# **Homework 1**

AMATH 482

Christian Diangco

## **Abstract**

This paper covers the theory and use of the Fourier transform and Gaussian filtering in order to determine the location of a submarine. The process is discussed in Sec. 3. Algorithm Implementation and Development, after which results and visualizations are displayed in Sec. 4. Computational Results.

## Sec. 1. Introduction and Overview

In this assignment, we determine the path of a submarine in the Puget Sound from noisy acoustic data. In order to do this, we utilize the Fourier transform and Gaussian filtering. By getting the maximum of the FFT average of the data, we obtain the submarine’s central frequency. A Gaussian filter centered on this central frequency is then used to denoise the signal and obtain the trajectory of the submarine.

## Sec. 2. Theoretical Background

In the field of signal processing, one approach for approximating a signal, when given a noisy signal, is to take the sum of simpler components. If we have a signal , then the signal can be approximated as

,

Where are real-value coefficients and are chosen functions. In the case of the submarine signal, is a point in space.

The discrete Fourier transform (DFT) is one approach for this method of signal approximation that uses trigonometric functions for , e.g.

.

We denote the DFT of as .

The fast Fourier transform (FFT) is an algorithm for computing the DFT of a function given a set of equidistant points. We can use its inverse (IFFT) to go back to the spatial domain from the frequency domain and obtain the signal.

Another part of signal processing is filtering. Filtering is an important step in signal processing in order to denoise a signal. When we filter a fourier transform, we are making it so that we only keep the frequency components below a defined threshold. Frequencies above this threshold are set to zero. For this assignment, I applied a 3D Gaussian filter.

## Sec. 3. Algorithm Implementation and Development

For the first task of this assignment, we must find the center frequency of the submarine through averaging of the Fourier transform and visual inspection. In order to do this, I took the FFT of the signal at each timestep, where I reshaped the data at that time step into a 64x64x64 cube, since the measurements come from a uniform grid of that size. After summing the FFTs, I obtained their average like so:

fft\_avg = abs(fft\_sum/49)

The index of the maximum value in this array for the FFT average corresponds to the central frequency of the submarine. I then verified this index by plotting the FFT average at the obtained z-value and others. The amplitude of the frequency was indeed the highest at this z-value.

After obtaining the central frequency, I created a 3D Gaussian filter centered on this central frequency to denoise the data. I defined that filter as a function that takes arguments and , where are grids of points in the frequency domain, and is the value for sigma to be used for the filter. Given these arguments, the function returns

exp(-((x - x\_shift)\*\*2 + (y - y\_shift)\*\*2 + (z - z\_shift)\*\*2)/(2\*s\*\*2))

x\_shift, y\_shift, and z\_shift numeric values used to center the Gaussian filter over the central frequency, defined as

x\_shift = int(center\_x - 32) \* (2\*pi/10)

Where center\_x is the x-value for the spatial location central frequency. y\_shift, and z\_shift are defined similarly. I subtracted 32 and multiplied the difference by in order to convert the value for the central frequency from the spatial domain to the frequency domain.

I then applied this filter to the FFT of the signal at each timestep, and took the inverse of the filtered FFT using NumPy’s ifftn function. I then found the location where the filtered signal was strongest, which corresponds to the submarine’s location at that time step. The process is described in this pseudocode:

Create arrays for storing coordinates of the submarine at each time step

Loop over each timestep:

signal = 64x64x64 cube of data at this time step

fft\_signal\_filtered = Take the FFT of signal and apply Gaussian filter

signal\_filterd = Take inverse of fft\_signal\_filtered

signal\_max\_arg = Find location of maximum of signal\_filtered

Add values of signal\_max\_arg to arrays containing submarine’s coordinates

## Sec. 4. Computational Results

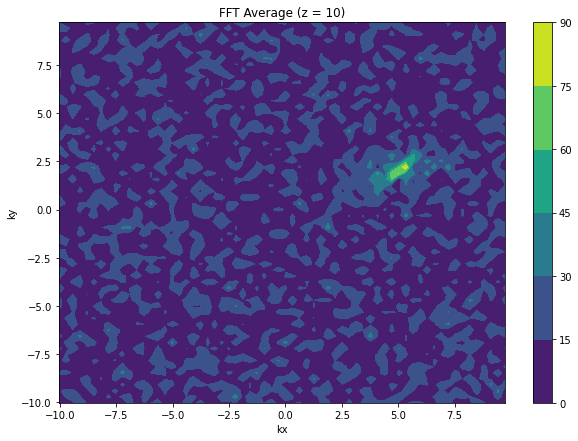
The value I obtained for the location of the central frequency by locating the maximum of the FFT average is

x = 49, y = 39, z = 10,

which corresponds to the following values in the frequency domain:

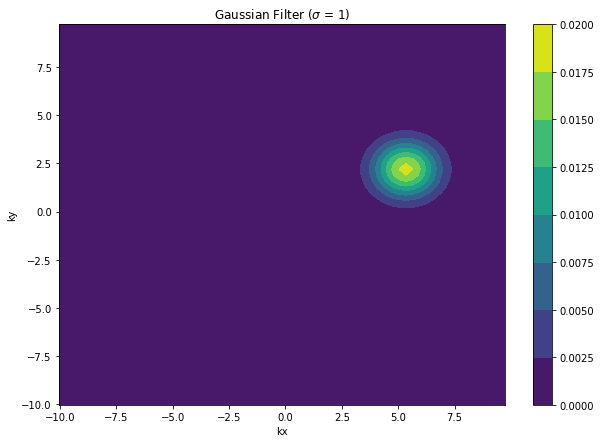
kx = 10.68, ky = 4.40, kz = -13.82

. The distance from this location to the origin in the frequency domain is the central frequency of the submarine. Below is a plot of the FFT average at z = 10.



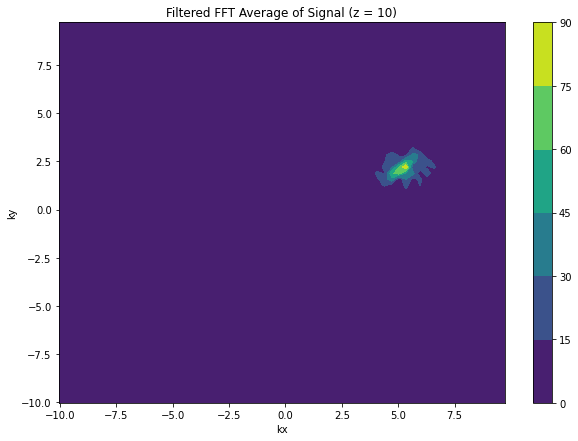
(Figure 1) Plot of FFT average at z = 10. The x-y location of the green and yellow shape corresponds to (49, 39) in the spatial domain.

For my Gaussian filter, I obtained the best results when using . A plot of my Gaussian filter is pictured below.



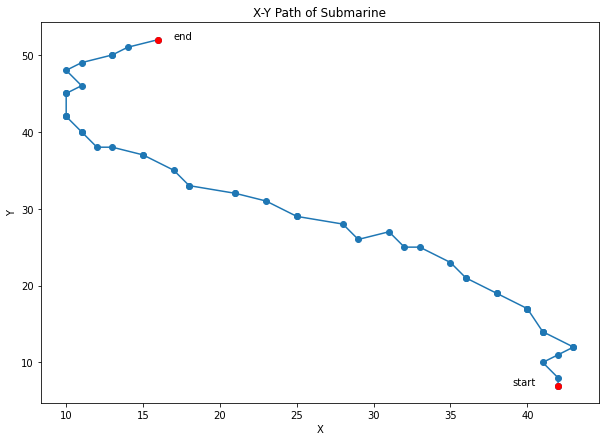
(Figure 2) Plot of Gaussian filter where . The filter is centered over the central frequency seen in Figure 1.

To test my filter, I applied it to the FFT average at z = 10 from Figure 1.

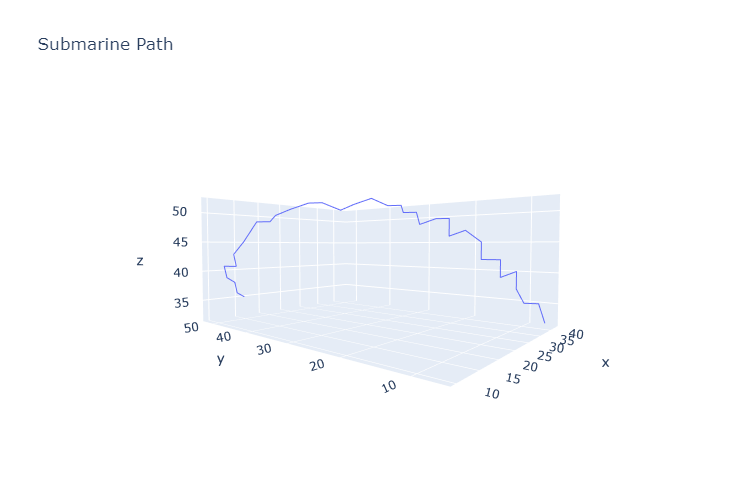


(Figure 3) Plot of filtered FFT average at z = 10. Notice that the FFT has been denoised, isolating the central frequency.

Using my filter to denoise the signal at each time step, I obtained the following trajectory for the submarine:



(Figure 4) 2D plot of x-y trajectory of submarine.



(Figure 5) 3D plot of trajectory of submarine.

## Sec. 5. Summary and Conclusions

Using FFT and Gaussian filtering, I was able to denoise the data and find the submarine’s trajectory. Once the central frequency of the submarine was found, I was able to create a filter centered on that frequency. Using this filter, the signal was denoised and the submarine’s trajectory was plotted.

## Acknowledgments

I would like to acknowledge the students in the AMATH 582/482 Discord who allowed me to have fruitful discussion about this assignment. I would also like to acknowledge Katherine Owens for her guidance on how to approach the assignment in the Canvas discussion board.

## References

Harris, C.R., Millman, K.J., van der Walt, S.J. et al. Array programming with NumPy. Nature

585, 357–362 (2020). <https://doi.org/10.1038/s41586-020-2649-2>

Hosseini, B. (2022, January). Signal Processing with DFT. Seattle; University of Washington.

Hosseini, Bamdad. “High-Dimensional Extensions &amp; Image Processing.” University of

Washington, Seattle, Jan. 2022.

J. D. Hunter, "Matplotlib: A 2D Graphics Environment," in Computing in Science &

Engineering, vol. 9, no. 3, pp. 90-95, May-June 2007, doi: 10.1109/MCSE.2007.55.