DEEP LEARNING ASSISTED FOURIER TRANSFORM IMAGING SPECTROSCOPY

by

Cory Juntunen

A Thesis Submitted in

Partial Fulfillment of the

Requirements for the Degree of

Master of Science

in Engineering

at

The University of Wisconsin-Milwaukee

December 2020

ABSTRACT

DEEP-LEARNING-ASSISTED IMAGING FOURIER TRANSFORM SECTROSCOPY

by

Cory Juntunen

The University of Wisconsin-Milwaukee, 2020
Under the Supervision of Professor Yongjin Sung

Fourier transform spectroscopy (FTS) has been combined with fluorescence microscopy to allow for high-throughput screening using multi-color probes. In principle, FTS is built on an interferometer, and the intensities of interferograms recorded for varying optical path differences (OPD) provide the emission spectrum of the fluorophores. Here we use deep learning to reduce the sampling number, and thus to increase the data acquisition speed of FTS-based fluorescence microscopy. Even though compressed sensing has been demonstrated to reduce the sampling number, the deep-learning-based approach is able to classify the types of fluorescent dyes without reconstructing the emission spectrum; thereby, it can further reduce the required sampling number. Further, using deep learning, we aim to demonstrate a robust classification without relying on the laser interferometer, which is typically installed in parallel with the main beam path to monitor the actual OPD. We use a 1-D convolutional neural network (1DCNN) together with weight decay regularization and ReLU activation in the hidden layers. For the classification, we use a categorical cross-entropy loss function and an optimization algorithm with adaptive learning rate. For the proof of concept, we simulate 10 fluorescence emission spectra with close emission peaks, then show the NN can distinguish all the different types with 95%

accuracy from about 1/10 of the interferograms typically required in FTS. We also experimentally demonstrate our method using bovine pulmonary artery endothelial (BPAE) cells labeled with three fluorophores. Our approach may lead to compact, fast, robust FTS-based fluorescence microscopy.

I dedicate this thesis work to my wife Aleysha, thank you for all the encouragement and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1DCNN	1-Dimensional Convolutional Neural Net
Adam	Adaptive Moment Estimation algorithm
AI	Artificial Intelligence
BPAE	Bovine Pulmonary Artery Endothelial
CNN	Convolutional Neural Network
DL	Deep Learning
DL-FTIS	Deep Learning Assisted Fourier Transform Imaging Spectroscopy
EXL	Excitation Light Module
FFC	Fluorescence Filter Cube
FFT	Fast Fourier Transform
FOV	Field of View
FTIR	Fourier Transform Infrared
FTIS	Fourier Transform Imaging Spectroscopy
FTS	Fourier Transform Spectroscopy
FWHM	Full Width at Half Maximum
HeNe	Helium Neon
ML	Machine Learning
NUFFT	Non-Uniform Fast Fourier Transform
OPD	Optical Path Difference
OPL	Optical Path Length
PDF	Probability Density Function
ROI	Region of Interest
SGD	Stochastic Gradient Decent
SNR	Signal to Noise Ratio

ACKNOWLEDGEMENTS

I'm very grateful to my advisor, Professor Yongjin Sung, for his invaluable guidance and support over the last couple of years. Thank you for all of the patient and insightful explanation, instruction, and discussion. I greatly look forward to continuing under your mentorship in the PhD program.

Thank you to the thesis committee members: Professor Mahsa Ranji, and Professor Roshan D'Souza for your time and feedback.

Chapter 1 – Introduction

Fluorescence microscopy is commonly used to detect a variety of structural and functional information of biological specimens. A wider range of applications open up as the number of fluorophores that can be simultaneously monitored increases (Yang, Hou, Nelson, & Seibel, 2013) (Orth, et al., 2018). For imaging multiple fluorescent dyes in a sample, optical filters mounted on a rotational wheel is typically used. The simplicity of this technique is the main advantage, but there are several disadvantages. Because switching between different filters is not fast, the imaging throughput is low, and the method is hard to apply to a dynamic specimen. Also, the number of fluorescent probes is limited by the number of optical filters that can physically fit in the system. Further, due to a relatively broad bandwidth of each color filter, fluorescent probes with similar emission spectra cannot be used. Alternatively, a technique called Fourier transform spectroscopy (FTS) can be used to measure the entire spectrum of a sample with nanometer spectral resolution, solving the limitations of traditional multi-spectral imaging. The downsides of FTS are the complexity and cost of the system as well as the requirement of recording many interferograms, typically more than 1000. We aim to address these problems using deep learning.

Traditionally FTS uses a reference laser to correct for experimental noise (Lanoue, Genest, & Gibeault, 2006). The sample beam travels through the Michelson interferometer to the detector, while a reference laser is introduced into the system and travels along the same beam path as the sample beam. As the reference laser generates an almost perfect sinusoidal interferogram, it can be used to correct for the experimental noises (e.g., vibrations from the environment, translation stage error) that affect the optical path difference. The reference laser travels through the interferometer as well and is split off to be measured by another detector. Using deep learning, we aim to eliminate the optical components required for correcting experimental noise, which include the reference laser, two beam splitters or prism mirrors, and a detector (photodiode). Eliminating these components would reduce the cost and

size of the traditional FTS system. Some alternatives to the traditional reference laser system have been proposed, however additional components are added, such as a linear encoder (Haugholt, et al., 2011) (Haugholt, et al., 2013).

Deep learning has been demonstrated with various optical microscopy techniques (Ounkomol, Seshamani, Maleckar, Collman, & Johnson, 2018). In our approach, deep learning is combined with FTS to classify multiple fluorescent dyes in a sample at high imaging throughput. This approach allows us to bypass manual labeling protocols, which is time consuming, requires specialized reagents, and may sacrifice cells (Christiansen, et al., 2018). Deep learning is particularly appealing as many commercially-available fluorescent dyes have overlapping broad emission spectra. Deep learning has demonstrated superior performance over human ability while distinguishing such patterns (Aggarwal, 2018). A compressed-sensing-based method was proposed to reconstruct the sample spectrum from a smaller number (300-500) of interferograms (Wadduwage, et al., 2017). We hypothesize that the interferogram sampling can be reduced further using deep learning if a spectrum doesn't need to be reconstructed while still detecting the correct fluorescent signal.

Chapter 2 – Fourier transform spectroscopy (FTS)

2.1. Introduction

Fourier transform spectroscopy relies on the Fourier transform relationship between the spectrum of light and the interferograms recorded for varying optical path differences (Goodman, 2015). A common interferometer type is the Michelson interferometer. In the Michelson style interferometer, the beam is split as shown on the left in Figure 1. One beam reflects off of a stationary mirror while the other beam reflects off of a moving mirror which is attached to a translation stage. The reflected beams are combined by the same beam splitter, and the intensity values are recorded for varying positions of the moving mirror, i.e., for varying amounts of optical path difference (OPD) between the two beams. As the OPD increases, the light waves constructively and destructively interfere as shown on the right in Figure 1. The interferogram recorded for varying OPD is perfectly sinusoidal, if the source is monochromatic. For a source with finite bandwidth, both the envelope and the frequency of the interferogram are modulated. Computing the Fourier transform of the interferogram results in the spectrum of the original beam, the input to the FTS system.

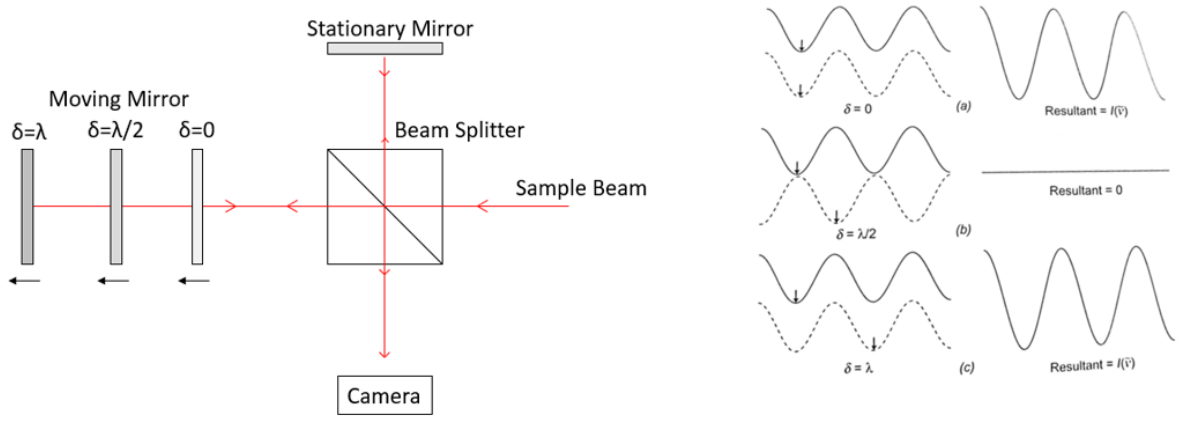

Figure 1: Beam Path of Michelson Style Interferometer (Left), Interference of Light (Griffiths & de Haseth, 2007) (Right)

2.2. Experimental Noise Correction

Since the experimental setup requires collecting the interferograms at nanometer precision, any experimental error that can affect the OPD can significantly degrade the quality of the results. For example, the mirrors mounted on separate posts can vibrate at slightly different times, resulting in the OPD being slightly different than anticipated. The impact that the RMS positional error, Δx, has on the maximum signal-to-noise ratio (SNR), is shown in Equation 1.

$$SNR_{max} = \frac{4}{\Delta x \tilde{v}_{max}} \quad (1)$$

From Equation 1, obtaining a maximum SNR of 10^4 at a maximum wavenumber of 25,000cm^{-1} (for an initial wavelength of 400nm) would require a translation stage with the rms positional error of 0.16nm which is hard to attain in the conventional FTS system. To address this problem, a reference laser is typically used, and the noise is corrected in the data processing. In this research, we have developed our own experimental noise correction (ENC) algorithm, which will be explained in more detail. Figure 2 compares the spectra of the light from a 488nm laser diode, which were reconstructed from the same measured interferograms with and without our ENC algorithm. Computing the spectrum without accounting for the experimental error results in random, noisy profile shown as the dashed line. Applying our ENC algorithm we obtain the sharp peak centered at 488nm, which is shown by the solid line.

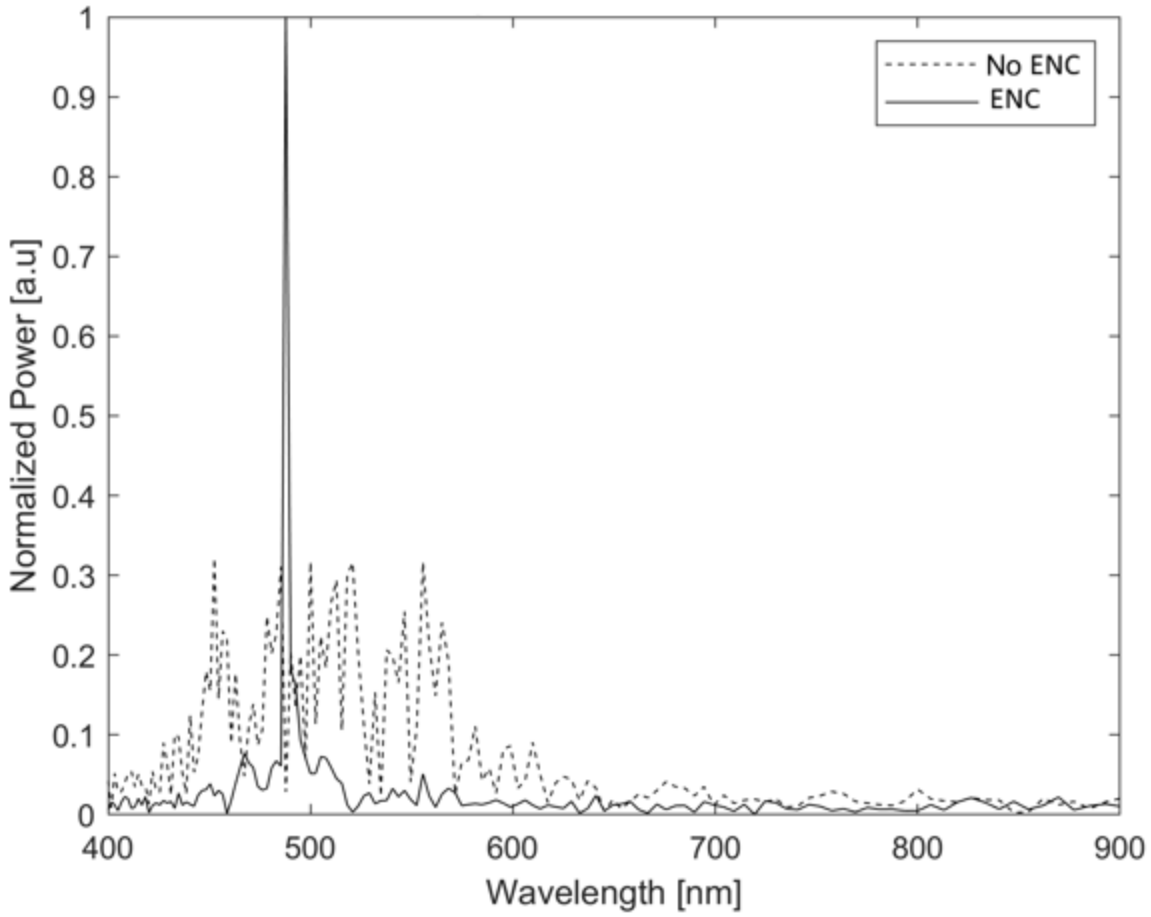

Figure 2: Spectrum of 488nm Laser Diode with and without Experimental Noise Correction (ENC) Algorithm

The noise correction algorithm works by calculating the theoretical interferogram of a narrow band laser and measuring the difference between that and what we experimentally measure. As the reference laser interferogram can also be contaminated with noise, the first step of the noise correction algorithm is to smooth the reference laser interferogram via local curve fitting and by sorting intensity values per incremental step size of the translation stage. The raw interferogram from the reference laser is shown in Figure 3 indicated with the dashed line. The theoretical interferogram of the reference laser at a known narrow wavelength band can be calculated by Equation 2, where A is the amplitude of the wave, λ is the wavelength, x is the OPD, ϕ is the phase, B is the vertical offset, and C is the horizontal offset.

$$I = A\cos\left(\frac{2\pi}{\lambda}(x - C) + \phi\right) + B \tag{2}$$

The amplitude depends on the intensity of the laser sensed by the detector (photodiode). Since the amplitude of the laser and the sensitivity of the photodiode can slightly vary in each experiment, the amplitudes are found by Equation 3, where N is the amount of datapoints and I is the intensity at the detector, and the offset is found from the mean of the collected intensities shown in Equation 4.

$$A = \max(I) - \frac{1}{N}\sum_{i=1}^{N} I_i \tag{3}$$

$$B = \frac{1}{N}\sum_{i=1}^{N} I_i \tag{4}$$

From Equation 2, the captured interferogram should appear to be a smooth sinusoidal wave; however, the raw data shown in Figure 3 is clearly different from it. The major causes include the error in the translation stage step size and asynchronous mirror vibration.

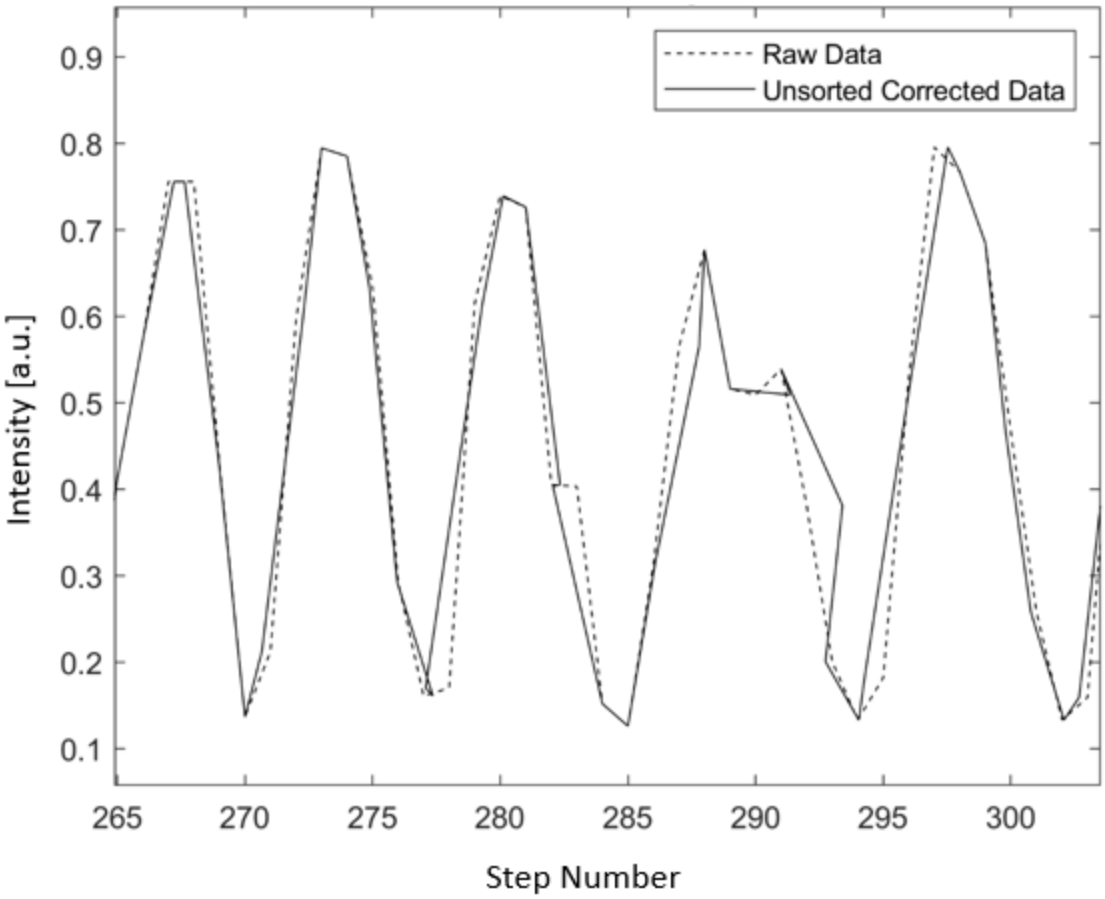

Figure 3: Initial Reference Laser Interferogram Compared to Unsorted Smoothed Interferogram

To find the actual distance traveled by the translation stage, we smooth out the collected interferogram by local curve fitting, shift each central datapoint horizontally to the intersection on the local curve fit, and then sort the intensity values in incremental order per their new shifted stage positions. The solid curve in Figure 3 shows the shifted and unsorted interferogram values. A few observed locations show variance from experimental noise which are smoother after performing the local shift. However, other observed locations show a spike which will need to be corrected using incremental sorting, or it will cause errors with true peak detection. Incremental sorting is needed when the central point of one local curve fit is corrected beyond that of an adjacent corrected point. Figure 4 shows the local data and curve fit which is generated at a selected center datapoint. Specifically, the curve fit is generated using a nonlinear least squares trust-region based optimization algorithm. Equation 2 is simplified by solving

Equations 3 and 4 for amplitude and offset, which is then inputted into the algorithm along with the raw interferogram data, with the variables to optimize being λ and ϕ. Due to the curve fit and the closest intersection shown in Figure 4, the center datapoint which is located at point 6 on the x axis is shifted slightly to the right, indicated by the square located at the intersection between the curve fit line, and the horizontal line.

