u)C(v)}{\sqrt{MN}}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}\cos\frac{(2i+1)u\pi}{2M}\cos\frac{(2j+1)\,\mathrm{v}\,\pi}{2N}\,f(i,j)$$

where $i,u=0,1,\ldots,M-1$, $j,v=0,1,\ldots,N-1$, and constants $C(u)$ and $C(v)$ are defined by

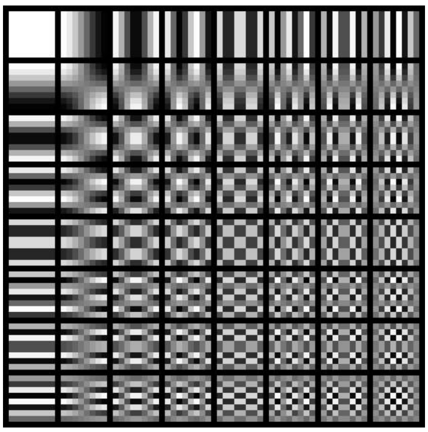
$$C(\xi)=\begin{cases}\dfrac{\sqrt{2}}{2}, & \xi=0\\ 1, & \text{otherwise}\end{cases}$$

Assume the 8×8 image $f(x,y)$ given in Figure 1-a.

a. Determine the value of $F(0,0)$.
b. $F(0,0)$ is known as a DC coefficient. Using the computation experience from the previous part, explain what the meaning of DC coefficient is and why it is called a "DC value".
c. Figure 1-b displays 8×8 2-D DCT basis functions. Using the definition of DCT transform, mathematically prove that $F(0,2)$ is actually related to the pattern in the position $(0,2)$.

10	20	30	40	50	60	70	80
90	100	110	120	130	140	150	160
160	150	140	130	120	110	100	90
80	70	60	50	40	30	20	10
10	20	30	40	50	60	70	80
90	100	110	120	130	140	150	160
160	150	140	130	120	110	100	90
80	70	60	50	40	30	20	10

(a)



(b)

Figure 1 (a) An 8x8 image given for the first part (b) 8x8 2-D DCT basis functions

II. Discrete Fourier Transform

If the Fourier transform of a function $f(x)$ is $F(u)$, determine the Fourier transform of the following functions:

d. $f(x+5)$
e. $f^2(x)$
f. $f(-x)$
g. $f(5x+8)$
h. $F(x)$

Next, consider the following 4×4 grayscale image.

33	**23**	**13**	**03**
32	**22**	**12**	**02**
31	**21**	**11**	**01**
30	**20**	**10**	**00**

i. Calculate the 2D DFT of the image.
j. Multiply the image by $(-1)^{x+y}$ and repeat the previous part.
k. How do the results relate to each other? Explain.

Now, Consider the given matrix below.

l. Compute its 2-D inverse DFT.

$$F(u,v)=\begin{bmatrix} 1 & 0 & 2 \\ 1 & 0 & 2 \\ -1 & 0 & -1 \end{bmatrix}$$

III. Sampling Theorem

Assume $F(u,v)$ (shown in Figure 2) is the 2D Fourier transform of the function $f(x,y)$. $F(u,v)=1$ in the shaded region and zero outside.

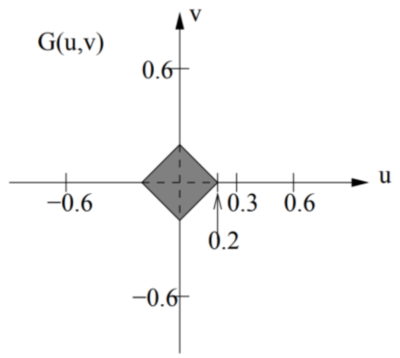


Figure 2 Sketch of the Fourier transform of the given function

m. The function $f(x,y)$ is sampled by a 2D impulse train to

$$g(x,y)=f(x,y)\cdot\sum_n \delta\left(x-n/0.6\right)\cdot\sum_m \delta\left(y-m/0.6\right)$$

In other words, the sampling distance is $\Delta=1/0.6$ in both directions. Sketch $G(u,v)$ in the (u,v)-plane and grade the axes.

n. The same function is sampled by another 2D impulse train to

$$h(x,y)=f(x,y)\cdot\sum_n \delta\left(x-n/0.3\right)\cdot\sum_m \delta\left(y-m/0.3\right)$$

Sketch $H(u,v)$ in the (u,v)-plane and grade the axes.

o. Which one of the functions $g(x,y)$ and $h(x,y)$ show aliasing? Specify which sampling distance Δ will give aliasing for $F(u,v)$ and which not. Mark the aliasing in your sketch.

p. Now if we consider $V(u,v)=\mathcal{R}_{45^\circ}\left[F(u,v)\right]$ (i.e. we get $V(u,v)$ if we rotate $F(u,v)$ 45°). Find the 2D inverse Fourier transform of $V(u,v)$.

q. Specify which sampling distance Δ shows aliasing for $V(u,v)$ and which not.

Note: You may complete the sketch in Figure 1 which is available at the assignment folder.

2. Fundamentals of Fourier Transform (II): 2D Convolution (10 Pts.)

Keywords: *Frequency Domain, 2D Convolution, Image Filtering*

Following the previous problem, here we extend our notion of image processing in the frequency domain to an extremely beneficial process: **2D Convolution**. The whole idea of **Image Filtering** in the frequency domain is theoretically based upon convolution and its properties.

