HOMEWORK 2

Eliott Simon
i6184618
Department of Data Science and Knowledge Engineering
Introduction to Image and Video Processing

May 8, 2020

1 Question 1

The periodic signal that I created is the following:

$$f(x,y) = e^{-j(x+y)} \tag{1}$$

(a) **By Hand:**
The 2D FT is given by the following formula:

$$G(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x,y)e^{-\sqrt{-1}.2\pi ux}e^{-\sqrt{-1}.2\pi vy}dxdy \tag{2}$$

Thus if we apply {1} to {3}, we get:

$$G(u,v) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} e^{-j(x+y)}.e^{-j2\pi ux}e^{-j2\pi vy}dxdy \tag{3}$$

We split the exponents:

$$e^{-j(x+y)}.e^{-j2\pi x}.e^{-j2\pi y} = e^{-jx}.e^{-jy}.e^{-j2\pi ux}e^{-j2\pi vy} \tag{4}$$

Thus, for the first integral,

$$\int_{-\infty}^{+\infty} e^{-jx}.e^{-jy}.e^{-i2\pi ux}.e^{-j2\pi vy}dx = e^{-jy(2\pi v+1)}\int_{-\infty}^{+\infty} e^{-jx-j2\pi ux}dx \tag{5}$$

Since we are integrating a complex exponential we need to use the delta-function.

$$e^{-jvy(2\pi+1)}\int_{-\infty}^{+\infty} e^{-jx(1+2\pi u)}dx = e^{-jvy(2\pi+1)}2\pi[\delta(1+2\pi u)] \tag{6}$$

Now we need to integrate with regards to the y's (second integral)

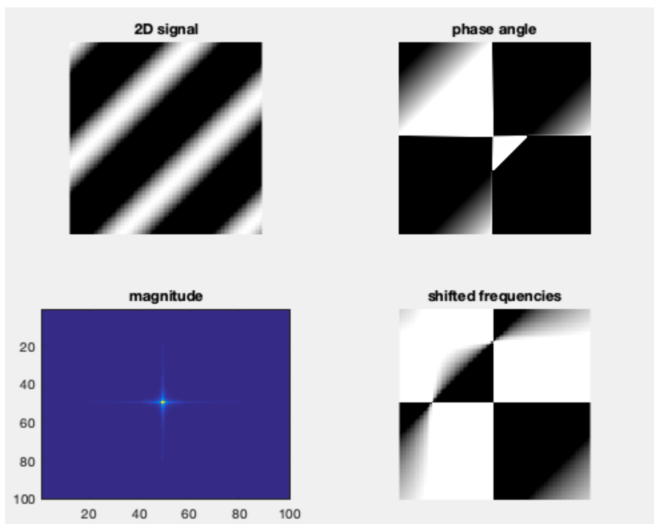
$$\int_{-\infty}^{+\infty} e^{-jvy(\pi+1)}2\pi[\delta(1+2\pi u)] = 4\pi[\delta(1+2\pi u)][\delta(1+2\pi v)], \tag{7}$$

Thus,

$$G(u,v) = 4\pi[\delta(1+2\pi u)][\delta(1+2\pi v)], \tag{8}$$

For continuous time signal, it is possible to compute the 2D FT. It is, though, **not** possible to compute its 2D FFT directly. This stems from the fact that no matter for which dimension, the Fast Fourier Transform (FFT) is a method which computes the *discrete* Fourier Transform of a signal. This