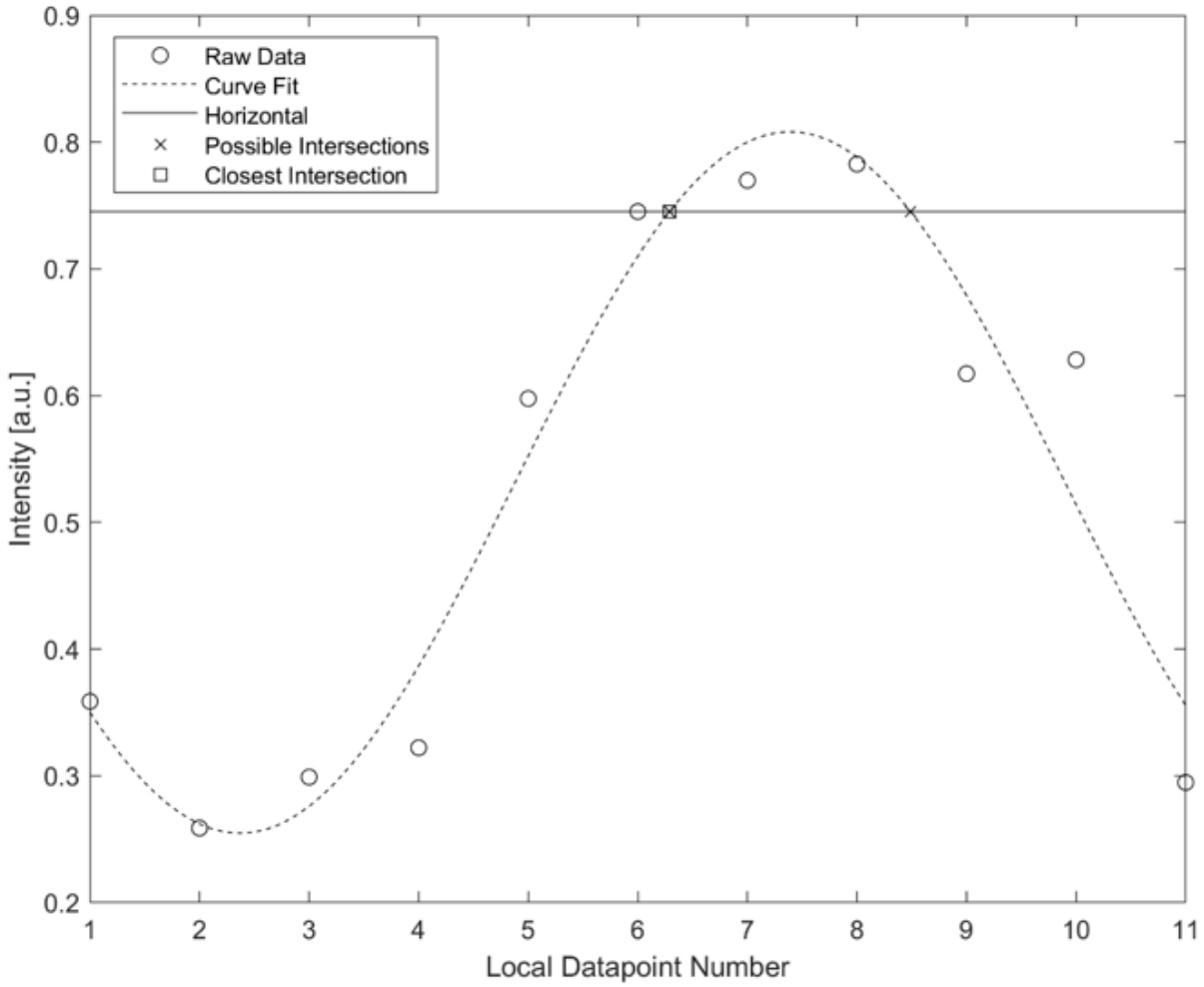

Figure 4: Local Curve Fitting Technique Used to Smooth Interferogram Data

The curve fit wave represents a single wavelength which should closely match the narrow band reference laser interferogram, however we see that the raw datapoints indicated by the circles don't fall directly on the curve fit. This observable error could be caused by the translation stage, asynchronous

mirror vibration, or several other potential experimental noise sources. This is the visual representation of the error mentioned in the discussion of Equation 1 which needs to be corrected in order to maintain an acceptable SNR.

Figure 5 shows the resulting sorted reference interferogram data by the solid line, compared to the raw data shown by the dashed line. In comparison with Figure 3, the resulting interferogram in Figure 5 is smoother. The resulting interferogram doesn't have false peaks and valleys which would cause problems with the next steps.

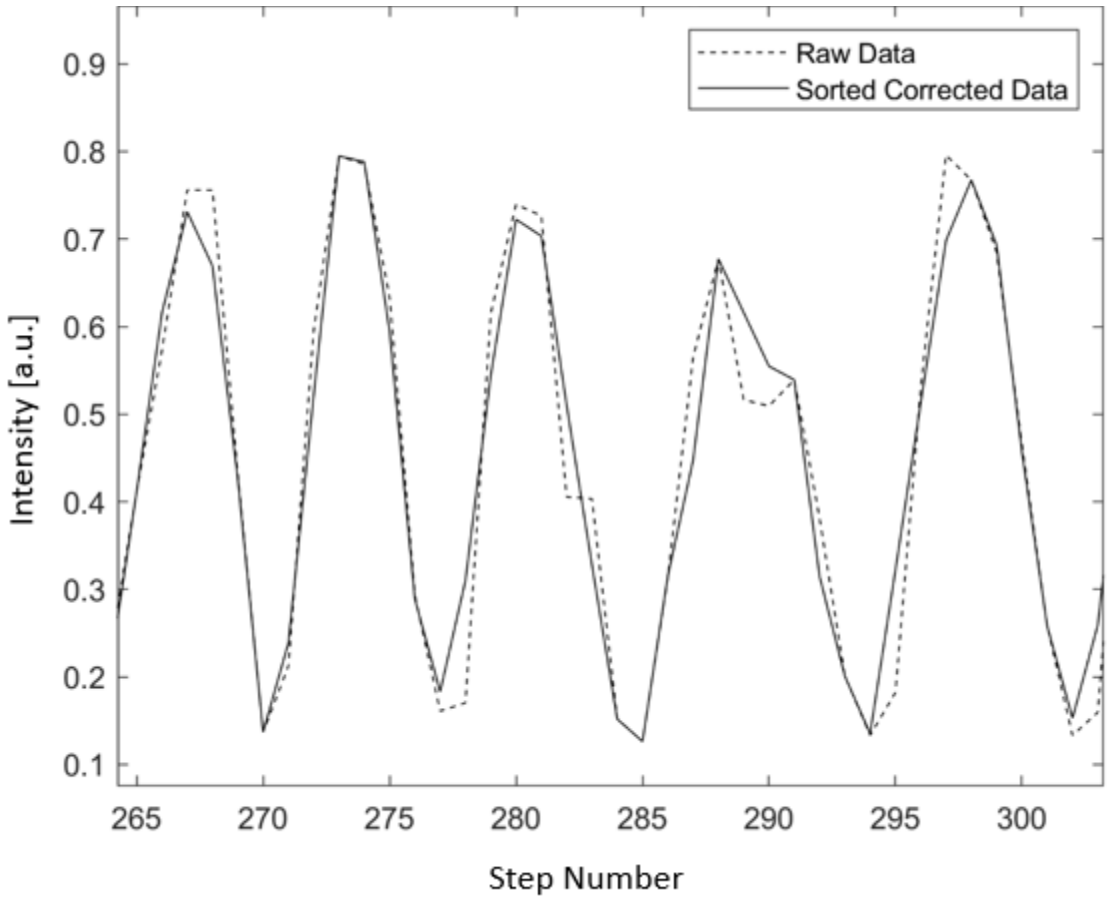

Figure 5: Raw HeNe Interferogram Data Compared to Smoothed and Sorted HeNe Interferogram Data

The next step is to locate the peaks and shift each wave period of the raw data onto the theoretical curve for the reference laser interferogram from Equation 2. The peak datapoint locations were identified and are circled in Figure 6.

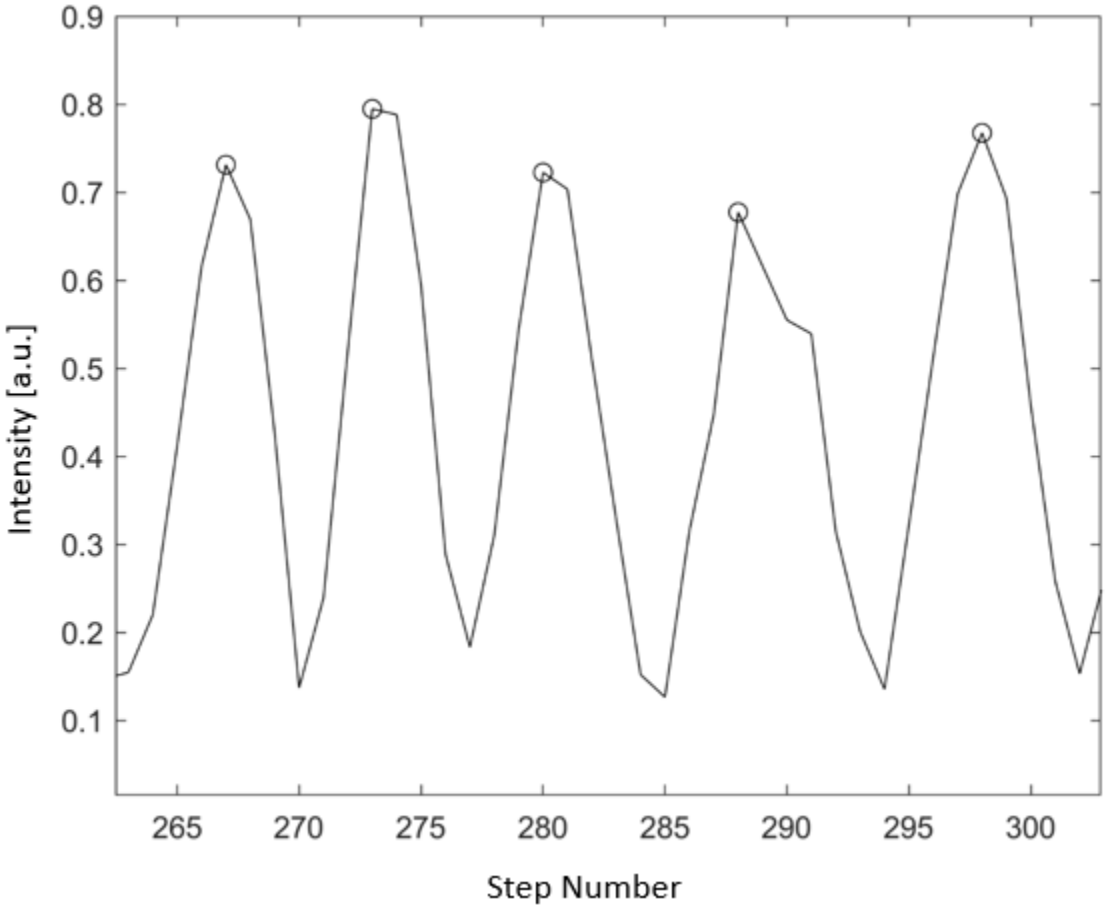

Figure 6: Peak Locations of the Reference Interferogram

The false peaks and valleys which were corrected in the previous steps allow accurate location of the true peaks and valleys. Next, each individual period of the reference interferogram wave is observed. A curve generated by Equation 2 is fit to the data using the same nonlinear least squares trust-region based optimization algorithm with the only variable being the horizontal offset. The raw data is shown by the circles, whereas the theoretically calculated period is shown by the dashed line in Figure 7.

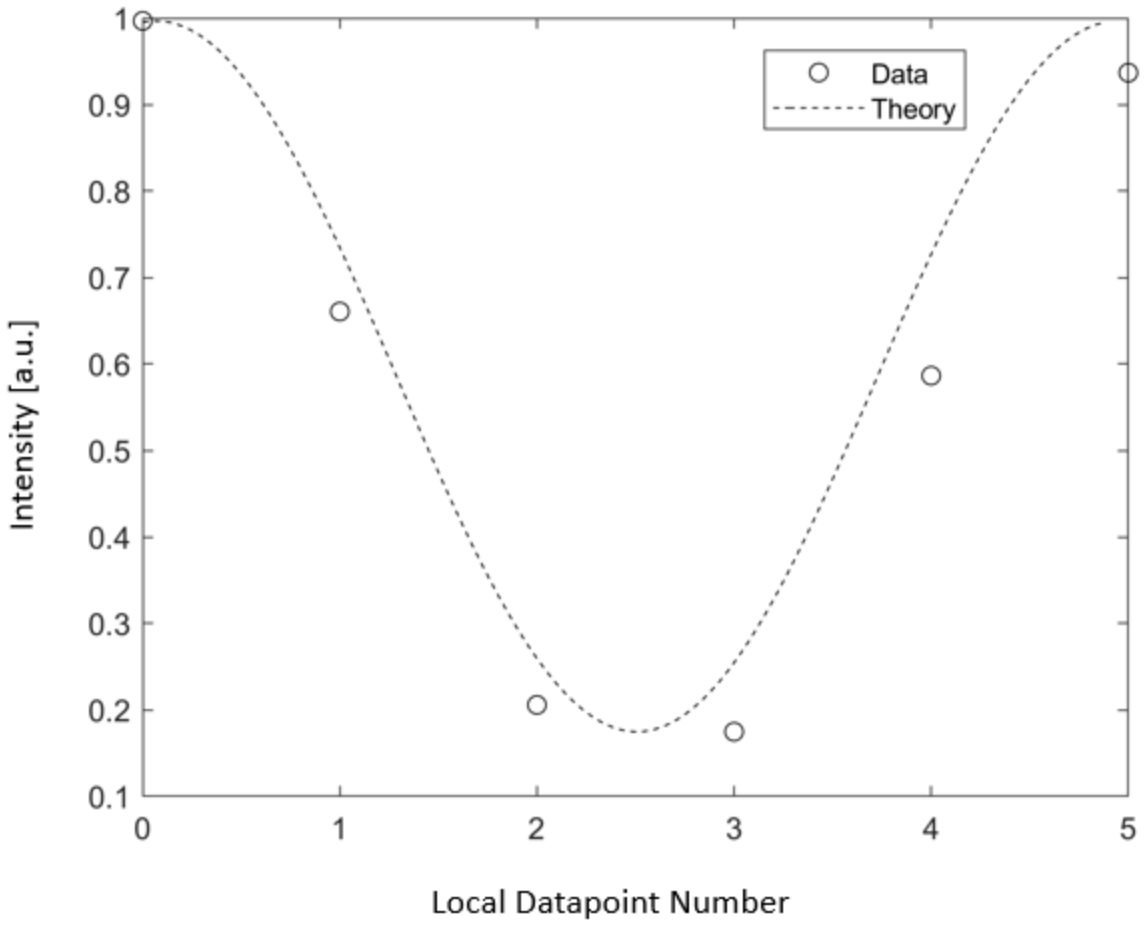

Figure 7: Period of Raw Data Compared to the Theoretical Curve

The horizontal offset which is found by the optimization algorithm is applied to the data to center the period of the raw data onto the theoretical curve shown by the dashed line. This is shown by comparing the dashed line with the circled raw data and the solid theoretical curve in Figure 8.

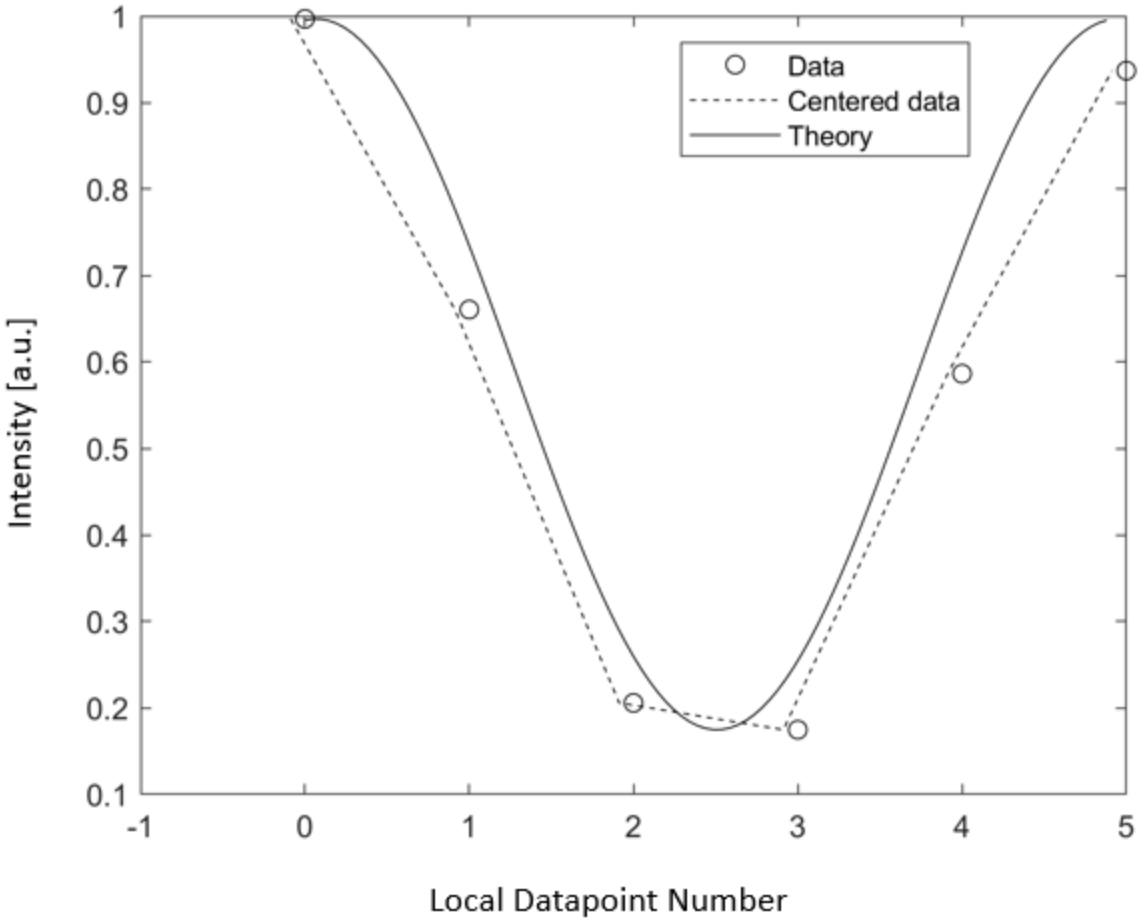

Figure 8: Period of Data Shifted by the Horizontal Offset found by Optimization

Once the data is centered, the elongated wavelength of the reference laser interferogram due to experimental error can be observed. This error is corrected by moving each raw datapoint horizontally to the closest intersection of the theoretical curve shown in more detail in Figure 9.