First, consider the following 3×3 filters in spatial domain:

$$h_1(x,y)=\begin{bmatrix}0&1&0\\1&2&1\\0&1&0\end{bmatrix},\quad h_2(x,y)=\begin{bmatrix}-1&-1&-1\\-1&9&-1\\-1&-1&-1\end{bmatrix}$$

a. What's the equivalent filters $H_1(u,v)$ and $H_2(u,v)$ in the frequency domain?
b. Are they low-pass or high-pass filters? Prove mathematically.

Next, assume the following filter transfer functions in the frequency domain:

$$H_1(u,v)=8-2\cos(2\pi u)-2\cos(2\pi v)-2\cos(2\pi(u+v))-2\cos(2\pi(u-v))$$

$$H_2(u,v)=\frac{j}{2}\left[\sin\left(2\pi u/M\right)-\sin\left(2\pi v/N\right)\right],\quad u=0,1,\ldots,M-1,\quad v=0,1,\ldots,N-1$$

c. Find the coefficients of the spatial filters and present them as 3×3 operators.
d. Are these filters low-pass, high-pass, band-pass or band-reject filters? Explain.

Now, assume an input image if filtered in the frequency domain using the following filters. Explain the effects of each filter on the output image.

e. $V_1(u,v)=-4\pi^2(u^2+v^2)$
f. $V_2(u,v)=\left[1+4\pi^2(u^2+v^2)\right]$
g. $V_3(u,v)=\exp\left(-\dfrac{u^2+v^2}{2D_0}\right),\quad D_0=1$
h. $V_4(u,v)=\exp\left(-\dfrac{u^2+v^2}{2D_0}\right),\quad D_0\to 0$
i. $V_5(u,v)=\exp\left(-\dfrac{u^2+v^2}{2D_0}\right),\quad D_0\to\infty$

Finally, the relationship between two images $f_1(x,y)$ and $f_2(x,y)$ in the frequency domain is:

$$F_1(u,v)=F_2(u,v)\left(2-\frac{\sin(3\pi u)}{3\pi u}\frac{\sin(3\pi v)}{3\pi v}\right)$$

j. Find the relationship between the images in spatial domain.
k. Which images will have sharper edges? Why?
l. What's the name of filtering which transforms $f_2(x,y)$ to $f_1(x,y)$?
m. Assuming the following new relationship between the two images in the frequency domain, specify which image will have sharper edges:

$$F_1(u,v)=F_2(u,v)\left(1-\frac{\sin(3\pi u)}{3\pi u}\frac{\sin(3\pi v)}{3\pi v}\right)$$

3. Image Analysis in Frequency Domain is Literally an Optical Illusion! (8 Pts.)

Keywords: *Frequency Domain, Fourier Analysis, Magnitude, Phase*

The focus of this problem is mainly on what it means to represent images in the **Frequency Domain**, what can be inferred from **Fourier Representation** of an image and how image representation in spatial and frequency domain relate to each other.

First, consider the images in Figure 3.

a. Specify which image(s) have Fourier transforms $F(u,v)$ with the following properties.

i. The real part of $F(u,v)$ is zero at all u,v.

ii. The imaginary part of $F(u,v)$ is zero at all u,v.

iii. $F(u,v)$ is purely real and positive for all u,v.

iv. $F(0,0)=0$

v. $F(u,v)$ has circular symmetry

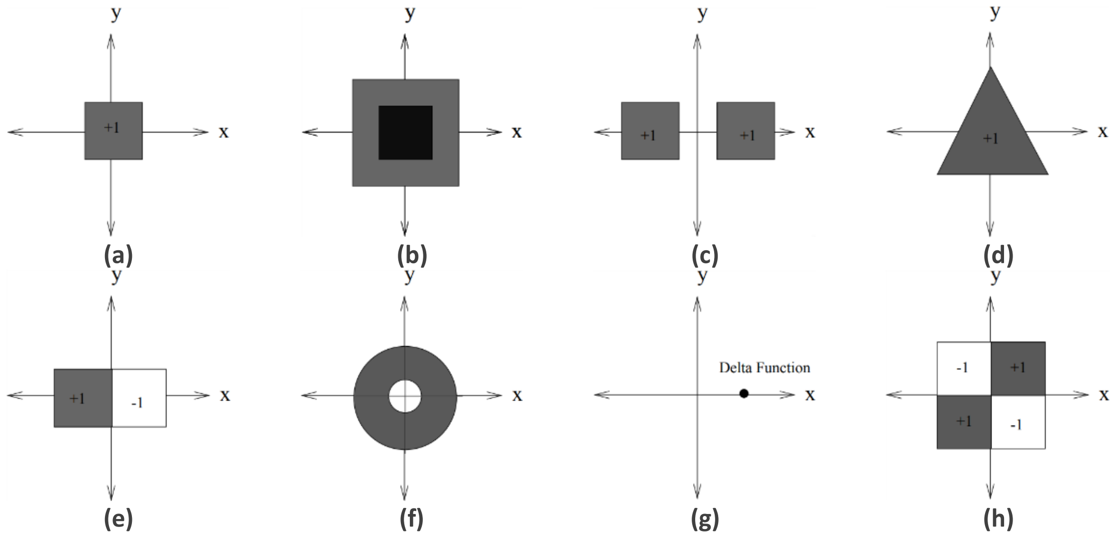


Figure 3 Images with different structures given for the first part

In the next part, we will get a hands-on experience with analysis of image representation in frequency domain. Consider the optical illusions in Figure 4.

b. Compute and display the magnitude and phase of the images. Explain which specific structure in each image yields the results you obtained. Also specify the relations between the images in the spatial domain and the frequency domain.