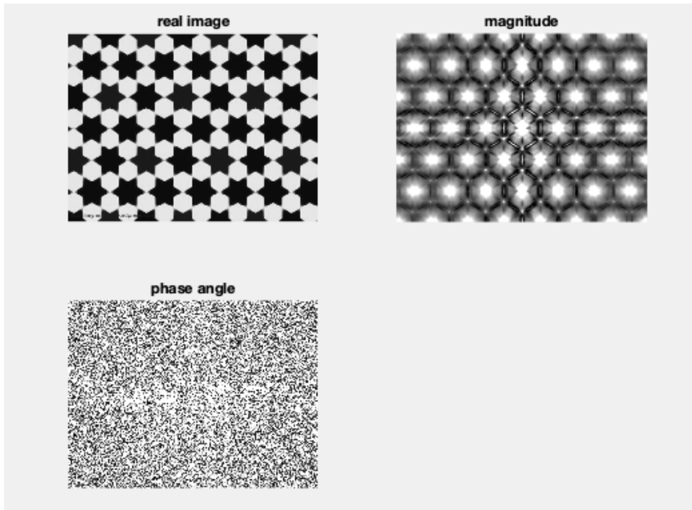
means that we have to use some tricks(e.g the discretization of the signal) in order to find its 2D FFT.
The trick is to partition the 2D continuous signal into intervals with a given frequency f. This way, we can find the formula corresponding $f = \frac{1}{T}$ and thus find the period $T = \frac{1}{f}$. The 2D FFT of the signal is a matrix that is the size of the original signal, filled with complex frequencies. (b) Now that we found our 2D FFT, we can use the shift property of Fourier Transforms in order to get the zero-frequencies in the center of the image.
Now that our frequencies are at the right spot we can find their magnitude by taking the absolute values of all the frequencies. The same way, we can find the phase angle in the interval $[-\pi, \pi]$ of the 2D FFT.



As the signal is super simple, the magnitude values, after shifting the center of frequency coordinates to the center of the image, are centered in the image (see magnitude).

1.1 Question 1.2

The results that I get after creating and shifting the fft2 of the image are the following:

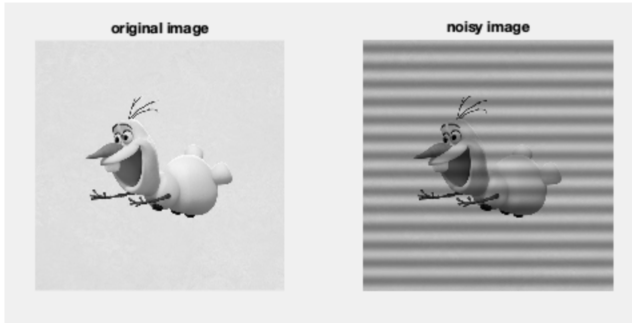


For this image, the magnitude values in the FFT already look bigger than the previous exercise. This would mean that the image has already more information than the signal in question 1a). Another important thing is that there are repeating patterns in the original images as well as in the frequency domain.
Now we are asked to remove the strongest frequency from the FT. Intuitively, the strongest frequency would be the point in the center of the image after shifting the zero frequencies of the FT. We find it and then set it to 0. The result looks like it is the same as the last picture since we only removed 1 single frequency.