In Figure 9, the raw reference laser interferogram datapoints are represented by circles. These datapoints each have a horizontal line drawn through them, and intersections with the theoretical reference laser interferogram curve are noted by x's. The closest horizontal intersection for each point is shown with a star.

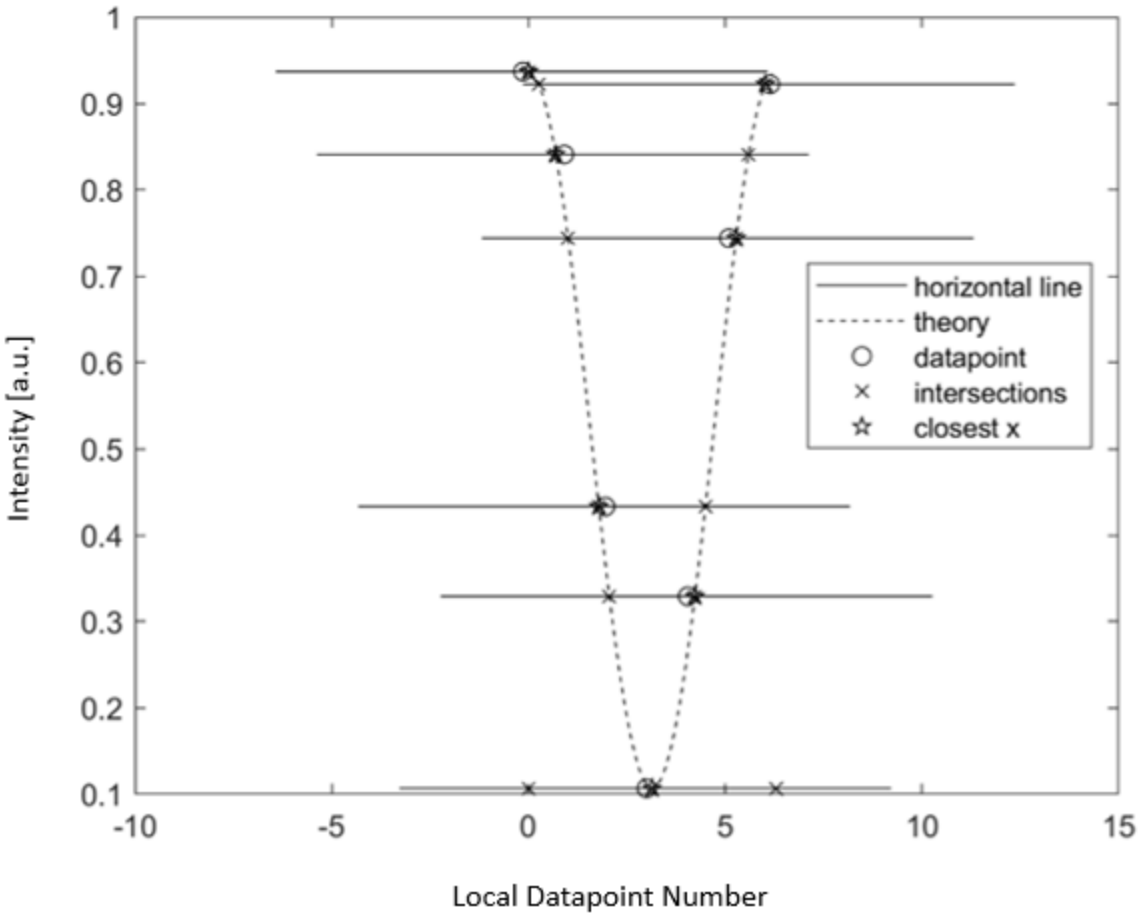

Figure 9: Closest Intersection between the Theoretical Curve and Horizontal to the Centered Data

This entire process of local curve fitting, shifting of a center datapoint and sorting, and finding the distance between raw datapoints and the theoretical curve is repeated for each datapoint, and for each period of the interferogram wave, and the new shifted x values are put into incremental order. Originally, the final step was to linearly interpolate the datapoints at uniform distances so that the FFT algorithm could be applied. The results from this approach were acceptable, however this process induced additional noise due to the interpolated points not being interpolated in a sinusoidal fashion, as well as inducing other frequency errors caused by lowering the peak heights, as interpolated data always results in points less than the maximum, and greater than the minimum peaks and valleys.

To circumvent the induced error, the non-uniform Fast Fourier Transform (NUFFT) algorithm was applied (Bagchi & Mitra, 1999), replacing the linear uniformly interpolated points, and the datapoints sampled at non-uniform intervals due to translation stage error were used directly in the NUFFT algorithm.

Now that the amount of horizontal shifting of each point of the reference laser interferogram has been calculated, these distances are the final distances for the experimental noise correction and will be used in the NUFFT algorithm for each interferogram captured from the sample, which leads to the next step. As the reference laser interferogram is collected at the photodiode, the sample interferograms are collected by the high-resolution camera. These images, which resemble the original sample combined with interference fringes, are processed by collecting the intensity at each pixel with a fluorescent signal. This was done by iterating through the raw images and collecting the intensity values at each pixel in the ROI, for each increment of the entire scan range. These intensity values result in the sample interferograms and are shifted horizontally by the amount found during the resulting reference laser interferogram experimental noise correction (ENC) process.

2.3. Computing the Spectrum using the Fourier Transform

The next step is to apply an apodization function to the interferogram. Apodization is the process of removing the side lobes of an interferogram, to limit the amount of error in the final results, since most of the error is from the noise on the side of the interferogram (Griffiths & de Haseth, 2007). Many apodization functions have been suggested, which perform better in different scenarios. Here we use the Norton-Beer medium apodization function (Equation 5) with the coefficients found in Table 1.

$$A(\delta) = \sum_{i=0}^{n} C_i \left[1 - \left(\frac{\delta}{\Delta}\right)^2\right]^i \tag{5}$$

Table 1: Coefficients C_i for the Norton-Beer Apodization Functions

	C_0	C_1	C_2	C_3
Boxcar	1	0	0	0
Weak	0.384093	-0.087577	0.703484	0
Medium	0.152442	-0.136176	0.983734	0
Strong	0.045335	0	0.554883	0.399782

The Norton-Beer medium apodization function is shown in Figure 10.

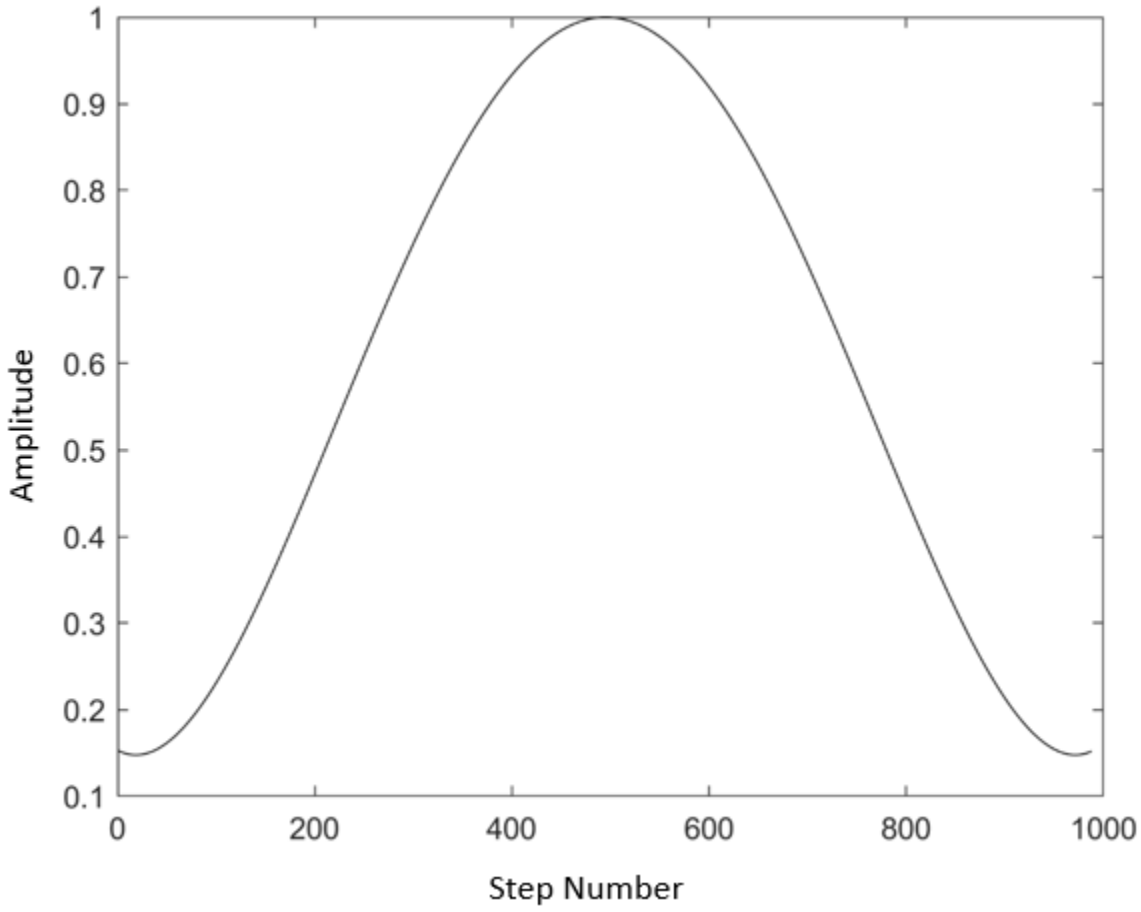

Figure 10: Norton-Beer Medium Apodization Function Generated for a Sample Interferogram

The raw interferogram array is multiplied by the apodization function array. The result of applying the apodization function (Figure 10) to a raw interferogram (Figure 11) is shown in Figure 12.

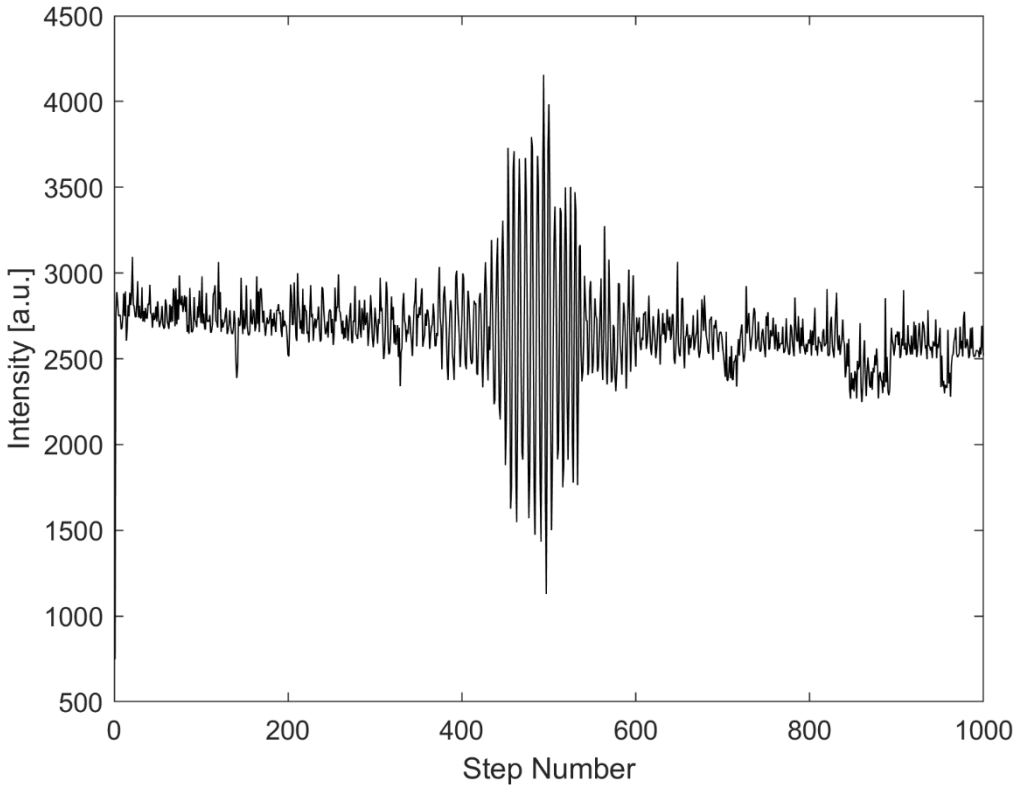

Figure 11: Raw Interferogram Before Apodization

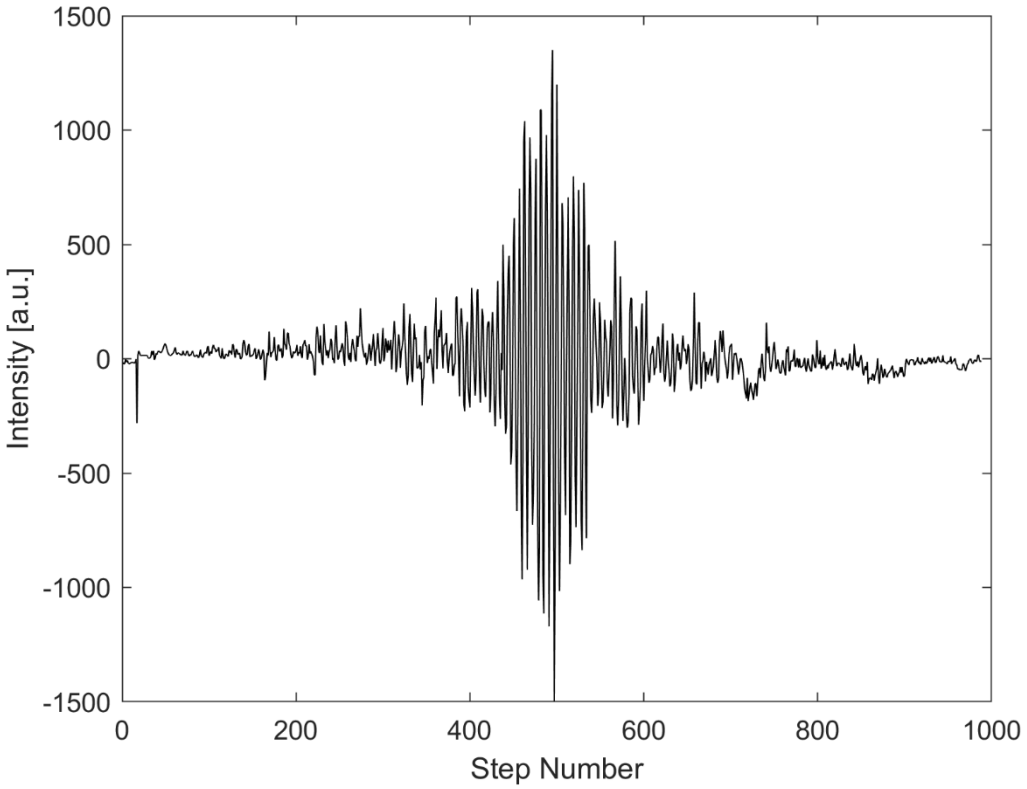

Figure 12: Resulting Apodized Sample Interferogram

By observing Figure 12, the side lobes appear to be minimized while the center of the interferogram remains unmodified, which is the purpose of the apodization function. As briefly mentioned earlier, this is done to minimize noise in the side lobes, which would otherwise result in increased error. After the sample interferogram is apodized, the non-uniform Fourier transform is performed on the resulting interferogram via the NUFFT algorithm, resulting in real and imaginary complex terms.

These resulting complex terms are used to correct for phase angle using Equations 6 and 7.

$$\theta_v = \tan^{-1}\left(\frac{Re(\tilde{v})}{Im(\tilde{v})}\right) \tag{6}$$

$$B(\tilde{v}) = Re(\tilde{v})\cos\theta_{\tilde{v}} + Im(\tilde{v})\sin\theta_{\tilde{v}} \tag{7}$$

The calibration results of the reference laser spectrum after experimental error correction is shown in Figure 13, as well as additional calibration results of a band pass filter using this procedure, is shown in Figure 14

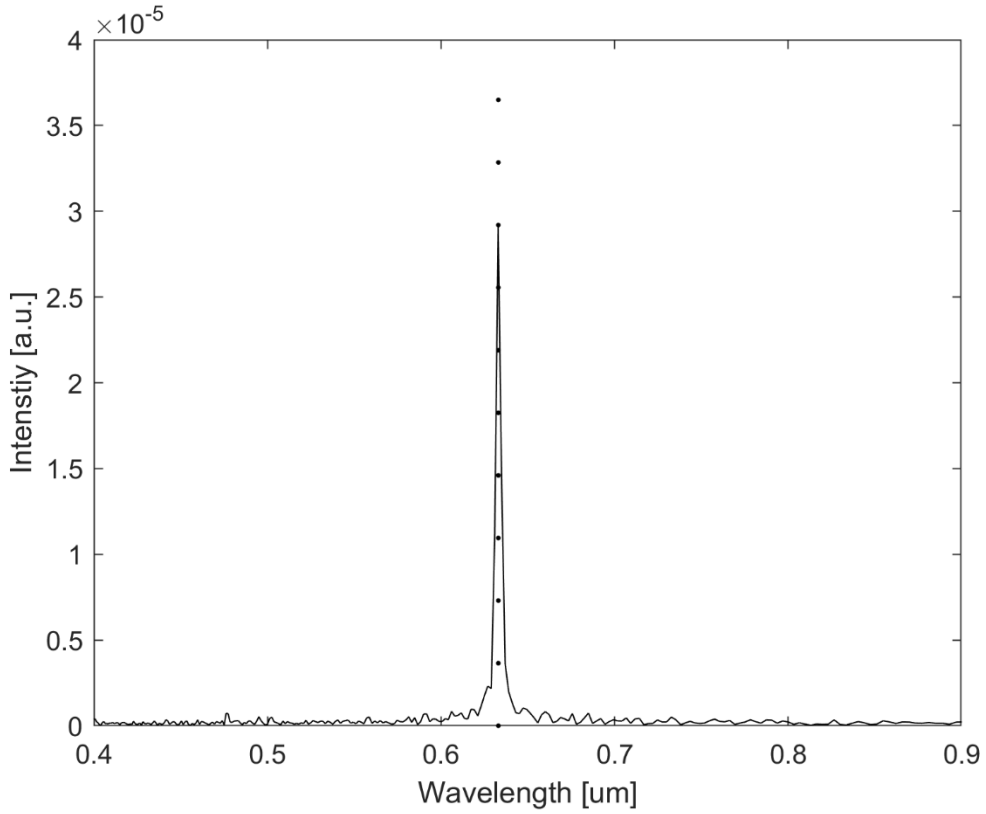

Figure 13: Resulting HeNe Laser Spectrum found from Experimental Noise Corrected HeNe Laser Interferogram

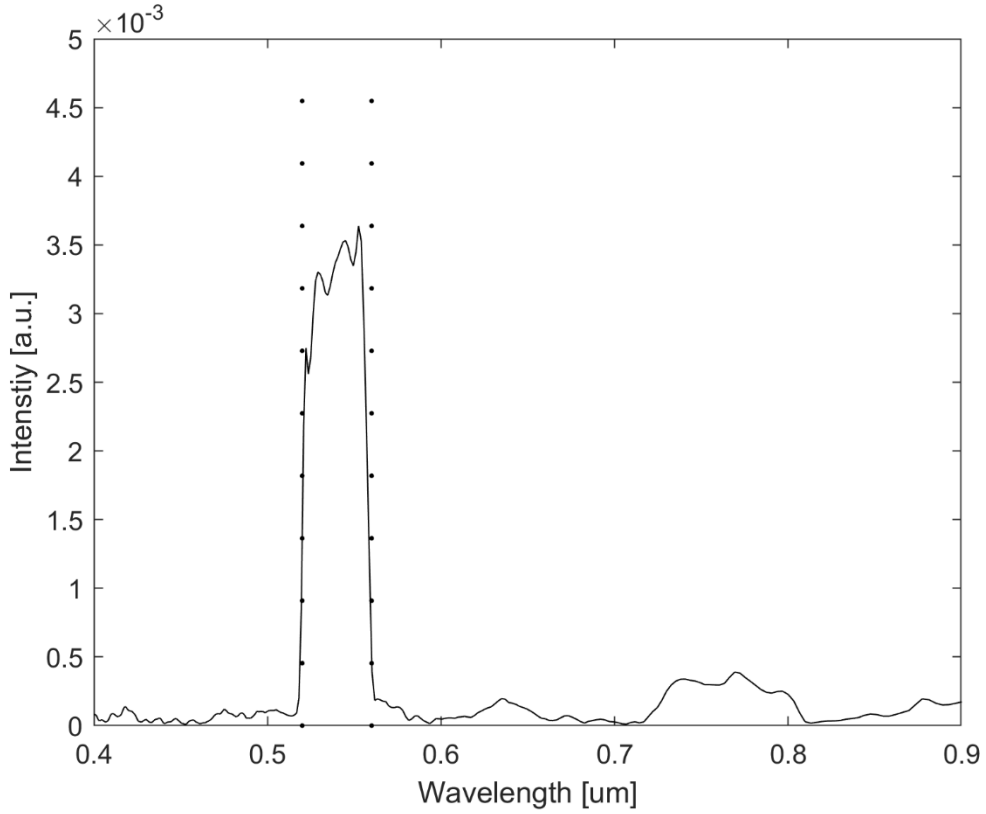

Figure 14: Resulting Spectrum of a 520nm-560nm Band Pass Filter used for Procedural Calibration

Chapter 3 – Deep Learning Methods

3.1. Deep Learning Introduction

Deep learning is a machine learning algorithm featuring the artificial neural network consisting of several layers. In classical programming, an engineer codes rules and logic to find a solution given a data set, shown on the left in Figure 15. Deep learning works by "training" on data with known answers which is shown on the right in Figure 15. This allows the code to learn its own model, which can then be used to find the solution to a piece of data without a known value. Figure 16 shows why deep learning was selected over other conventional machine learning techniques.

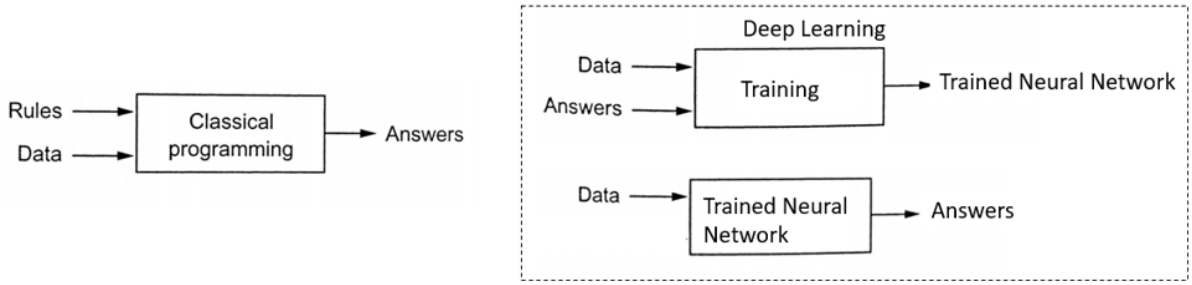

Figure 15: Classical Programming (Chollet, 2018) (left), Compared to Deep Learning (right)

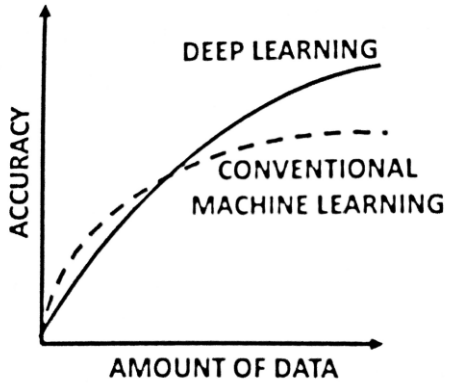

Figure 16: Comparison of Deep Learning and Traditional Machine Learning Performance (Aggarwal, 2018)

By observing Figure 16, we see that deep learning has been shown to result in better performance compared to other machine learning methods assuming sufficient data is available. The biggest drawback of deep learning is that it does notoriously require a large amount of data compared to other machine learning techniques. However, here our experimental data are images of samples which each contain a large number of pixels, and each pixel corresponds to one datapoint. This makes it easy for us to collect a large amount of data required for deep learning.

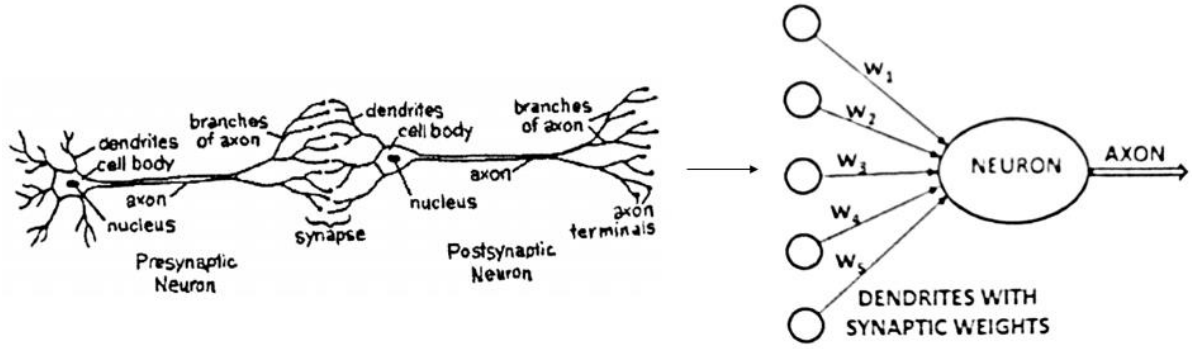

Figure 17: Comparison of Biological Neural Network (Left) and Artificial Neural Network (Right) (Aggarwal, 2018)

The design of neural networks was inspired by the human brain. On the left of Figure 17, we see the biological neural network represented by the interaction between two neurons. On the right of Figure 17 we see a diagram of an artificial neural network showcasing a single neuron with synaptic weights connecting to other neurons.

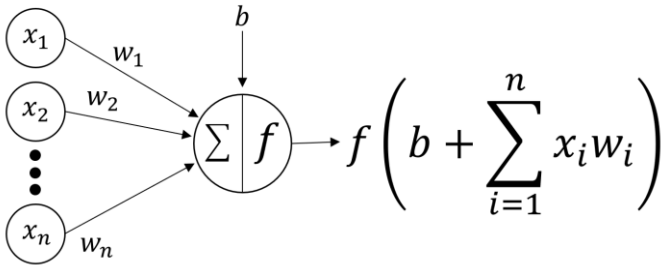

Figure 18: Mathematical Calculation of the Value of a Single Node (Aggarwal, 2018)

Mathematically speaking, a value is computed at each neuron, which is called node in deep learning, as shown in Figure 18 (Aggarwal, 2018). These values travel downstream through several layers, which implies the word “deep” in deep learning, until the values reach an output layer. Figure 19 shows an example diagram of a fully connected neural net which is comprised of a few layers of nodes, resulting in a value at the output layer.

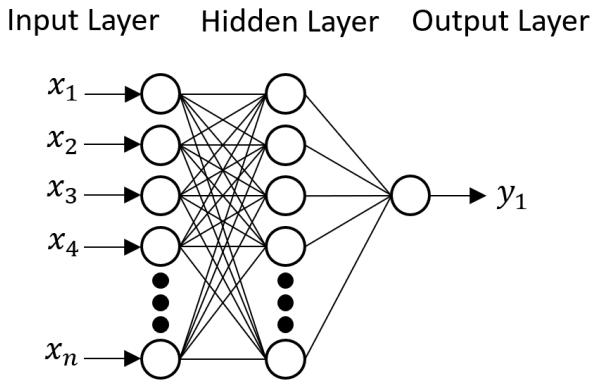

Figure 19: Fully Connected Neural Network (FCNN) Comprised of Several Individual Nodes

The value (or values) at the output layer can be used for various tasks including the classification of a hand-written digit and the estimation of the future value of real estate (Chollet, 2018).

Specifically, the type of neural network that we use is called a convolutional neural network (CNN). The CNN was inspired by the connection pattern between neurons of the human visual cortex. Layers convolve the input and pass the result to the next layer, this works similarly to the fully connected neural net, except with the addition of the convolution. We use CNN specifically because they are shift invariant and have been shown to be very accurate with pattern recognition (Chollet, 2018).

Neural networks functionally “learn” by using training data with known answers, as illustrated in Figure 20. The data is inputted to the neural network, and the values computed at the nodes (using weights) travel through the intermediate layers until resulting in a prediction from the output layer. The

predicted values are compared with the known values, by using a loss function. The resulting loss is used by an optimization algorithm to update the weights. This process is repeated until the loss is minimized.

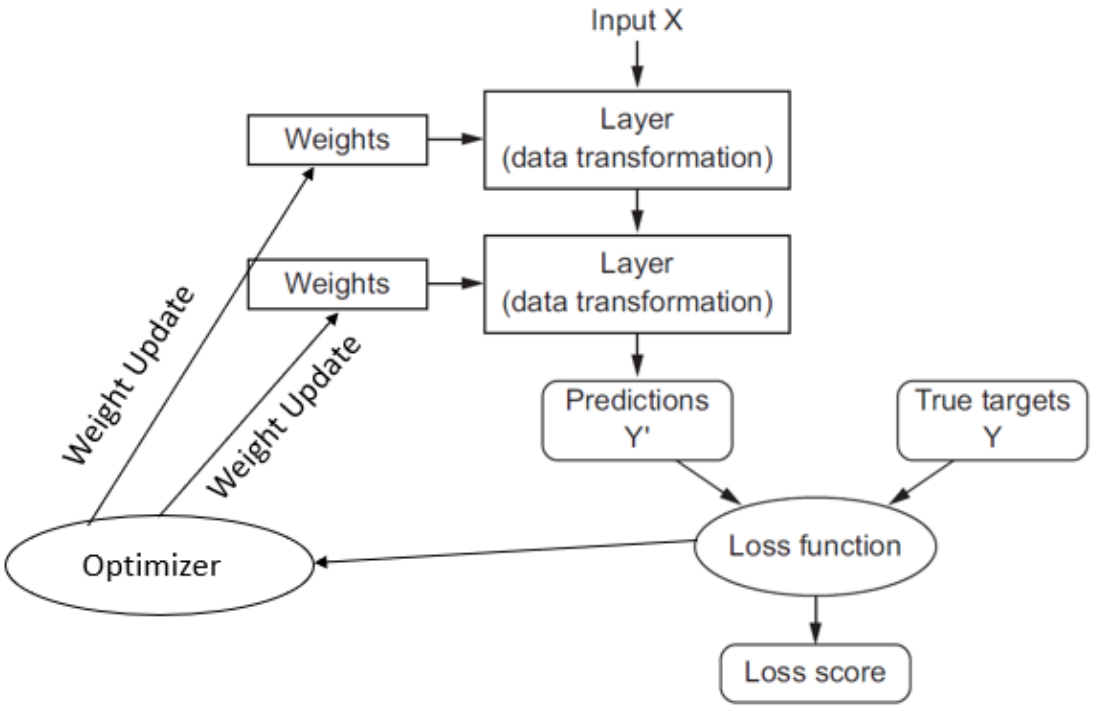

Figure 20: Neural Network "Learning" Process (Chollet, 2018)

3.2. Neural Network Architecture

An important part of the simulations and experiments is the neural network architecture. Figure 21 shows the neural network architecture used in this study. The interferograms are fed into the neural network shown on the left of Figure 21. Next, the data passes through several convolutional layers with increasing kernel size and ReLU activation functions. The resulting values pass through a fully connected neural network with decreasing number of nodes, until finally the values reach the output layer.

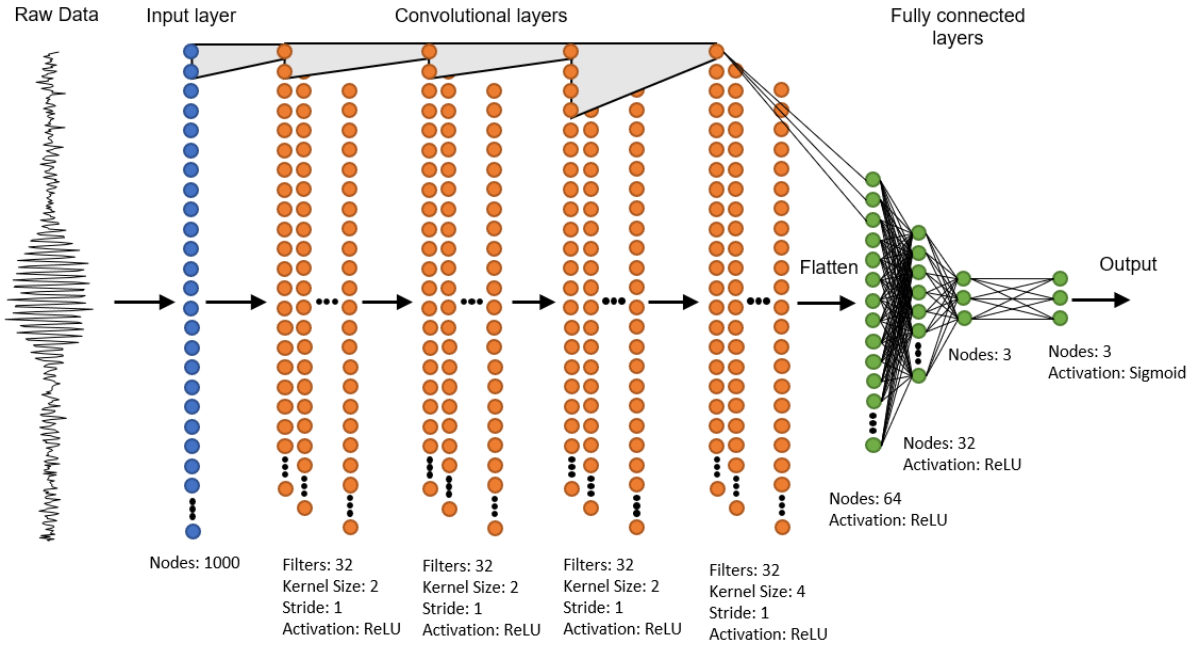

Figure 21: Neural Network Architecture

Several types of binary, linear, or non-linear activation functions can be applied to the output of nodes of a neural net. Here we use the rectified linear unit (ReLU) activation function, which is regarded as the most popular activation function in deep learning (Chollet, 2018). To describe the ReLU function, two representations are shown. An equation is shown in Equation 11 and graphical representation is shown in Figure 22.

$$f(x) = max\{0, x\} \tag{11}$$

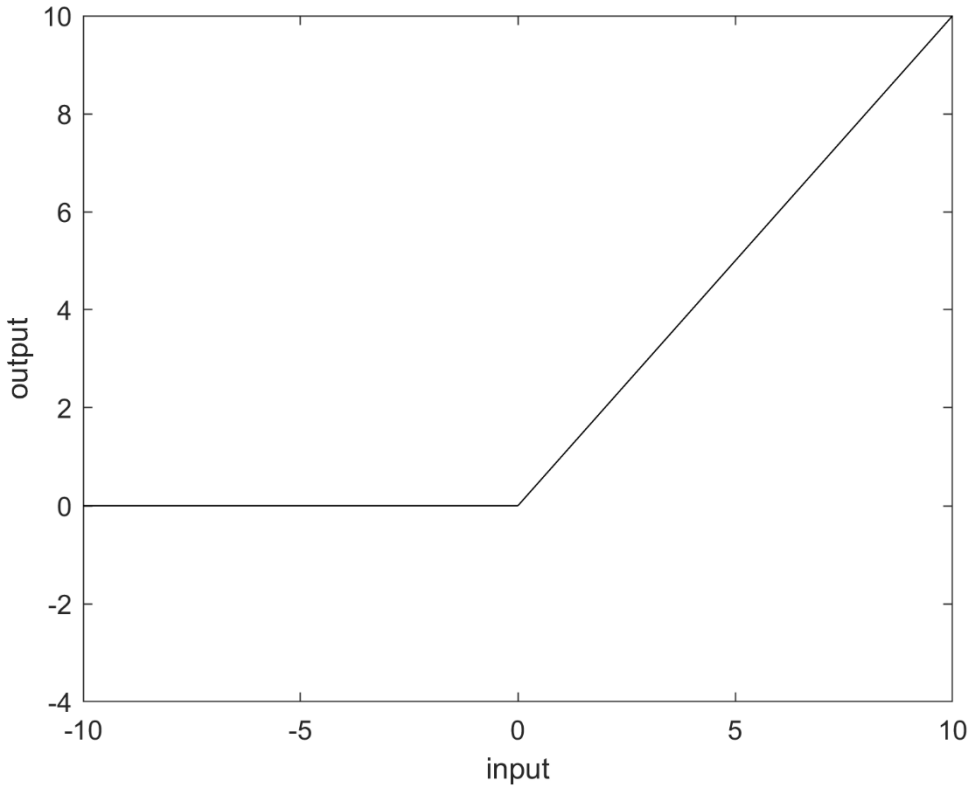

Figure 22: ReLU Activation Function

By observing the ReLU activation function in Equation 11 and Figure 22, we see that functionally it takes an input and if it is negative, it returns zero, otherwise it returns the input. This function prevents a node from being negative, instead setting the output to zero resulting in what is referred to as model sparsity. Compared to other activation functions, the ReLU activation function is computationally cheap to compute (as opposed to tanh for example) and allows the neural network to converge faster. There are other modified versions of the ReLU activation function such as the leaky ReLU which outputs a small negative value for negative inputs, as well as the parametric rectified linear unit (PReLU) which is similar to the leaky ReLU, except that the negative slope is parameterized.

Another type of activation function used specifically at the output layer is the sigmoid activation function. This activation function outputs a value between 0 and 1.

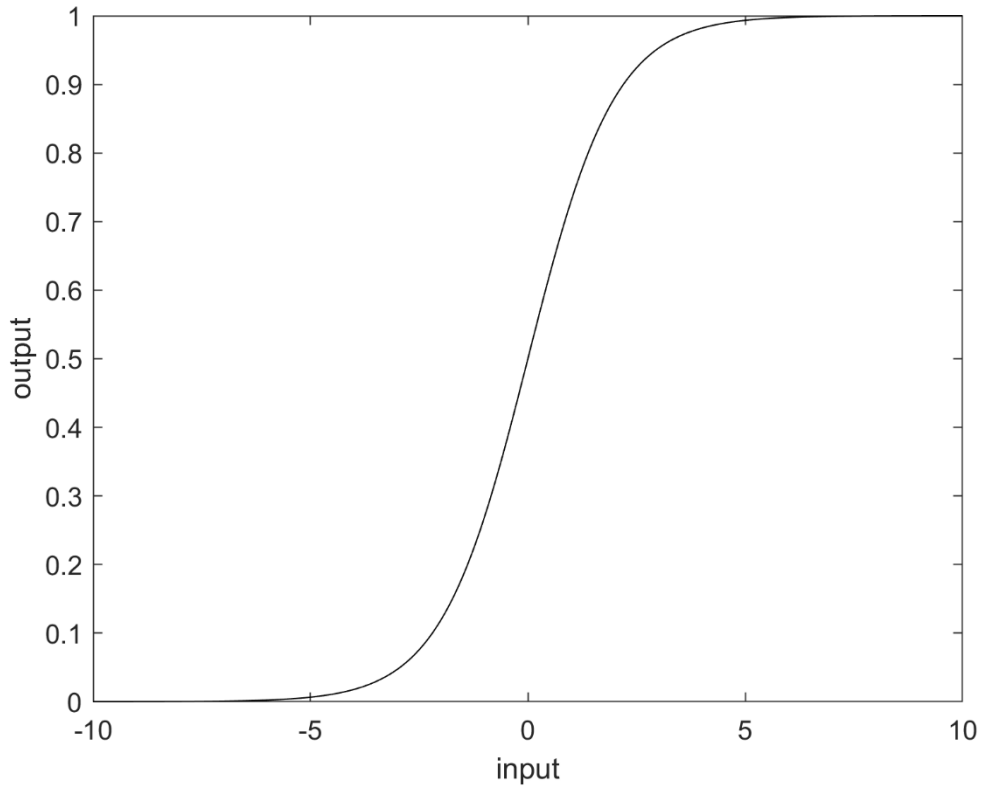

Figure 23: Sigmoid Activation Function

In this work, we use L2 regularization which is also called ridge regression or weight decay. L2 regularization adds the squared magnitude of a coefficient as a penalty term to the loss function. This helps with keeping weights small, which reduces overfitting and increases repeatability with randomly selected starting weight values. Along with the benefits of reducing overfitting and increasing repeatability, regularization helps to reduce test error, potentially at the cost of increased training error (Goodfellow, Bengio, & Courville, 2017).

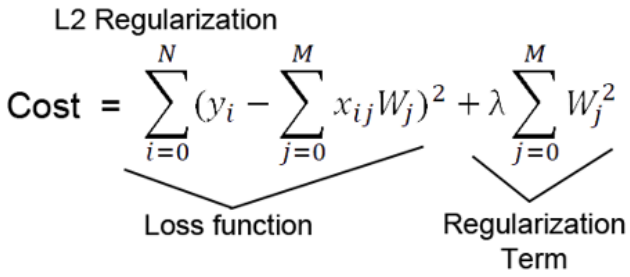

Figure 24: Regularization Term Added to the Loss Function

3.3. Numerical Simulation

As a proof of concept and to test the potential limitations of the selected methods, two numerical simulations were conducted. One simulation was to modify a dataset known as the MNIST dataset (LeCun, Cortes, & Burges, 2010) so that it contained spectral information similar to a multi-spectral sample stained with a fluorescent dye. The other numerical simulation conducted was using experimental data of microscopic fluorescent beads to generate additional fluorescent signals and to find the minimum amount of sampling required based on a specified amount of experimental error.

3.3.1. Hyperspectral MNIST Simulation

To develop our deep learning procedure, we started by modifying the equivalent of the “hello world” of machine learning. The MNIST hand-written digit classification dataset consists of 60,000 grayscale images of the 10 digits, along with a test set of 10,000 images (LeCun, Cortes, & Burges, 2010). For this simulation, the MNIST dataset was modified by giving each one of the hand-written digits a color. Figure 25 shows the colors selected to augment the MNIST data.

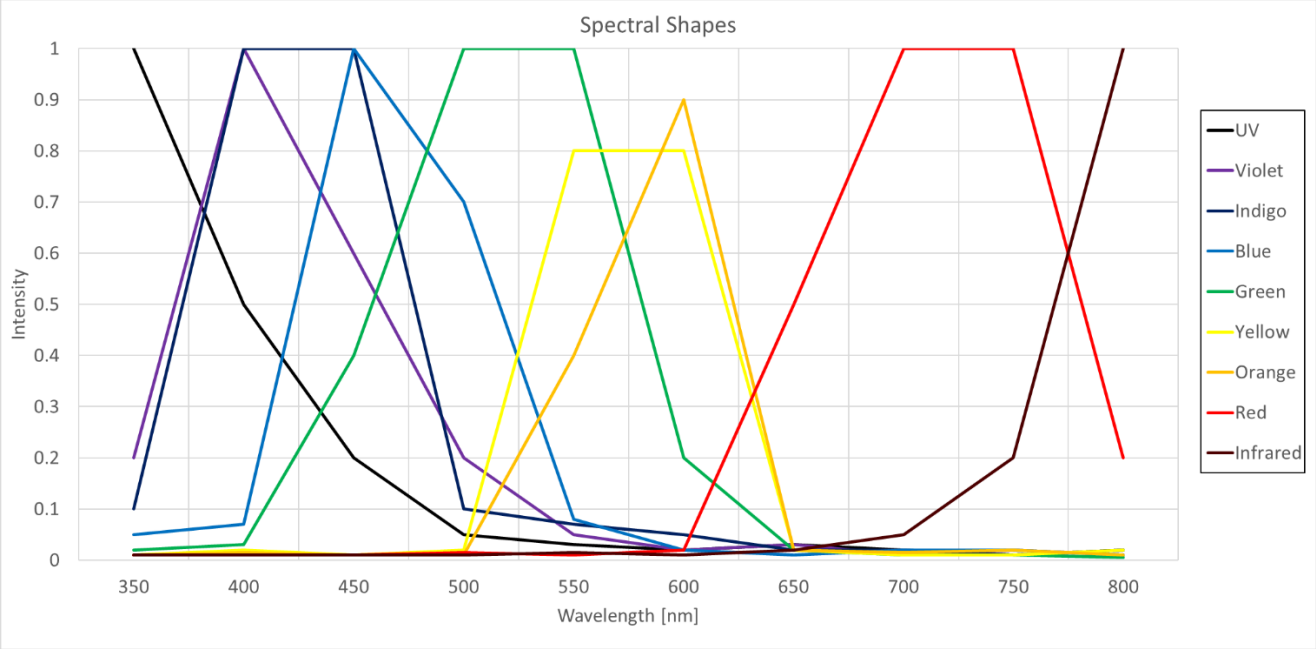

Figure 25: Spectral Shapes Used to Augment MNIST Data

The spectra were selected by coarsely discretizing the emission spectrum for nine different fluorophores. Specifically, they were selected so that there was a wide range of spectral bands as well as some variation of the spectral shape.

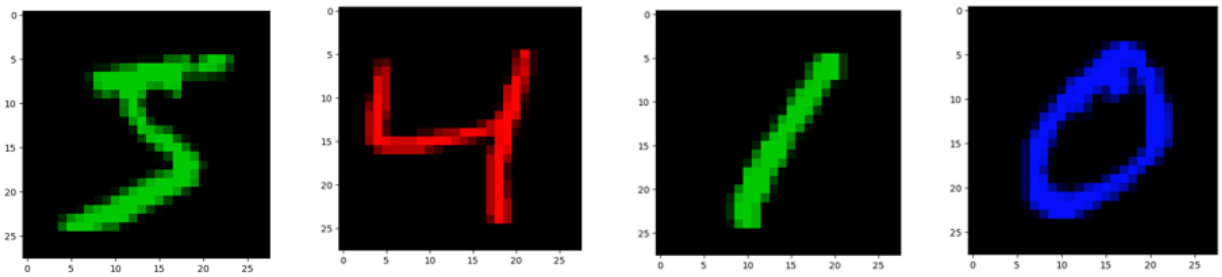

Figure 26: Some Examples of the Multi-Spectral MNIST Data

A simple and small convolutional neural network was developed to classify the 90 different types (10 digits and 9 colors) of the multi-spectral MNIST data. By using a small convolutional neural network, we were able to achieve about 85% accuracy on 10,000 pieces of training data. Figure 27 shows that using the same neural network, more data results in higher accuracy. At the same time, we also note that there is a point of diminishing returns where the accuracy stops improving without adjusting the neural

net architecture. The moderate accuracy may be attributed to the small number of training data set, the serious overlap of some simulated spectra, and non-optimal neural network. Still, we were able to confirm the feasibility of classifying multi-spectral data using a convolutional neural network.

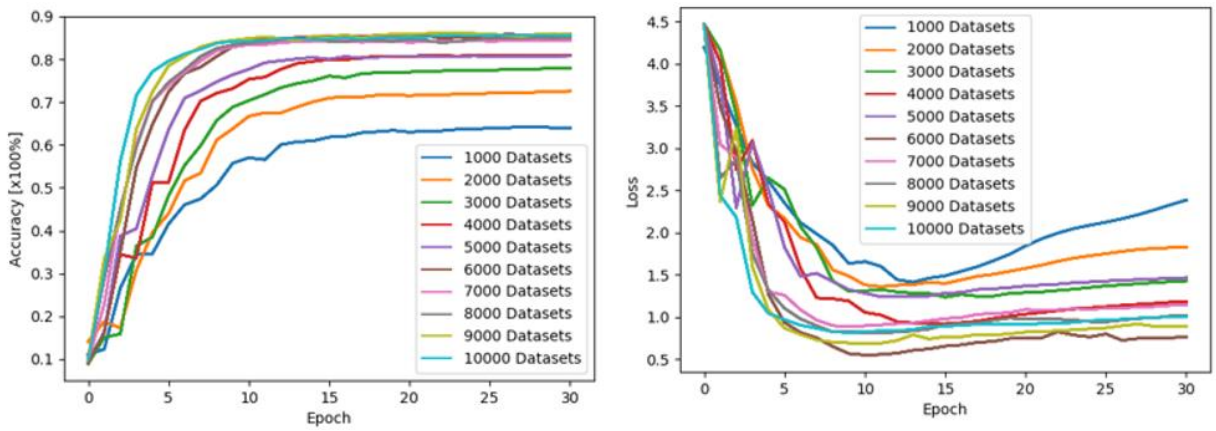

Figure 27: Multi-Spectral MNIST Neural Network Training Results

3.3.2. Multi-Spectral Bead Simulation

For confirmation that the physical experiment could be feasible, simulations were set up and conducted. Simulated data was generated to approximately match the emission peaks of the Thermo Fisher Invitrogen F36913 FocalCheck fluorescence microscope test slide #2, which was purchased and planned to be used for the physical experiment. This test slide contains 8 different types of fluorescent beads separated into 4 sections with a 5^{th} section containing a mixture of all eight bead types. The emission peaks are: 511nm, 524nm, 555nm, 565nm, 605nm, 613nm, 676nm, and 692nm. The simulation is very easy to upscale to 10 or more emission peaks, however on the physical experiment side, it would require additional considerations such as the choice of the excitation LEDs and the fluorescence filter set.

The simulated data was created by generating a distribution of experimental noise, a HeNe interferogram subjected to experimental noise, and a sample interferogram subjected to the same experimental noise. For the translation stage noise, a normal distribution (shown in Equation 12) with a

standard deviation of 10% of step size and a mean of 0nm was selected due to similarity between noise found on experimental results and simulated results.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{12}$$

Thermal stability noise was added in a similar way to the intensity values, and Poisson noise representing noise caused by a low SNR was also added.

Figure 28 and Figure 29 are shown as a comparison between an experimentally measured and corrected HeNe interferogram and a HeNe interferogram generated with experimental noise and corrected with the same algorithm.

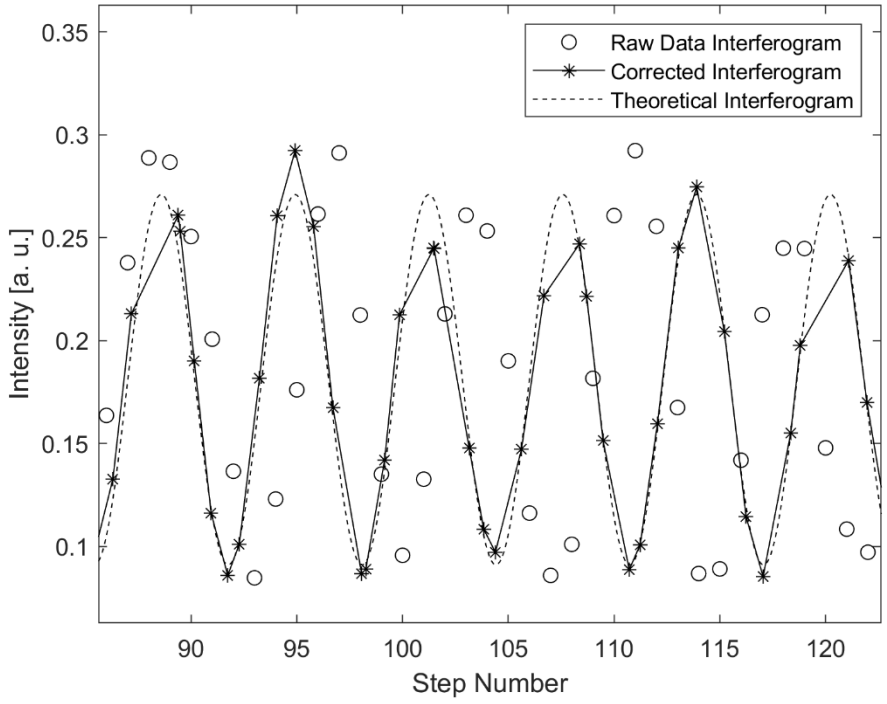

Figure 28: Experimentally Collected and Corrected HeNe Interferogram

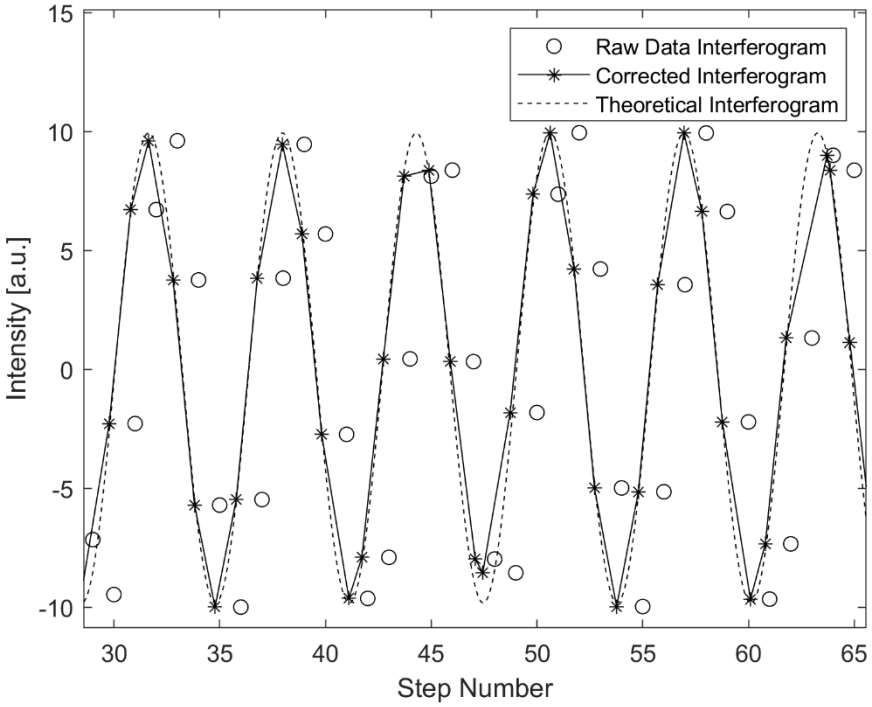

Figure 29: HeNe Interferogram with Simulated Translation Stage Error

This results in the simulated interferogram shown in Figure which looks comparable to the equivalent experimentally sampled interferogram shown in Figure 31.