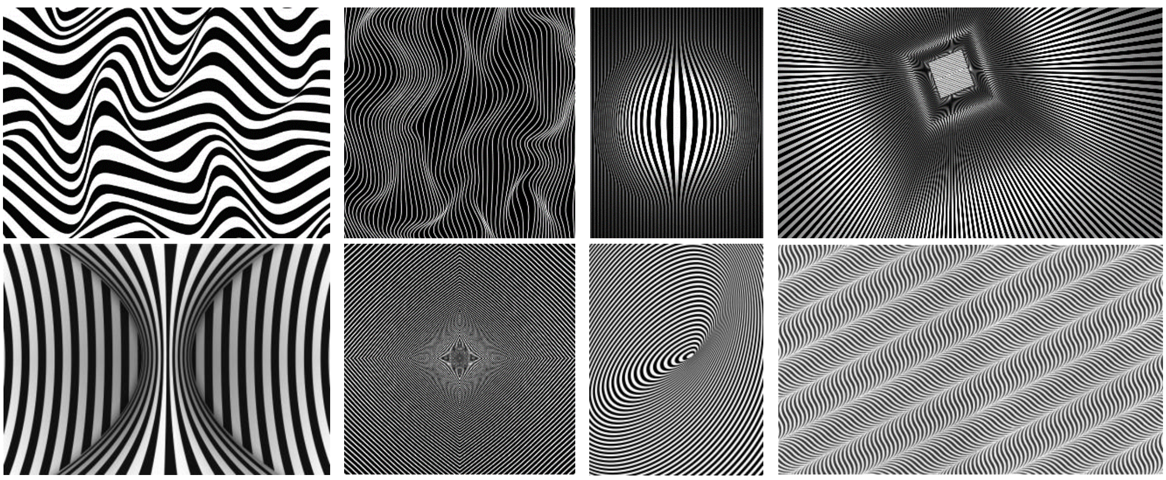


Figure 4 Some optical illusions with different patterns and structures

4. Keep Your Head Still, Fourier Does It For You! (12 Pts.)

Keywords: *Image Filtering, Frequency Domain, Fourier Transform, Inverse Fourier Transform*

One of the latest optical illusions which has gone viral through the social networks is known as *Shake Your Head Illusion*. Started by nanotech engineer Dr. Michelle Dickinson on January 2019 via a Twitter post, this illusion actually requires the viewer to literally shake his head to see the hidden content inside the image, Figure 5.

Dr. Dickinson tweet (here) says: "You can only see this optical illusion if you shake your head (I'm serious)". But we don't agree with that. Here, we are to prove there is no need to shake our heads to discover the content of this image and other illusions of this type.

a. Display and analyse the Fourier transform of the input images in Figure 6.
b. Remove the repetitive patterns by applying appropriate masks to the Fourier transform of the images. Clearly display the hidden content.

Note: You may need further pre-processing and post-processing operations to have the content clearly noticeable. Use the techniques you've learnt so far.

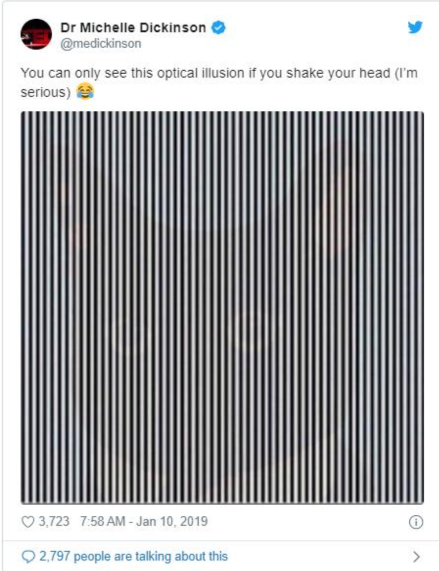


Figure 5 "Shake your Head Illusion" had gone viral after this tweet of Dr. Dickinson. Here, he has claimed that the only way to see this image is to shake your head. But is it really true?

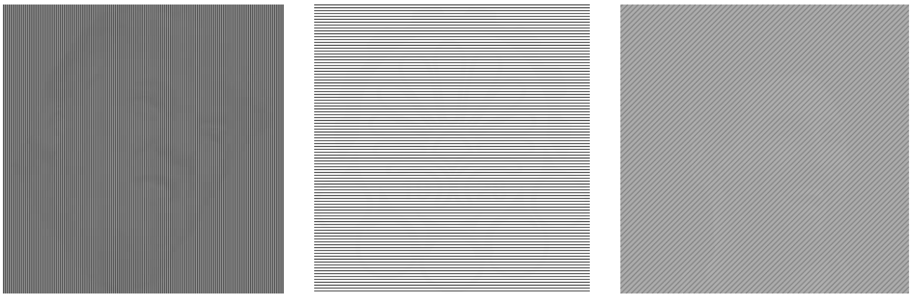


Figure 6 There is a hidden content in these images, covered behind a periodic pattern

In the next part, we'll try to create illusions of this type by ourselves.

c. Use the concept of inverse Fourier Transform to create three types of patterns: vertical lines, horizontal lines and diagonal lines (similar to Figure 6). Display these patterns.
d. Convert the images in Figure 7 to grayscale, and reduce their brightness to an appropriate level. Then add the patterns you created in the previous part to these images. Note that your results must represent the "shake your head" effect.

Figure 7 These images must be hidden in the designed patterns to create "shake your head" effect

5. Don't Always Trust What You See! (15 Pts.)

Keywords: *Image Filtering, Frequency Domain, Fourier Transform, Notch Filter, Color Spaces*

Take a look at the image in Figure 8. This is a famous oil painting by Dutch painter Johannes Vermeer, known as *Girl with a Pearl Earring*. You've probably seen it many times before. But is this image exactly the one painted by Vermeer back in 1665? Well, not quite.

Figure 8 This image is actually a grayscale version of the famous "Girl with a Pearl Earring" painting. Only the grid lines are colored.

The image you see is in fact a grayscale version of that painting. If you look more closely, you will notice tiny color grid lines which trick your brain to perceive and fill in the missing colors itself. This process is known as *Color Assimilation Grid Illusion*.

In order to prove that this effect is actually an illusion, we are going to use **Fourier Transform** to capture and then **Notch Filters** to remove these patterns. Note that, in contrary to the previous problem, you are now dealing with color images. Therefore you need to filter each channel separately, and then combine them again to obtain a filtered color result. On the other hand, based on the pattern characteristics, it might be better to first convert the images into other color spaces in order to better capture those grid lines patterns, and then convert the filtered result back to RGB space again to visualise it. Figure 9 compares amplitude of the Fourier transform in different channels and different color spaces.

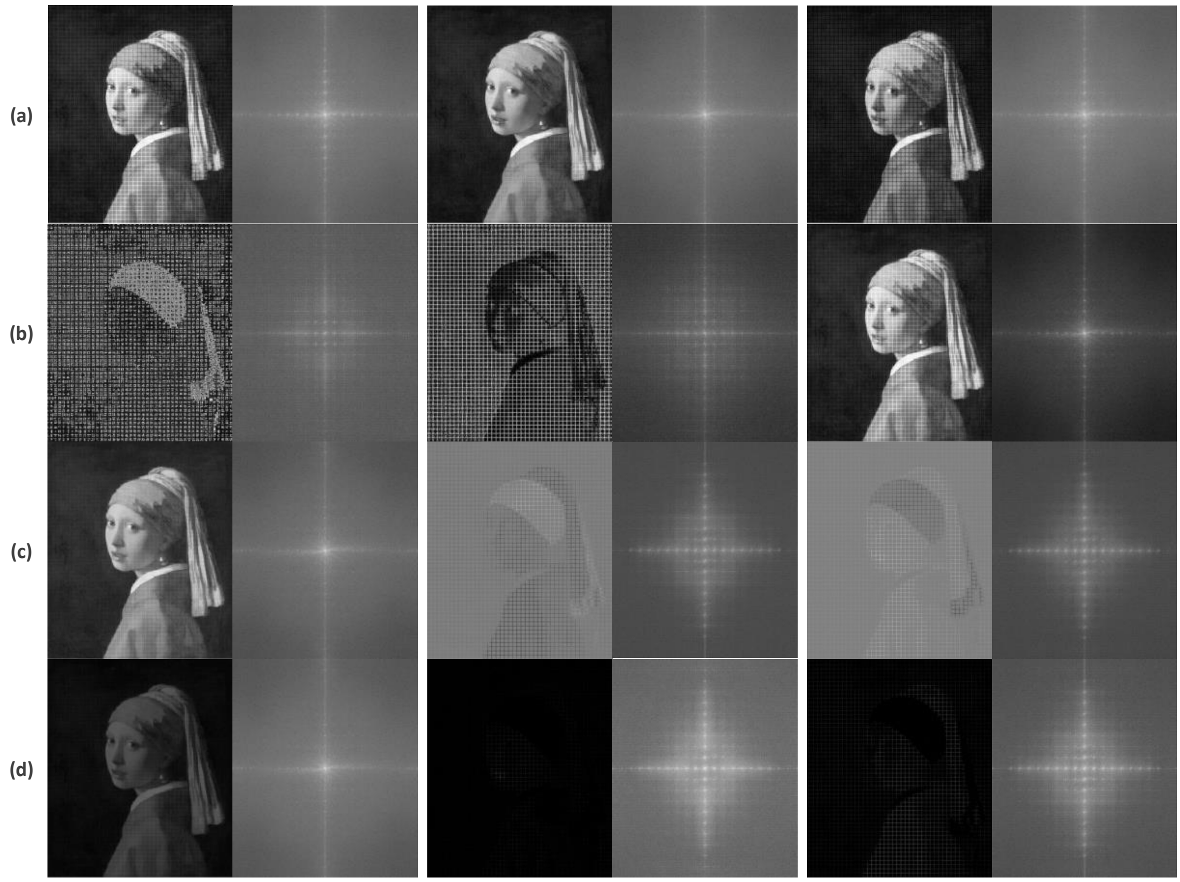
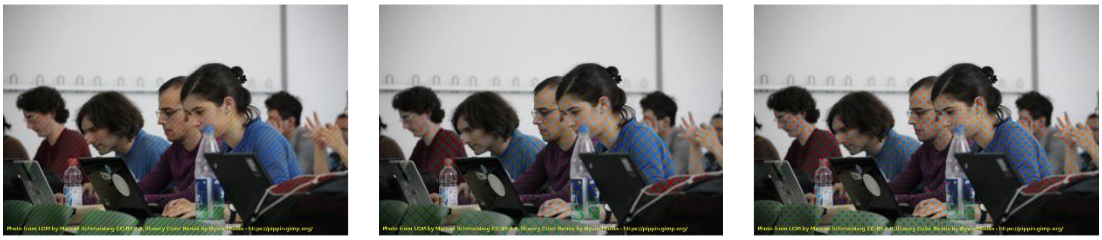


Figure 9 Comparison of Fourier transform of each channel in different color spaces (a) RGB (b) HSV (c)YCbCr (d) Lab

I. When Your Brain Paints!

First we want to prove that the color assimilation grid illusion is really an illusion.

a. For each one of the images in Figure 10, display the amplitude of the Fourier transform of each one of the channels in the color space which is best to highlight its color grid variations.



(a) **(b)** **(c)**

Figure 10 Different grids still put the same effect over the image (a) Diagonal lines (b) Diagonal grid (c) Dot grid

b. Repeat the previous part for the following images. This time, try to remove the patterns and obtain clear grayscale images by applying appropriate bandpass filters.

Figure 11 Removing the color patterns in these images would restore the original grayscale images

II. Impossible Munker's Illusions: Are They Actually The Same?

A similar technique can be applied to create images containing two or more shapes that appear to be in different colors, but are actually of the same color. These illusions is called *Munker's Illusions*.

c. Prove that the shapes inside the images in the following figure are of the same color.

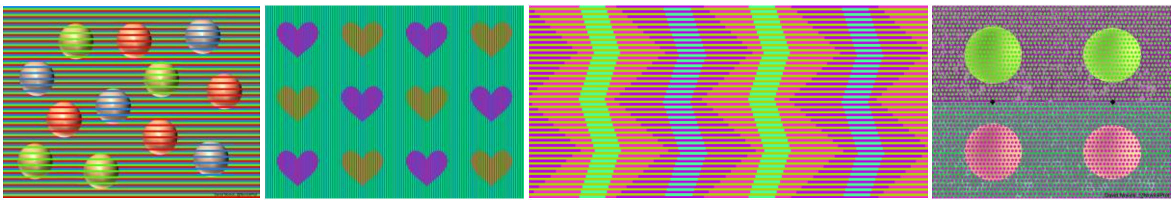


Figure 12 Shapes in these images (spheres, hearts and zigzag lines) are all identical

d. Repeat the previous part for the following images, considering the fact that the patterns are now much more complicated.

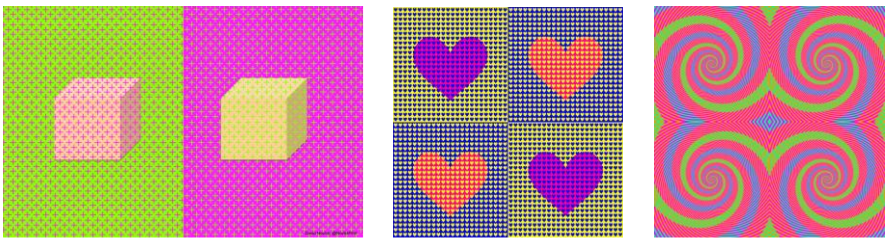


Figure 13 Color patterns in these Munker's Illusions examples are more complicated, but the shapes (cubes, hearts, blue and green spirals) are still identical

6. Albert Einstein or Marilyn Monroe? It Depends on Where You are Standing! (12 Pts.)

Keywords: *Image Filtering, Frequency Domain, Bandpass Filters, Hybrid Images, Image Alignment*

This problem aims to get you familiar with an amazing application of **Image Filtering** in **Frequency Domain**: *hybrid images*. Hybrid images are statistic images whose interpretation depends on the distance between the image and the observer. In an image, **High Frequency** details tend to be more noticeable from close range, while only the **Low Frequency** signal can be perceptible from a distance. The idea is to merge the high frequency content of one image with the low frequency content of the other, and obtain an image whose interpretation varies based on the viewing distance.

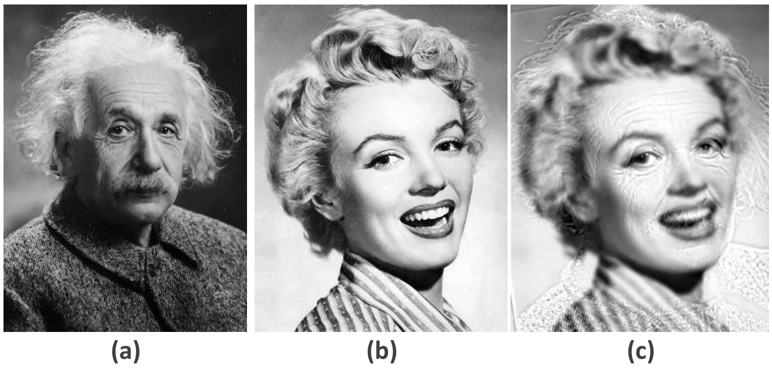


Figure 14 An example of hybrid images (a) Albert Einstein portrait (b) Marilyn Monroe portrait (c) Hybrid images containing both portraits

More precisely, given two images, the procedure starts with aligning their contents (here, faces) so that the final hybrid image makes more sense. Then, it continues with applying a low pass filter on one image and a high pass filter on the other. Finally, all that is left to do is to add the results of the filters, or calculate their averages. The cut-off frequency of each filter should be chosen empirically.