2 Question 2

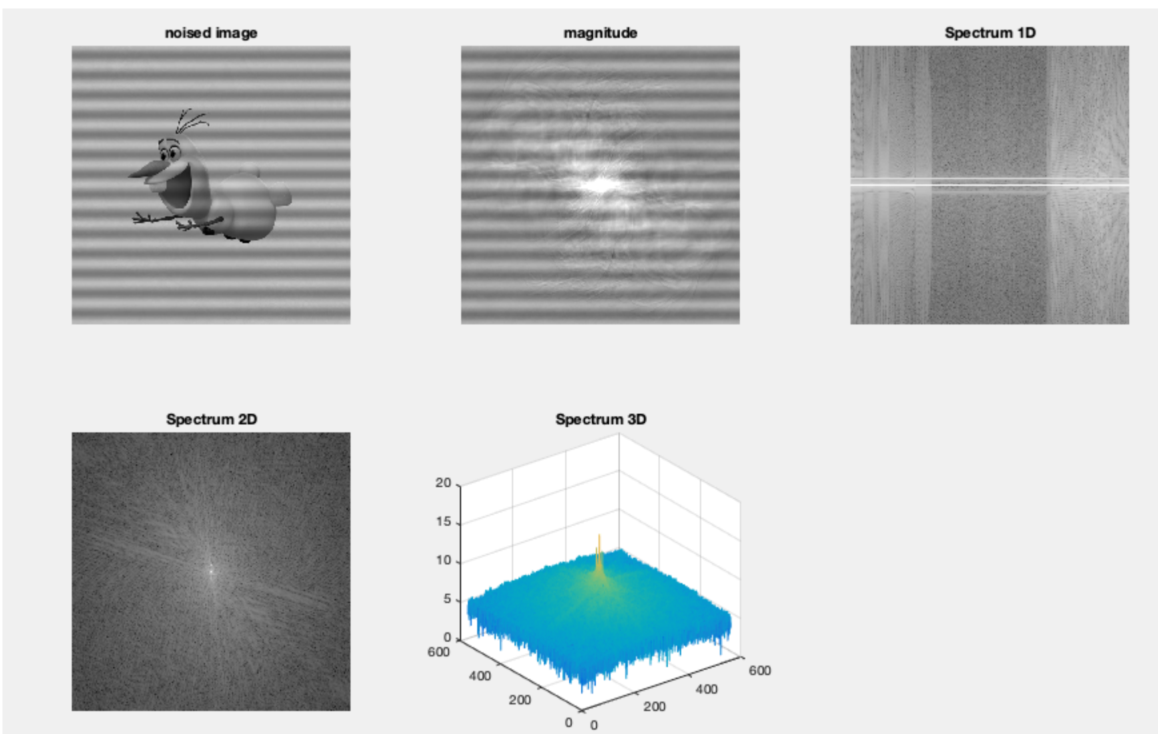
2.1 Question 2.1

Periodic noise can be filtered in the frequency-domain. I made an algorithm to generate some row vector of cosine waves given a period T and amplitude ω. I then map it to the column space in order to get a noise matrix which can be added to our image. I then multiply the original image with the noise matrix to get the noisy image:



2.2 Question 2.2

After computing and shifting the 2D FFT of the noisy image, these are the results:

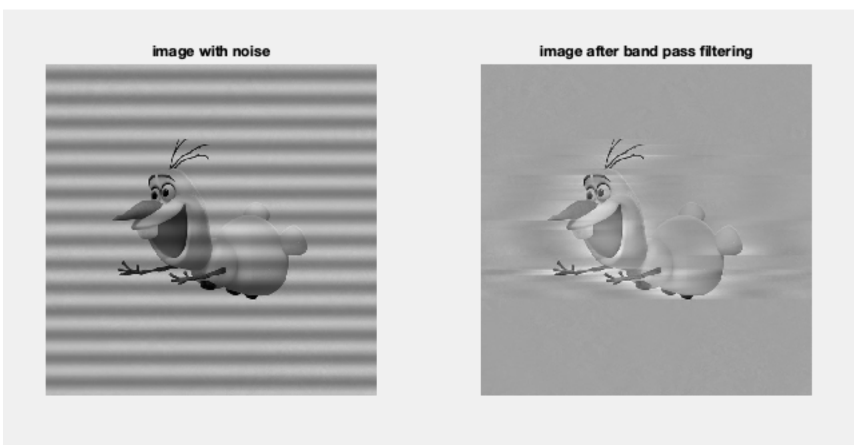


These results are interesting. First of all in the 1D spectrum (which has been computed using fft and not fft2), there are these two straight white lines.

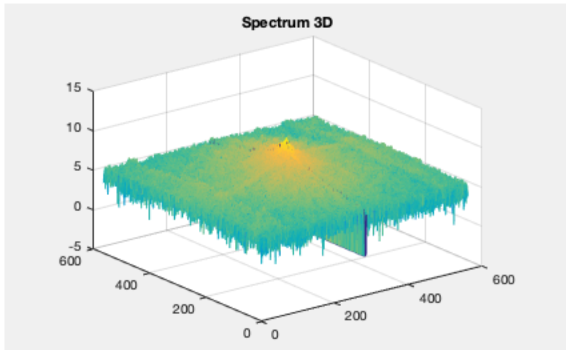
Besides, among all the spectrum, there seem to be a peak of frequency in the middle (which is olaf's shape) but there is also lots of noise which comes from our periodic noise. The noise which looked periodic in the spatial domain looks now like a salt-and-pepper noise distribution in the frequency domain.

2.3 Question 2.3

I tried to make my own filter, kind of following the notch-filter procedure and modify it a little. There exist many (not so many) types of frequency-filtering. The one I decided to use is the Band-Pass method. This method selects a region of interest (ROI) and accepts the signal inside this region. The frequencies which are higher or lower and outside the region are rejected. This algorithm would then allow to reduce considerably the noise of the image. Once we have our center points and the radius of the ROI, we can sample some x and y values at a given $t \in [0\ ,2\pi]$. Once we have generated these points, we can check if we want to filter them or not, regarding their frequencies.



Unfortunately, a lot of details of the original image were filtered because the ROI was probably too big however there is way less noise in the image.