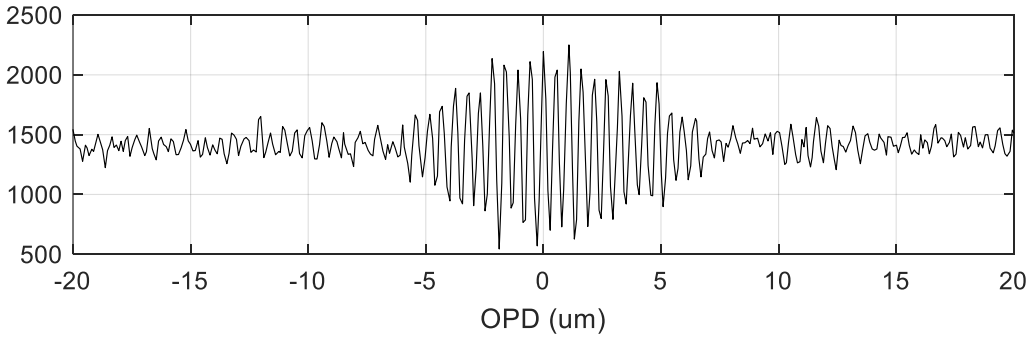

Figure 30: Interferogram Generated with Simulation

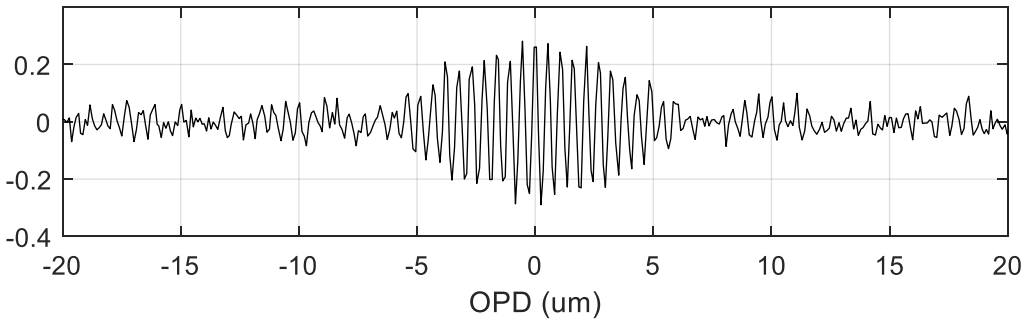

Figure 31: Experimentally Measured Interferogram

The HeNe interferogram was generated using a slightly modified version of Equation 2, shown in Equation 13, where A is the amplitude, B is the vertical offset, C is the horizontal offset, and $x_e(i)$ is the position plus the translation stage error from the generated distribution of noise at each point.

$$I_{HeNe} = A\cos\big(2\pi\omega(x_e(i) - C)\big) + B \tag{13}$$

As observed in Figure 28 and Figure 29, the HeNe interferogram is then corrected to eliminate the experimental noise, using the same algorithm for the simulated data which was developed for the experimental data.

The interferogram for a simulated bandpass filter sample was initially generated using Equation 14, where k_0 and k_1 are the starting and ending peak wavenumbers, respectively.

$$I_{sample} = \frac{1}{\pi x_e} * (\sin(2\pi k_0 x_e) - \sin(2\pi k_1 x_e)) \tag{14}$$

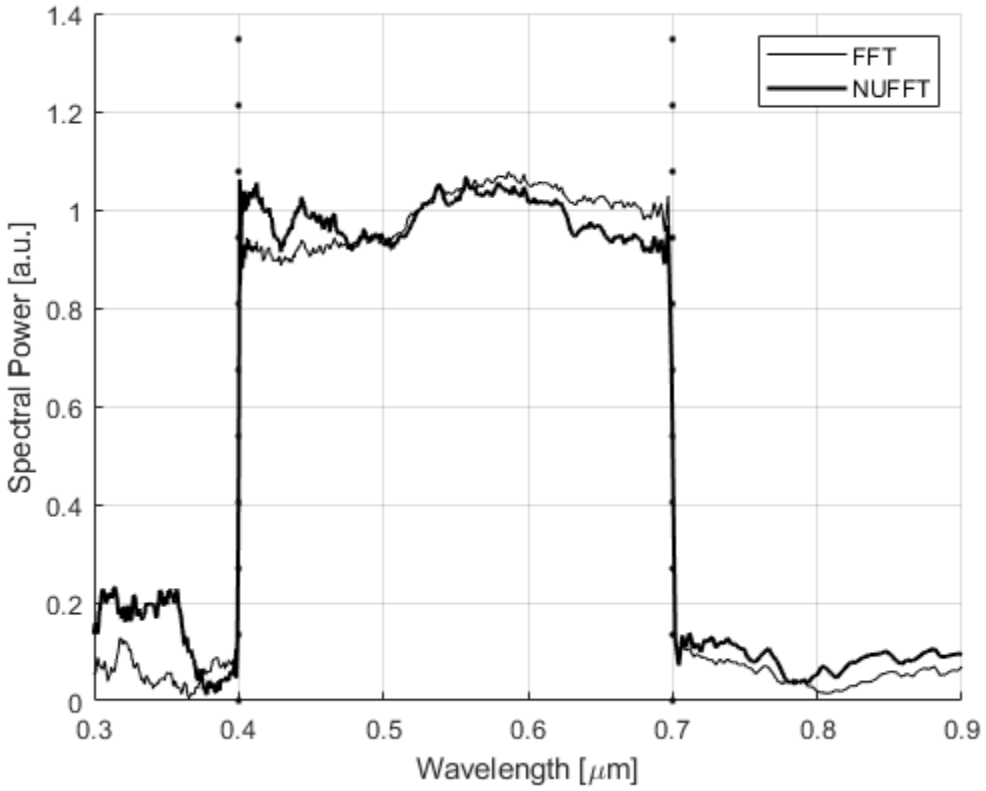

Figure 32: Comparison of FFT and NUFFT Spectrum of Simulated Bandpass Filter with Simulated Noise

In Figure 32, Equation 14 was used to generate the interferogram of a bandpass filter, with some simulated noise. Later in this paper, the error is measured and analyzed more rigorously, but for these initial results the noise added to the OPD at each step was randomly sampled from a normal distribution with a mean of 0 and a standard deviation of 0.02 micron. The vertical points show the band passed by the bandpass filter. We see that a bandpass filter spectrum can be computed by generating a simulated interferogram, however, to match the spectral profile of the bead more closely a gaussian curve was used. For the sample interferogram of each bead type, the simulated interferogram bandwidth was set equal to the FWHM of the experimentally collected and processed beads as shown in Figure 33.

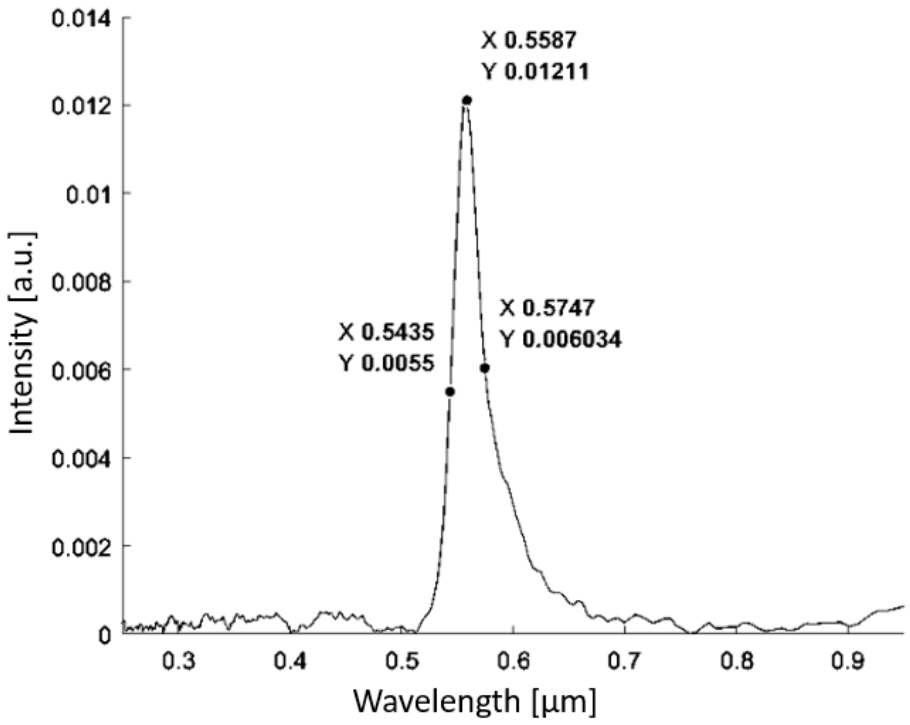

Figure 33: Spectral Profile of FocalCheck Bead used to Compute FWHM for Gaussian Simulated Data

For each bead type, 1000 sets of HeNe and sample interferograms were generated, and processed. The resulting spectral arrays were then exported from MATLAB and imported into Python.

Here the data starts out as a 3D array with dimensions of bead type, sample number, and spectral results. This array is converted into a 2D array with dimensions of sample number and spectral results, and the samples from each bead type are mixed. The spectral results are normalized by the maximum value of the data, so that the spectral values of each datapoint are between 0 and 1.

The ground truth labels for the entire dataset were specified by integers denoting which sample each dataset belonged to. These labels are one-hot encoded by converting the integer labels into binary vector labels. If the label is type one, the first element in the vector would be a one and the rest zeros. Likewise, if a label is type two, the second element in the vector would be a one while the rest would be zeros. One-hot encoded labels are created from the bead types, and the data is split into a training set,

and a test set. As the data was generated incrementally by sample type, the data was mixed before being introduced to the neural network. Mixing the data ensures that the neural network trains on equal amounts of sample types.

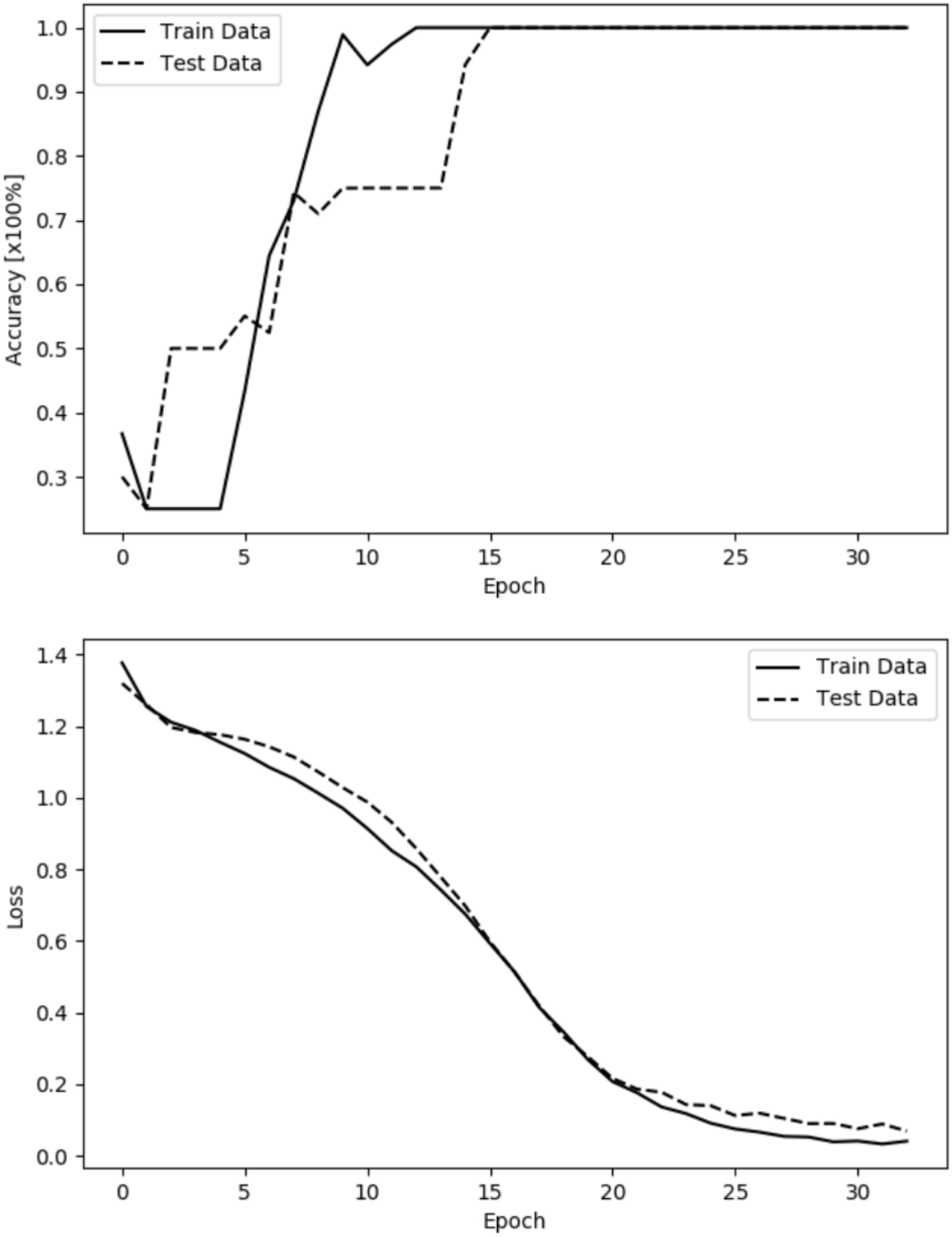

Figure 34: Classification Results of Deep Learning Calibration Experiment

The process of generating, processing, and classifying data was iterated. The data consists of 10 bead types each with 1000 samples, making a total of 100,000 datapoints per trail, for 10 trials each. This data was trained and classified using a standard 1D CNN architecture. The mean and standard deviation results of the corresponding accuracies are shown in Figure 35.

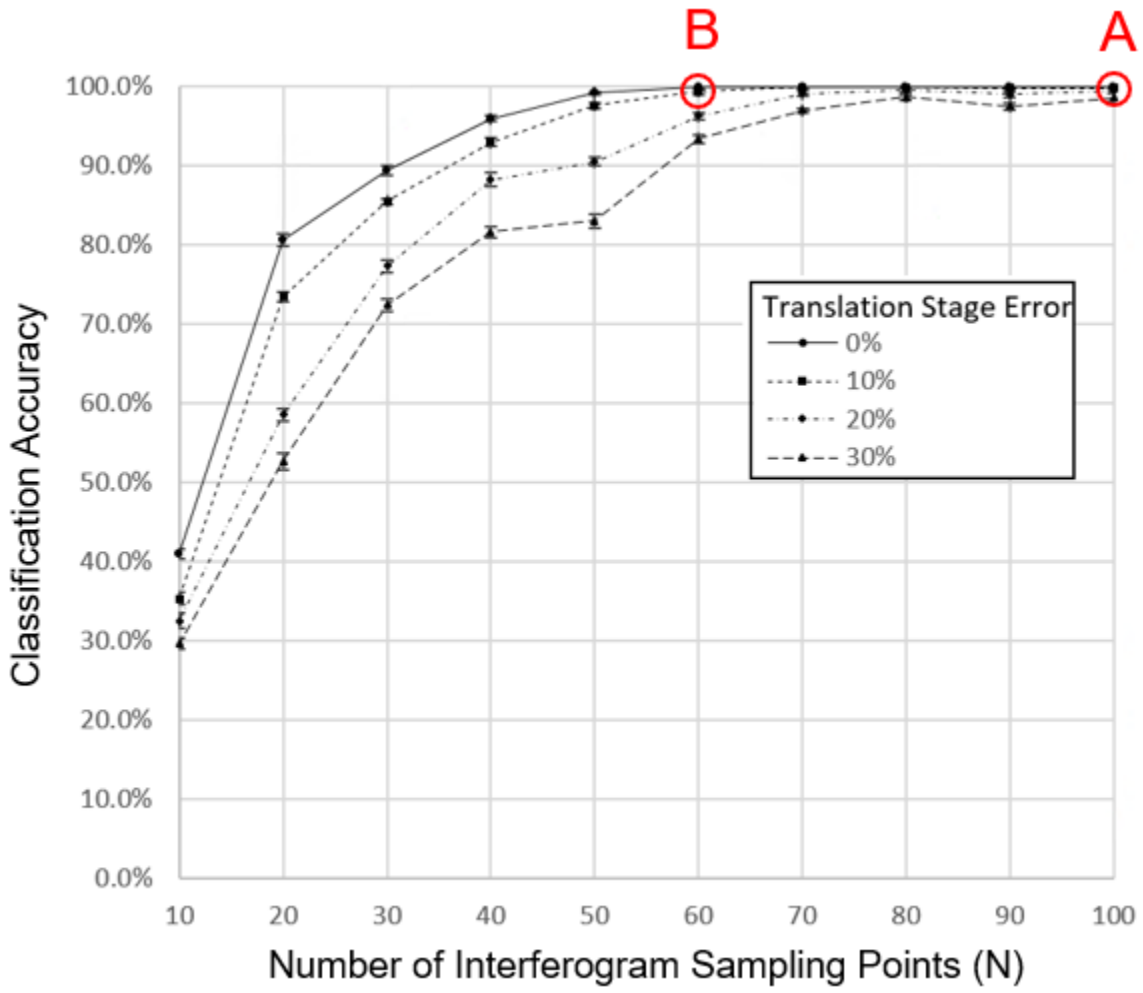

Figure 35: Classification Accuracy of Simulated FocalCheck Bead with Varying Amounts of Interferogram Sampling and Translation Stage Noise