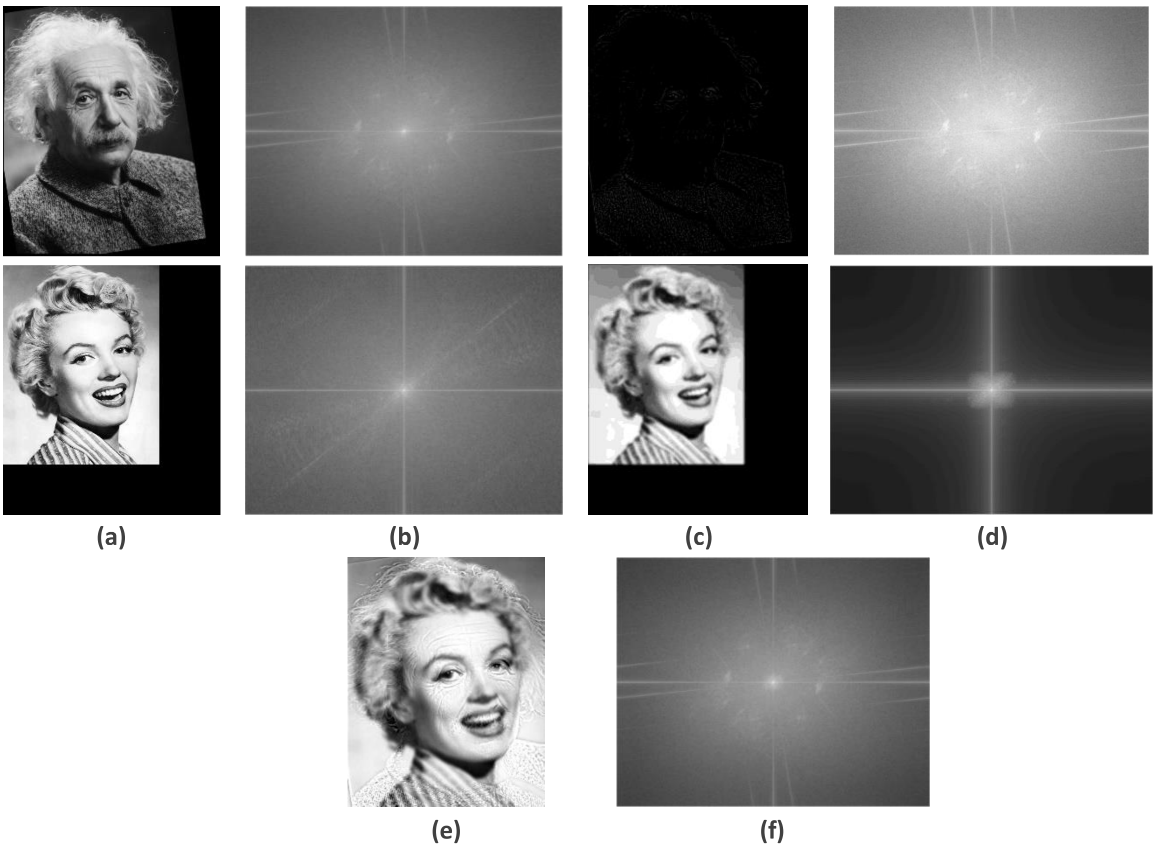


Figure 15 The process of creating a hybrid image (a) Aligned input images (b) Fourier spectrum of the aligned input images (c) The result of applying band-pass filters on the images (d) Fourier spectrum of the filtered images (e) Resultant hybrid image (f) Fourier spectrum of the hybrid image

So, read the provided input image pairs (Figure 16) and perform the following steps for each pair.

a. **Image alignment.** You are given a function (`align_imgs.m` for MATLAB and `align_imgs.py` for Python) which takes two images and two pairs of points, and align them so that the two pairs of points will have approximately equal coordinates. Use this function to align the input images. Display the resultant aligned images and their amplitudes of the Fourier transform.
b. **Image filtering.** Apply low-pass filter on the first image using a standard Gaussian filter, and high-pass filter on the other by subtracting the image filtered with Gaussian filter from the original one. Choose proper values for cut-off frequencies. Display the results as well as their logarithmic amplitude of the Fourier transform.
c. **Merge Images.** Merge the images you obtained in the previous part, and display the final hybrid image and the corresponding amplitude of the Fourier transform.
d. **Visualisation.** Apply a Gaussian filter with five increasing cut-off values on the hybrid images in order to illustrate the process of transformation of one image into another.

Figure 16 Two set of input images. These images must be aligned before adapting the algorithm

7. Photoshop?! Gaussian Filter is There for You! (18 Pts.)

Keywords: *Image Filtering, Frequency Domain, Fourier Transform, Bandpass Filters, Lowpass Filter, Highpass Filter, Gaussian Filter, Laplacian Filter, Image Blending*

Mixing two images and creating a blended image has always been a popular operation in the area of images. This operation – which is usually done using image editing softwares like Photoshop – generally leads to produce hilarious images which amuse everyone. We're going to see how **Image Filtering** in **Frequency Domain** is capable of creating these types of images, even much better than Photoshop!

Here's the idea. Imagine we want to mix specific regions of an image – defined by a mask – with another image, so that the result looks as natural as possible. One way to do so, is to simply crop those regions using the provided mask and place them in the destination image. But as expected, it leads to a stark result (Figure 17). To obtain a seamless and gradual result in the boundaries, one can compute a gentle seam between the two images separately at each band of image frequencies, and combine them eventually to get the final result.