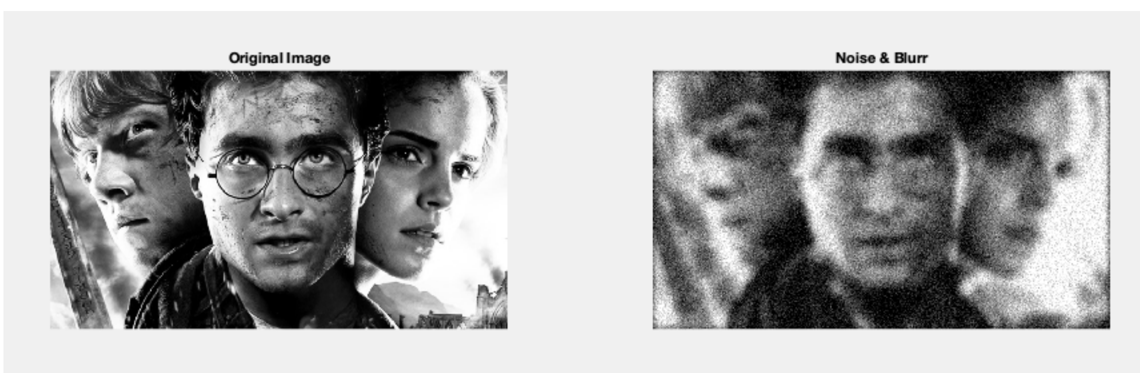


Some frequencies have been lowered after the filtering, the peak is now lower, which is why Olaf has less details now. I believe that the part which goes under the surface of the plot are the noise which have been filtered

3 Question 3

3.1 Question 3.1

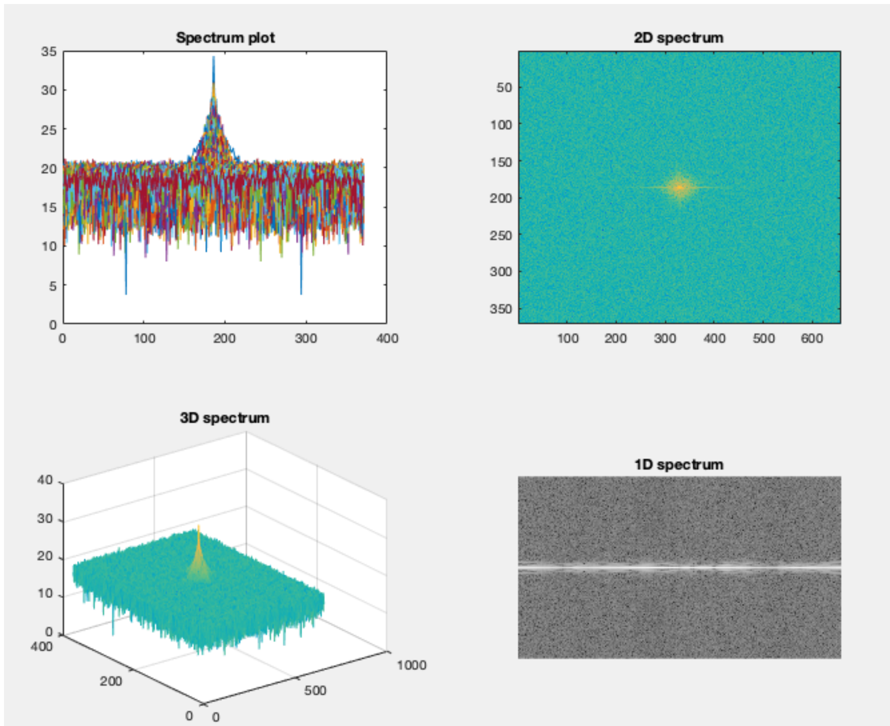
I added gaussian blurring and gaussian noise to the original image and got the following result:



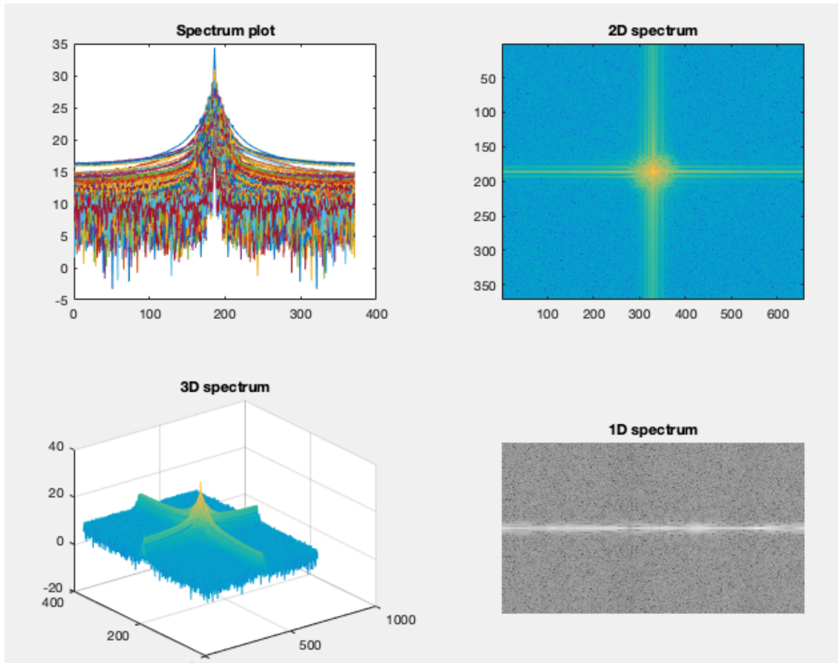
3.2 Question 3.2

We can either find the 2D FFT of the signal using fft2 or the FFT using fft (see code).

3.3 Question 3.3



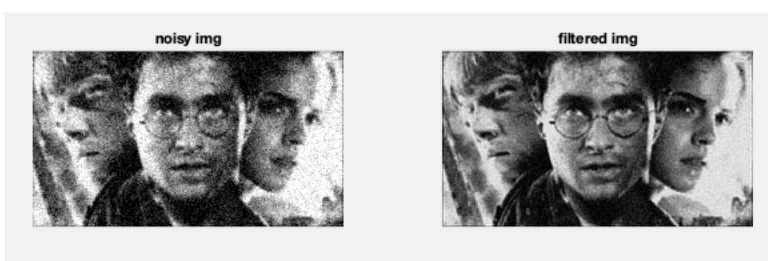
We can see that the frequencies are really clustered in the middle of each spectrum. For the 1D plot, there is a straight line in the middle as well. Apart from this, the rest of the image has pretty low frequencies. If we remove the noise and only keep the blurring, we get such a result:



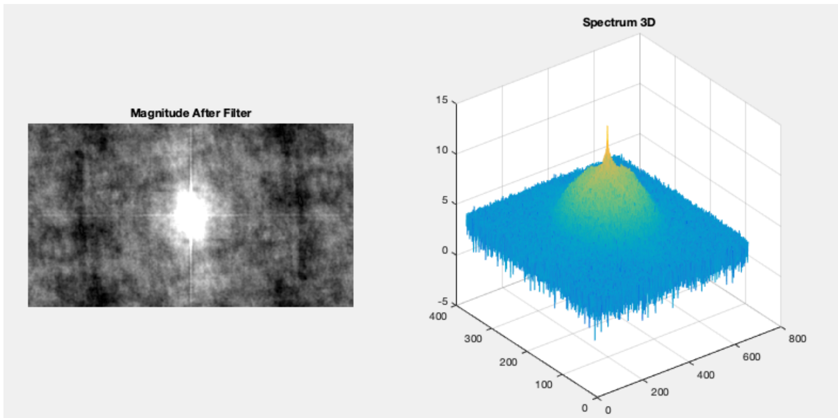
This shows that the noise is responsible for centralizing the frequencies towards the middle of the image.

3.4 Question 3.4

For this exercice I decided to use the wiener filter to try and filter the blurring. I decided not to use the naive inverse filter technique, since because of the double noise (blurring+noise), the filter would have worsen the image (it would work with only blurring). We can find the degradation function H(u,v) and then take its complex conjugated H*(u,v). This technique uses a MSE (mean square error) solver to find its optimum. After selecting the right expected value, this is the result that I got.



The spectrum of the new image is the following:



Now the functions resulting from the filterting looks like a gaussian distribution.

4 Bonus

This blind deconvolution is another technique to deblurr images. However it seems like it needs a well defined PSF, which I do not have, which might explain why I have this result: