Plot Finder Documentation

The basic principle underlying this algorithm is a sliding window Fast Fourier Transform (FFT) approach. Part 1 is a description of the FFT method and how it is used to find rows and ranges in the image. The solution uses this method in two phases, Phase I is used to generate a rough idea of the correct frequency in the image. This information is used to create a filter used in Phase II to generate the Raw wave pad. Part 2 is a description of these two phases. Next, the algorithm uses the raw wave pad to find the center point of rows and ranges in the image. (work in progress to be continued).

Part 1: FFT Background

The FFT is a divide and conquer algorithm that recursively breaks down the Discrete Fourier Transform of any vector. The Discrete Fourier Transform converts a finite sequence of equally spaced samples of a function into a same-length sequence of equally spaced samples of the discrete-time Fourier Transform which is a complex-valued function of frequency. In simple terms this allows one to move from the time/special domain to frequency domain of a vector and the Fast in FFT is simply a more computationally efficient way to compute this frequency domain vector as it required only N * log(N) operations opposed to the N^2 operations of the classical Discrete Fourier Transform. Now we will focus on computation of the classical DFT with the knowledge that in practice we will use the FFT for implementation but the mathematical principles underlying the algorithms are the same.

Part 1: DFT

The DFT transforms a sequence of N complex numbers into another sequence of complex numbers defined by equation 1.

$\{x_n\} := x_0, x_1, \ldots, x_{N-1}$

$\{X_k\} := X_0, X_1, \ldots, X_{N-1}$

$$Eq(1) \qquad X_k = \sum_{n=0}^{N-1} x_n * e^{-\frac{i2\pi}{N}kn}$$

Let’s walk through an example of how this works.