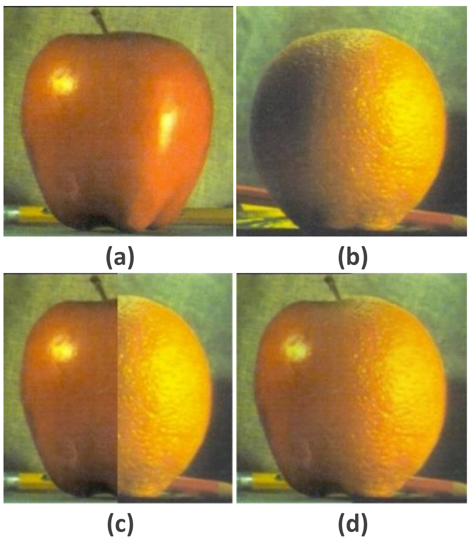


Figure 17 Images blended by the proposed method look very realistic and natural (a) Left pair (b) Right pair (c) Result of the copy-and-paste method (d) Result of the proposed method

More specifically, first both images are convolved with a highpass filter at different levels, each with increasing amount of sharpness. At the same time, mask of desired regions is also convolved with a lowpass filter at different levels, each with increasing amount of smoothness. Finally, at each level, highpass filtered images are combined using lowpass filtered mask as weights. The final result is obtained by adding the blended results at each level, Figure 18.

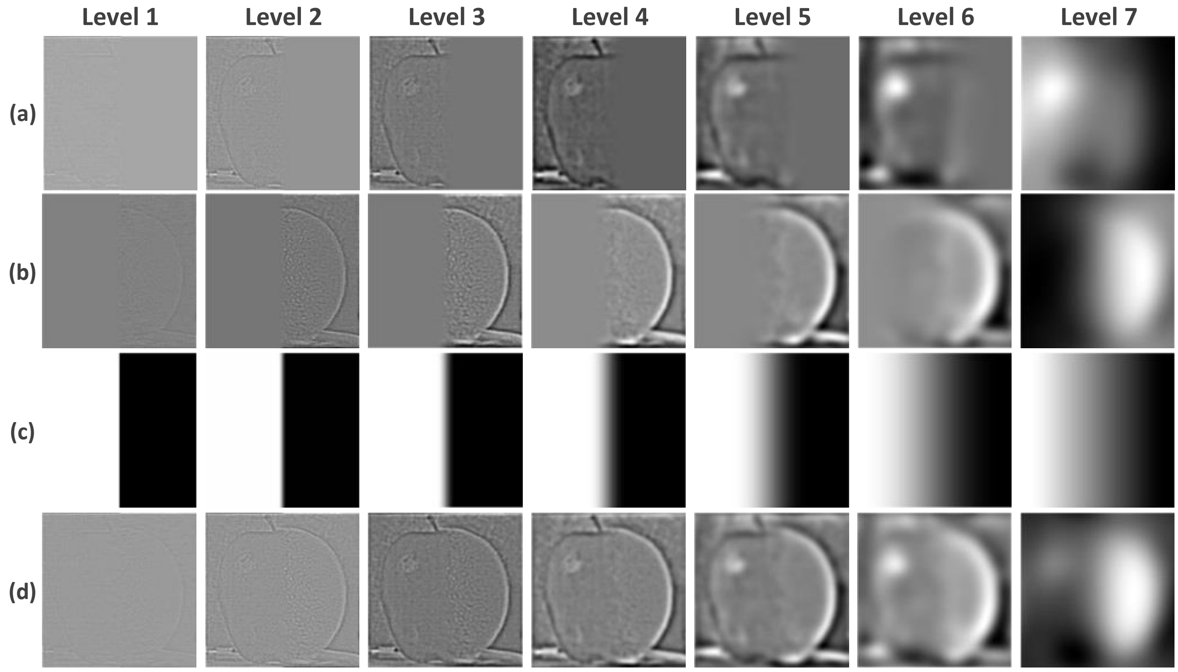


Figure 18 The procedure of the proposed method for mixing two images in different levels (a) The result of applying Laplacian filter on the left part (b) The result of applying Laplacian filter on the right part (c) The result of applying Gaussian filter on the mask (d) The result of combining two filtered images

Now let's see this in action. You are provided with three separate tasks, each with different considerations. The first one is asked to be blended in grayscale, while the other two must be done in color space. Also, the mask of desired regions is given in the first two tasks, while you have to design a mask of your own in the third.

You must apply the following procedure on each one of the given input pairs.

i. Build a set of Laplacian filtered versions of the images A and B using the equation $L_A^l = G_A^l - G_A^{l+1}$ (or $L_B^l = G_B^l - G_B^{l+1}$), where L_A^l (or L_B^l) and G_A^l (or G_B^l) are the Laplacian and Gaussian filtered versions of the image A (or B) at the level l, respectively. Set the number of levels $n = 6$, and the sigma value of the Gaussian filter at each level $\sigma = 2^{l-1}$.

ii. Build a set of Gaussian filtered versions of the mask image M. Set the number of levels $n = 6$, and the sigma value of the Gaussian filter at each level $\sigma = 2^{l-1}$.

iii. Compute the combined images C^l of each level l using

$$C^l = G_M^l L_A^l + (1 - G_M^l) L_B^l$$

iv. Obtain the final result C by summing C^l's through the levels $1, 2, ..., n$.

Note 1: Display all the intermediate results and also include them in your report.

Note 2: Your results must look as natural as possible. Try to find the most convenient parameters (mainly the number of levels, n) to accomplish this task.