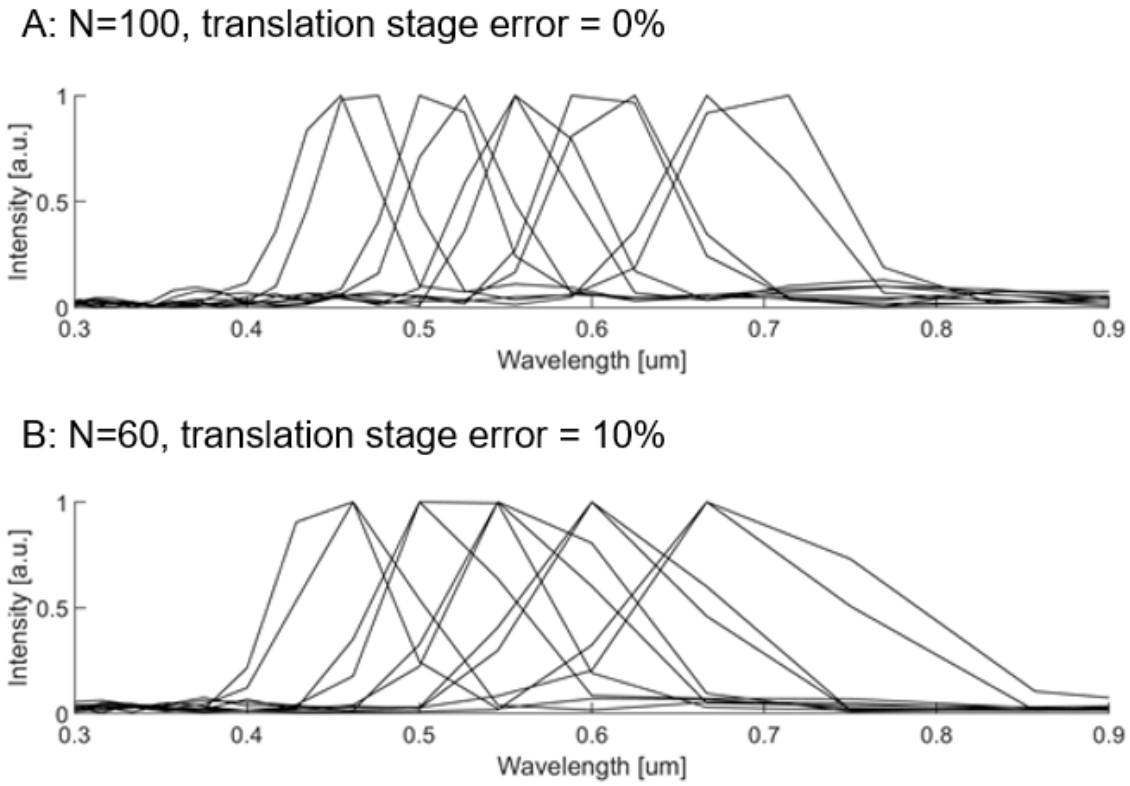

Figure 36: Fourier Transform Based Reconstruction of Bead Spectra

For comparison, the bead spectra corresponding to point A (N = 100, translation stage error 0%) and point B (N = 60, translation stage error 10%) in Figure 35 have been reconstructed using the conventional FTS processing described in Chapter 2. The result is shown in Figure 36. At point A on Figure 35, which is shown on the top of Figure 36, some of the fluorescent beads can be classified visually. However, at point B on Figure 35, shown on the bottom of Figure 36, it is very difficult to distinguish between the different types of fluorescent beads. Importantly, the neural network was able to classify the different types fluorescent beads with near to 100% accuracy for both the cases.

Chapter 4 – Experiments

4.1. Experimental setup

Figure 37 shows the FTS setup used for the experiment. A sample (S) is placed on the coverslip on the left. The sample is illuminated by the excitation light module (EXL) which passes initially through the fluorescence filter cube (FFC).

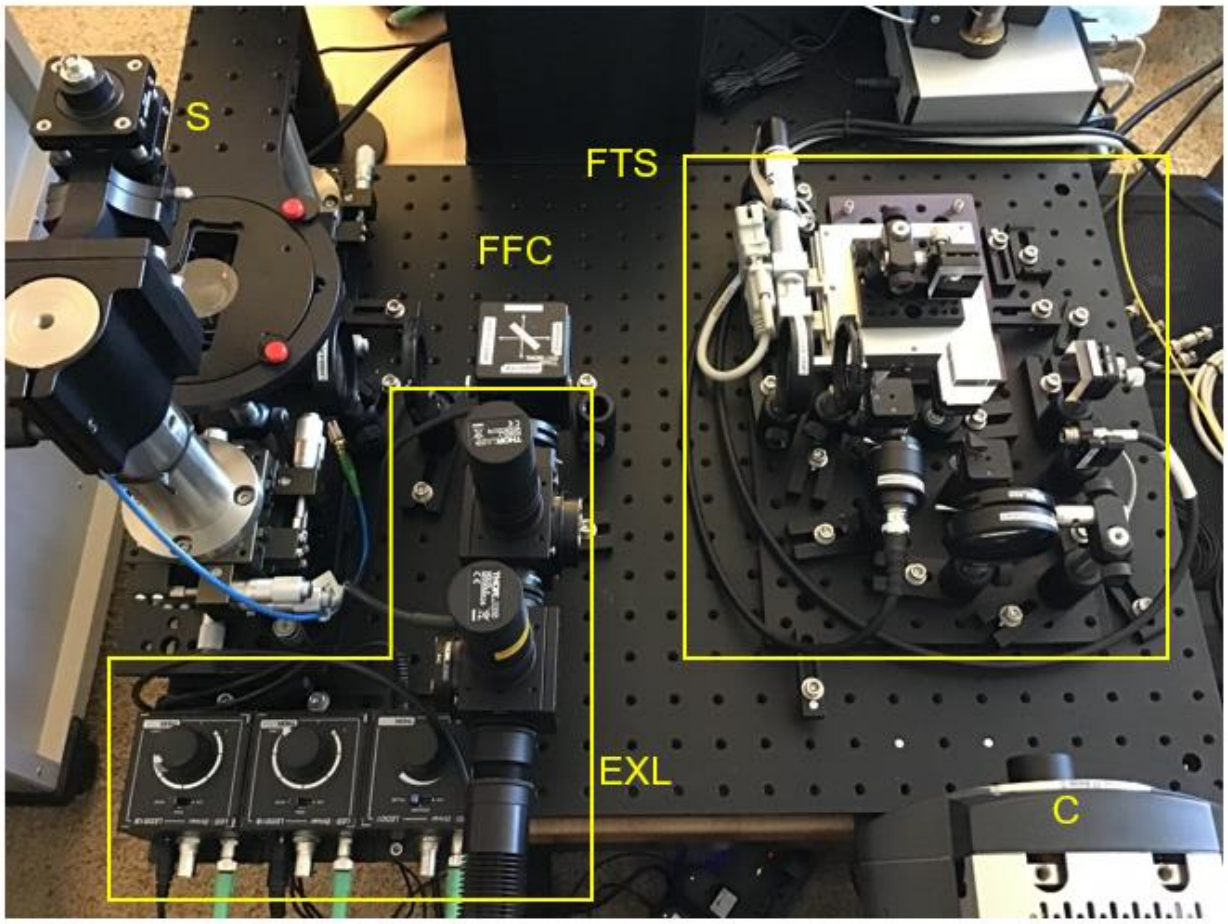

Figure 37: Multispectral fluorescence microscopy based on imaging Fourier-transform spectroscopy (FTIS). S: Sample stage; EXL: Excitation light module; FFC: Fluorescence filter cube; FTS: Fourier-transform spectroscopy module; C: Camera.

Light reflects off of the sample, back through the fluorescence filter cube into the Fourier transform spectroscopy module (FTS), and the resulting interferogram is collected by the camera (C). For the

camera, we used electron-multiplying charge-coupled device (EMCCD) camera (Andor, iXon Ultra 888). To record the reference laser interferogram, we used a silicon photodiode (Thorlabs, SM1PD1A).

A schematic diagram of the excitation light module, which consists of three individually-controlled LEDs (Thorlabs, M385L2, M505L4, and M565L3), is shown in Figure 38. The red, green, and blue light were merged by dichroic beam splitters (Thorlabs, DMLP425R and DMLP550R).

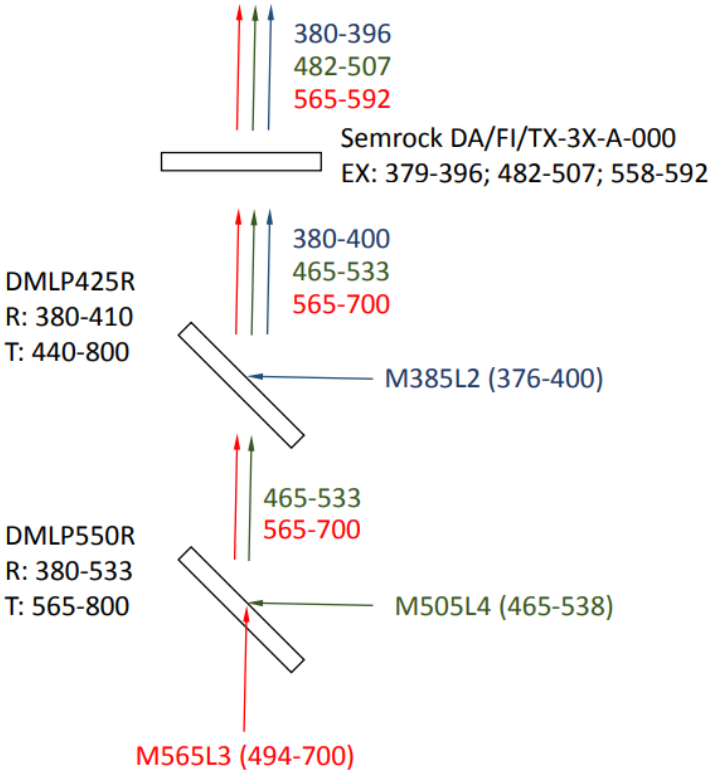

Figure 38: Excitation Light Module (EXL)

The light from the EXL passes through a dichroic filter block(Semrock, DA/FI/TX-3X), or as called here a fluorescent filter cube (FFC). A diagram of the FFC is shown in Figure 39.

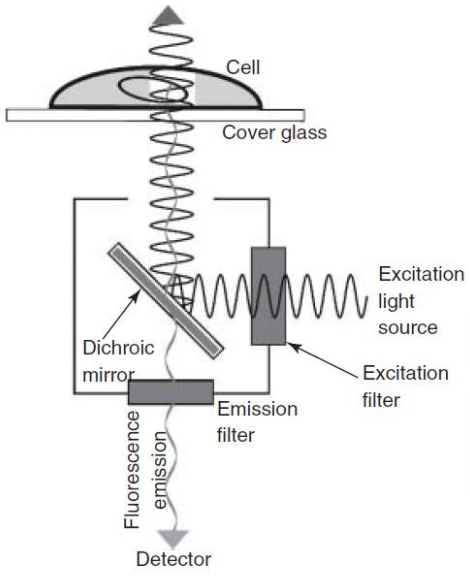

Figure 39: Schematic Diagram of Fluorescence Filter Cube (FFC) (Kubitscheck, 2017)

In Figure 39, the excitation light enters the FFC from the right; this is the light from the EXL. The excitation light reflects off of the dichroic mirror onto the sample. The fluorescent dye in the sample emits a fluorescent signal which passes back to the dichroic mirror. Now, the excited fluorescent light passes through the dichroic mirror and through the emission filter. The resulting signal is passed to the FTS module. Diagrams of each of the excitation and emission light passing through the filters are shown in more detail below. Using only the blue LED which is shown in Figure 40 by the blue dashed rectangle, we only excite the DAPI fluorescent dye.

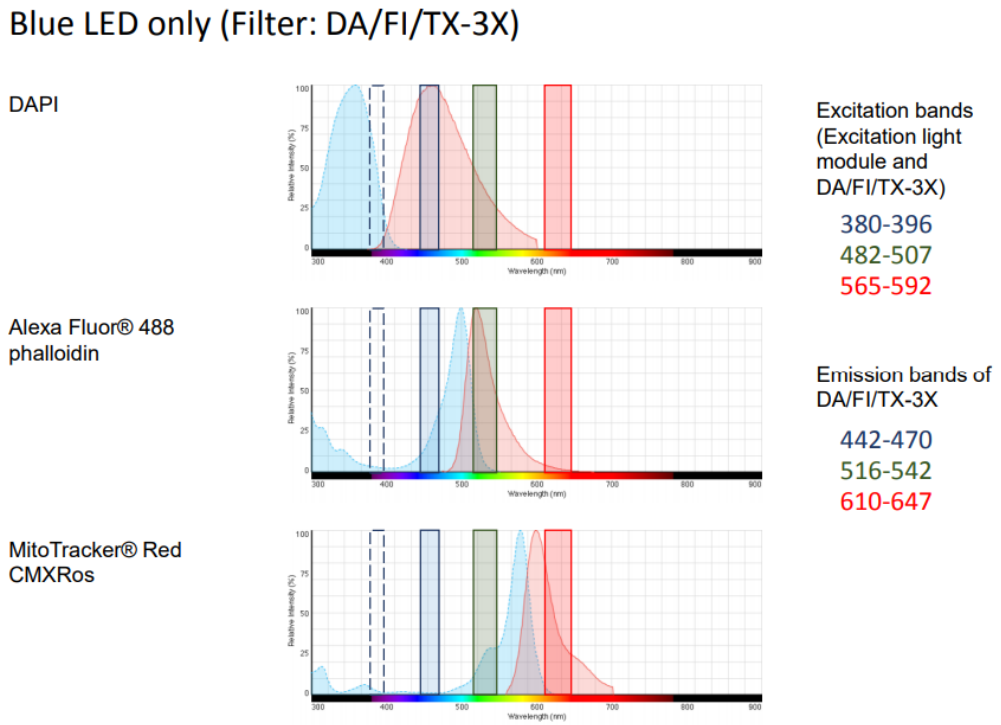

Figure 40: FFC Diagram with Blue LED Only

The interaction between the blue LED and the DAPI is shown in the top of the three subgraphs where the excitation spectrum is shown by the blue curve. The resulting emission spectrum of DAPI is shown on the top subgraph by the red curve. The FFC only passes the bands shown by the solid rectangles. In this case with the blue LED only, the FFC only passes the excited DAPI fluorescent dye.

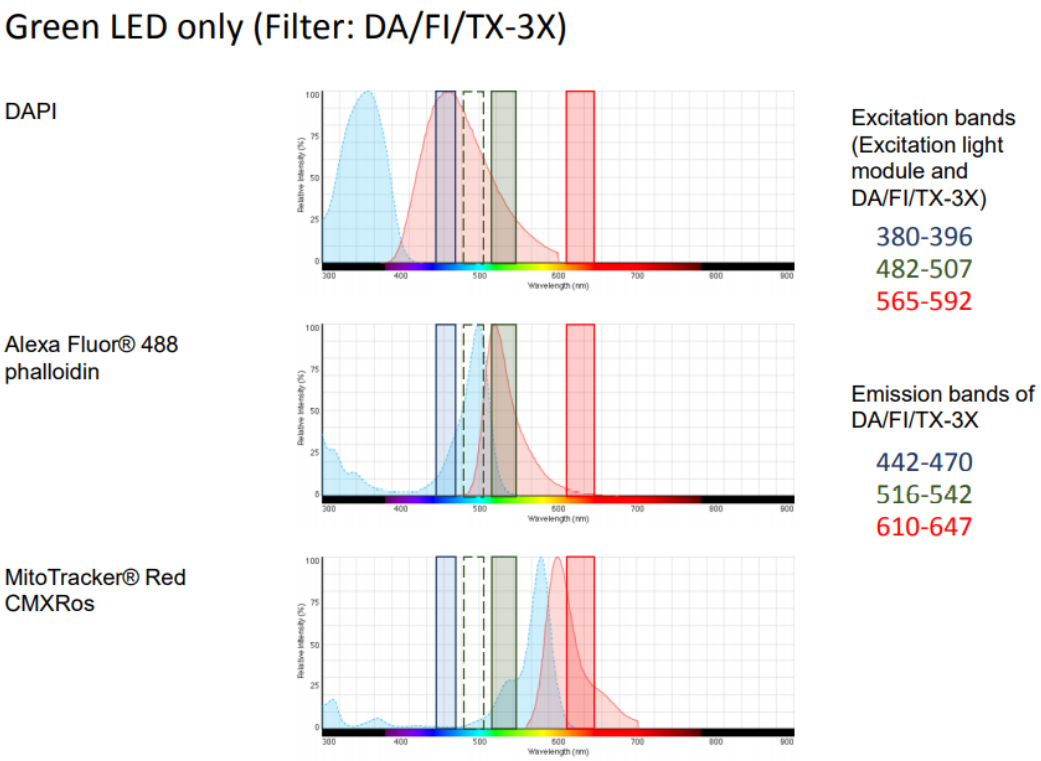

Figure 41: FFC Diagram with Green LED Only

Using the green LED shown in Figure 41 by the green dashed rectangle, only the Alexa Fluor 488 fluorescent dye shown in the middle subgraph is excited. Likewise, the FFC passes the band shown by the solid green rectangle.