(a) (b) (c) (d) (e)

Figure 19 Two examples of mixing images with the proposed method, one without alignment (top) and the other with alignment (bottom) (a) First image (base) (b) Second image (c) Mask of desired region (d) The result of simple copy-and-paste method (e) The result obtained by the proposed method

I. Joint Product of Audi and Saipa!

The first task is a simple one; it only contains mixing images in grayscale.

a. Convert the images to grayscale and mix them using the above mentioned method. Your results must be in grayscale. Set different values for the number of levels n, and report the best result you obtained.

(a) (b) (c)

Figure 20 Creating a new model using a mask and a Gaussian filter! (a) 2020 Audi Q7 (b) SAIPA 151 (c) Mask which keeps half of each cars

II. Messi Who? We Have Donald in Our Team!

Now a more complicated case. The goal here is to merge images in color space. The above-mentioned method must first be applied to each channels separately, and then the blended results in each channel must be combined together to construct a final image.

b. Apply the method to different channels of the first image and obtain a final colored result. Set different values for the number of levels n, and report the best result you obtained.

(a) **(b)** **(c)**

Figure 21 Bringing Donald Trump into the football field! (a) Image of Lionel Messi (b) Aligned and edited image of Donald Trump (c) Mask separating Trump's head

III. Just a Reminder: Wash Your Hands!

In the third and final task, the goal is to mix images in color space, while the mask is not given.

c. Define a zero matrix of the size of the t-shirt image, and then place the note image in the centre of it. Use this matrix to create a mask (set the nonzero values to 1) and a modified image of the note (set the zero values to the background color) to use in the above-mentioned algorithm.
d. Apply the method to different channels of the first image, and then obtain a final colored result. Set different values for the number of levels n, and report the best result you obtained.

(a) **(b)**

Figure 22 Avoid Coronavirus with regularly washing your hands! (a) T-shirt image (b) Note image

8. Further Study: Wavelet Transform and Beyond (20 Pts.)

Keywords: *Frequency Domain, Fourier Transform, Wavelet Transform, Ridgelet Transform, Curvelet Transform, Image Denoising*

Although **Fourier Transform** provides a powerful framework for **Image Enhancement**, it's not the only image processing tool in the **Frequency Domain**. Another type of transforms that comes handy in the area of images is **Wavelet Transform**. Similar to Fourier transform, Wavelet transform is also a representation of a real or complex function using a series generated by a **Wavelet**. A wavelet is a wave-like oscillation with an amplitude which begins at zero, increases, and then decreases back to zero.

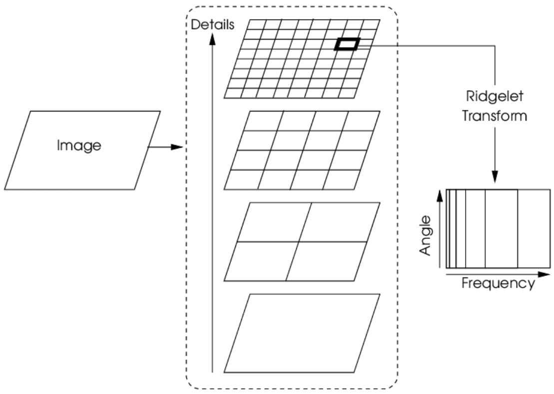


Figure 23 Illustration of the Curvelet transform applied to an image. First, the image is decomposed into subbands followed by a spatial partitioning of each subband. Then the Ridgelet transform is applied to each block. The finest details correspond to the highest frequencies.

Now we want to go one step further and investigate an extension to the wavelet transform, i.e. **Curvelet Transform**. To do so, we focus on 2002 IEEE Transactions on Image Processing paper "*The Curvelet transform for image denoising*". The paper PDF file is attached to this assignment. Please read it carefully and try to make sense of it. You will probably come across some new terms that you may not be familiar with, therefore further reading is inevitable.

Answer the following questions based on your grasp of the paper.

a. Write down a comprehensive summary of your understanding of the proposed algorithm in part IV.
b. Clearly explain the following terms: Wavelet transform, Radon transform, Curvelet transform, Ridgelet transform.
c. Explain all the steps that are required to obtain Curvelet coefficients.
d. How does Curvelet transform overcome Fourier and Wavelet transforms shortcomings? Justify your answer.
e. Write a brief review of the paper based on your understanding of the proposed method.

Note 1: you are free to utilize relations and images from the paper, but a mere translation wouldn't be of much worth.

Note 2: Your answers may not be totally accurate, but your efforts are worthwhile.

9. Some Explanatory Questions (8 Pts.)

Please answer the following questions as clear as possible:

a. Blur your vision and look at the "shake your head" illusions in problem 4. What do you see, and why?
b. Although performing convolution in 3D space, for example in video processing and considering three dimensions of space-space-time, is possible, but it' a poor approach which has almost never been used in video processing. Explain why.
c. How does aliasing appear in an image? Assume an image containing diagonal black and white lines (sinusoidal image). Will the aliasing cause the distance between the lines become longer or shorter? Will it change the direction of the lines?
d. How and under what circumstances the discrete Fourier transform relates to the Fourier transform?
e. In practice, even if Nyquist criterion is satisfied, aliasing cannot be avoided in general. Explain why.
f. The shifted 2D DFT usually forms the shape of plus sign, as it has larger values along the horizontal and vertical axes. Explain why.
g. Suppose for a discrete Fourier transform of an image, we multiply -1 to the phase image and then recombine it with the magnitude and use inverse Fourier transform to obtain a new image. Does the new image bear any resemblance with the previous one? Explain mathematically.

Good Luck!
Ali Abbasi