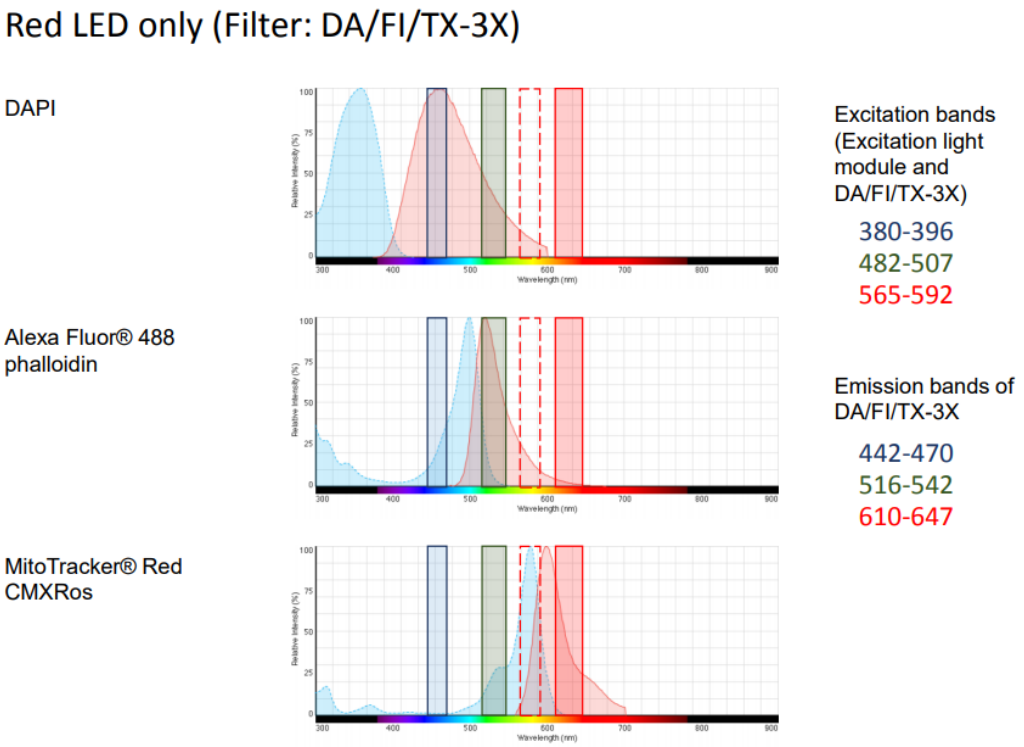

Figure 42: FFC Diagram with Red LED Only

Using the red LED shown in Figure 42 by the red dashed rectangle, only the MitoTracker fluorescent dye shown in the bottom subgraph is excited. Likewise, the FFC passes the band shown by the solid red rectangle.

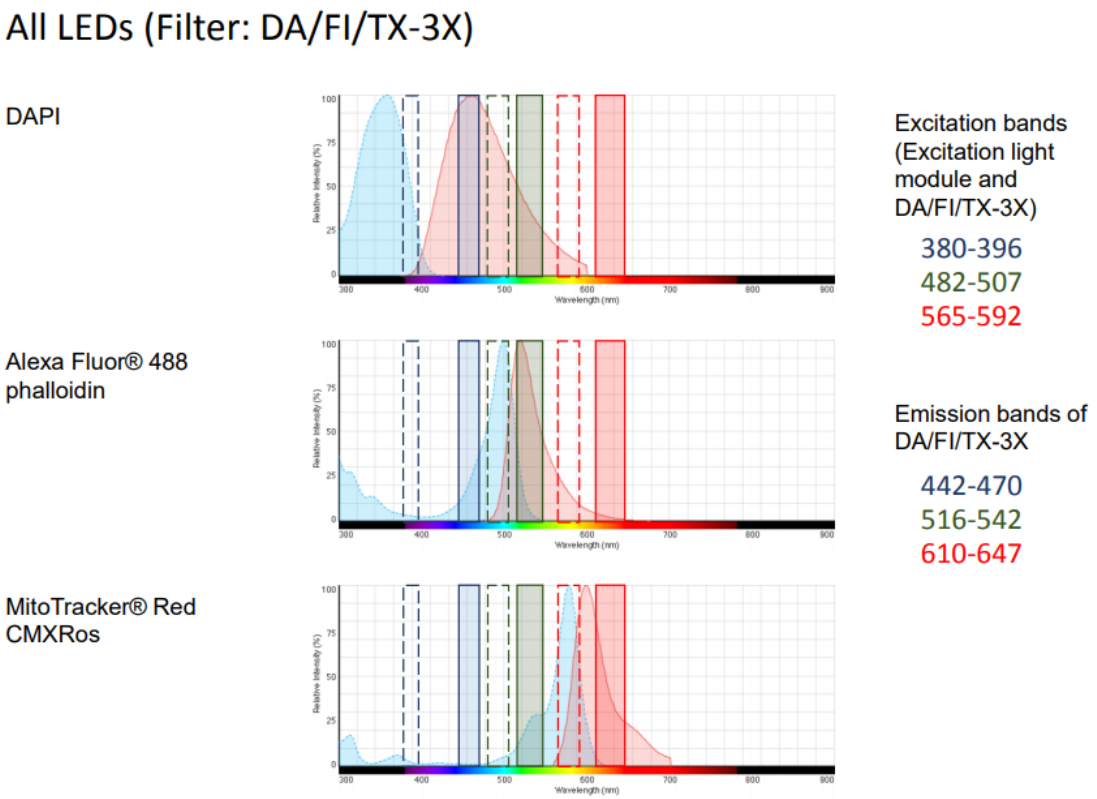

Figure 43: FFC Diagram with All LEDs

In Figure 43 all of the LEDs are used simultaneously as shown by the dashed blue, green, and red rectangles. When combined, all three fluorescent dyes are excited and are passed through the FFC.

4.2. Data Acquisition and Processing

For the sample (S), we used a Bovine Pulmonary Artery Endothelial (BPAE) cell, which was stained with three fluorescent dyes for F-actin, mitochondria, and nucleus (Thermo Fisher, F36924). The LED signal emitted from the EXL through the FFC excites the fluorescent dyes in the sample. The emitted fluorescent signal passes back through the FFC to the FTS module, which was described in Introduction earlier, and the camera collects the resulting images. The images captured at different steps of the moving mirror from the FTS module are the raw interferograms. This results in an interferogram for each pixel in the image, an example interferogram from a single pixel being shown in Figure 44.

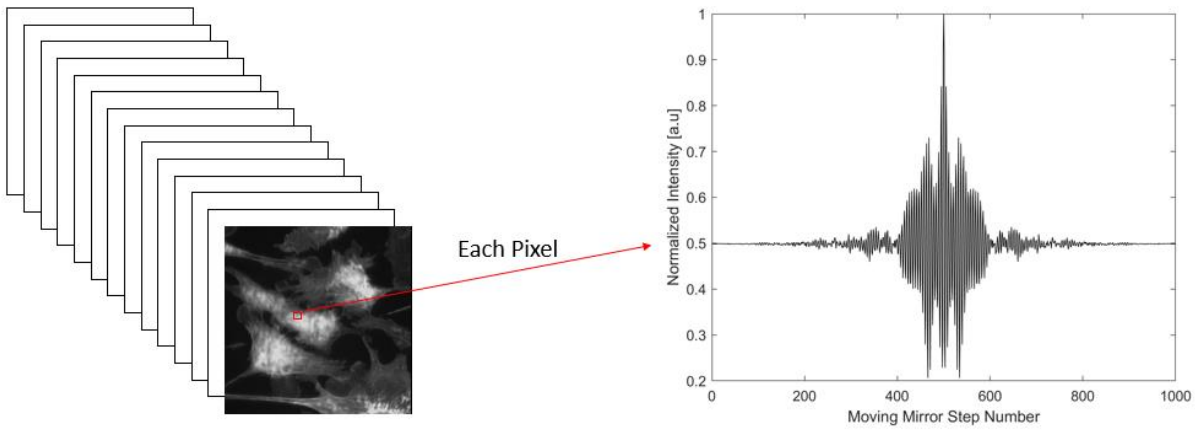

Figure 44: Interferogram Data Acquisition

For the training data, the EXL used only a single LED at a time while collecting interferogram image data. The single-channel data were combined to synthesize the three-color data with arbitrary mixing ratios, which will be described. Each of the single-channel images consisted of pixels of either the background, or the fluorescent signal passed by the FFC. In order to separate the two different types of pixels, a binary mask was created using a manual threshold, as shown in Figure 45.

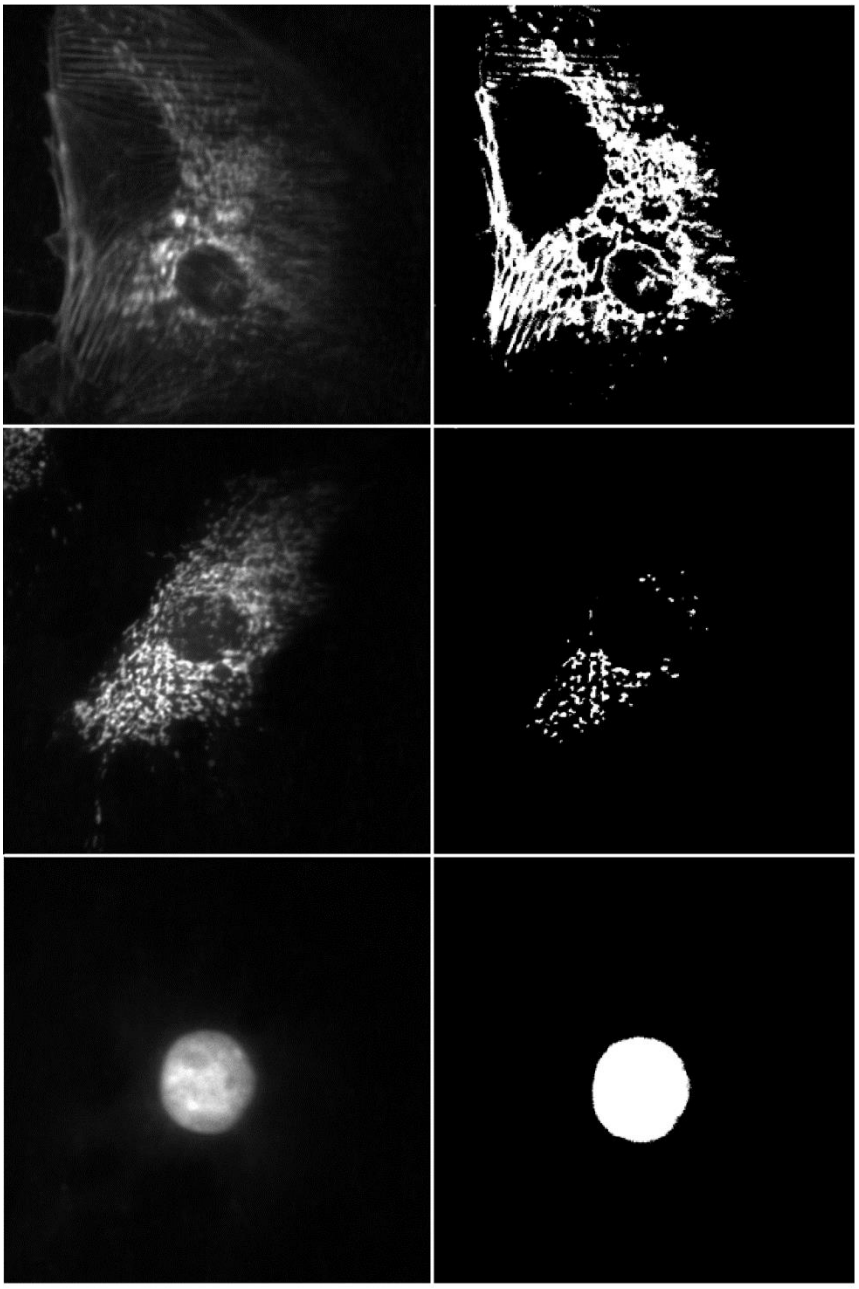

Figure 45: Each Organelle (Left) and Corresponding Identified Location (Right) (Top to Bottom: F-Actin, Mitochondria, Nucleus)

The x and y coordinates of the elements in the binary mask which are equal to 1 are saved to an array. Out of these locations, a set of unique random pixel locations are selected to the amount of the organelle with the least amount of locations in order to balance the training dataset. Then the interferograms for the selected pixel locations are collected by cycling through the images for each selected pixel location, and recording the intensity value.

In order to remove the noise effects from the background, the background signal was subtracted from the organelle signal. This is done by manually selecting a background region of 50x50 pixels in each set of images, then subtracting the average value from each pixel intensity value. After correcting for experimental noise, as described in Chapter 2, we can acquire the raw single-channel spectra (Figure 46), which will be used to train the neural network.

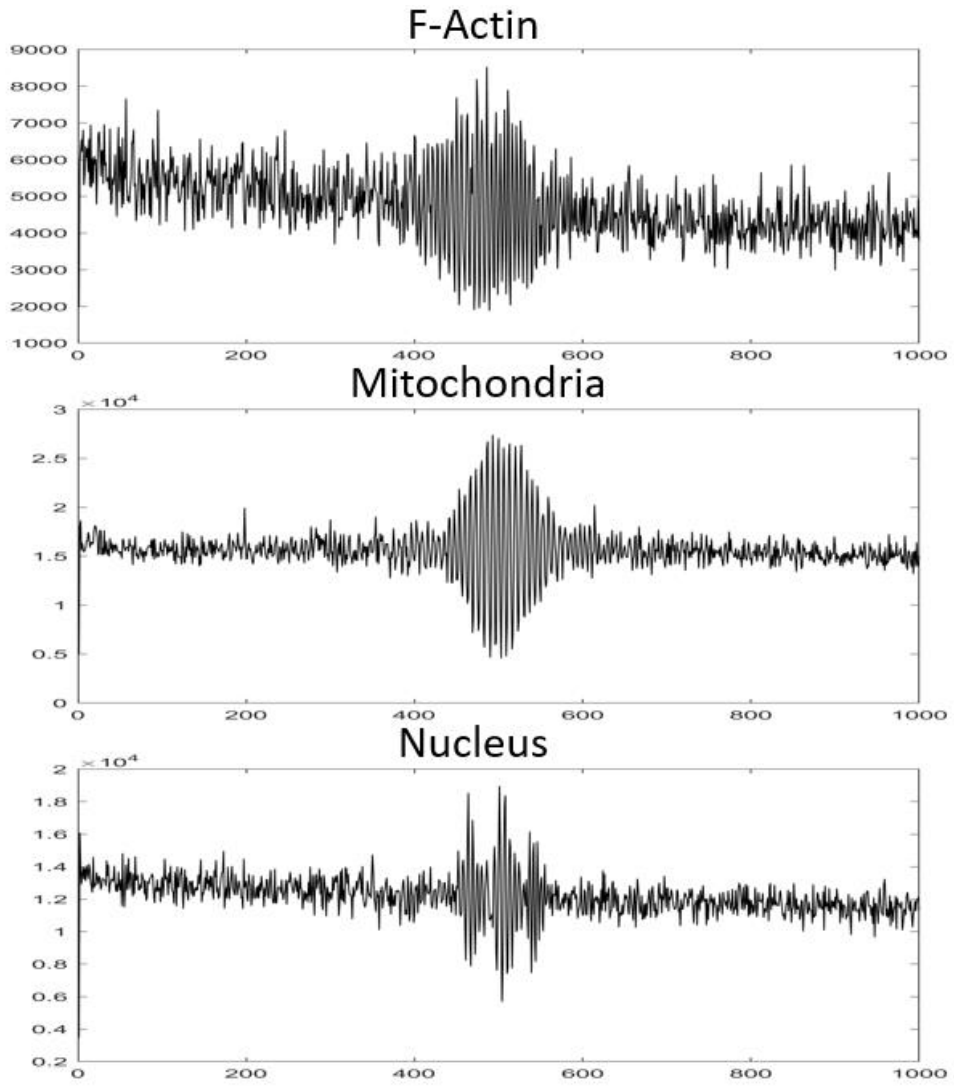

Figure 46: Raw Interferogram Data of Each Type of Organelle

Noteworthy, each pixel may contain the signals from multiple fluorophores. For accurate classification of such pixels, we train the neural network using the mixed data set synthesized from the three single-

channel data. In particular, each single-channel spectrum is given a weight, which is sampled from a uniform distribution where the organelle weights sum to 1. Some generated weights are shown in Figure 47 by the x, y, and z coordinates of the points. These mixed interferograms are used as the training data, and their respective weights are used as the training data answers. The neural net trains on this data and optimizes its weights until an acceptable amount of loss remains. Then raw interferograms with all fluorescent signals are inputted into the neural network which predicts the unknown answers. These details as well as more specific deep learning techniques used are described later in this chapter.

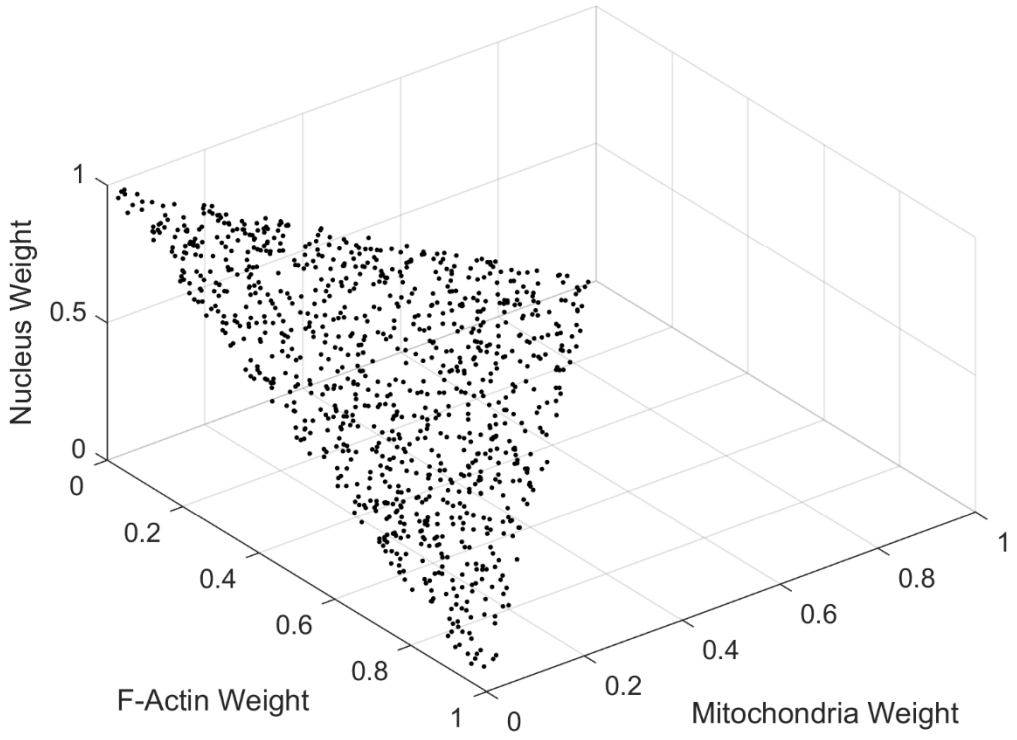

Figure 47: 1000 Weight Vectors Sampled from a Uniform Random Distribution

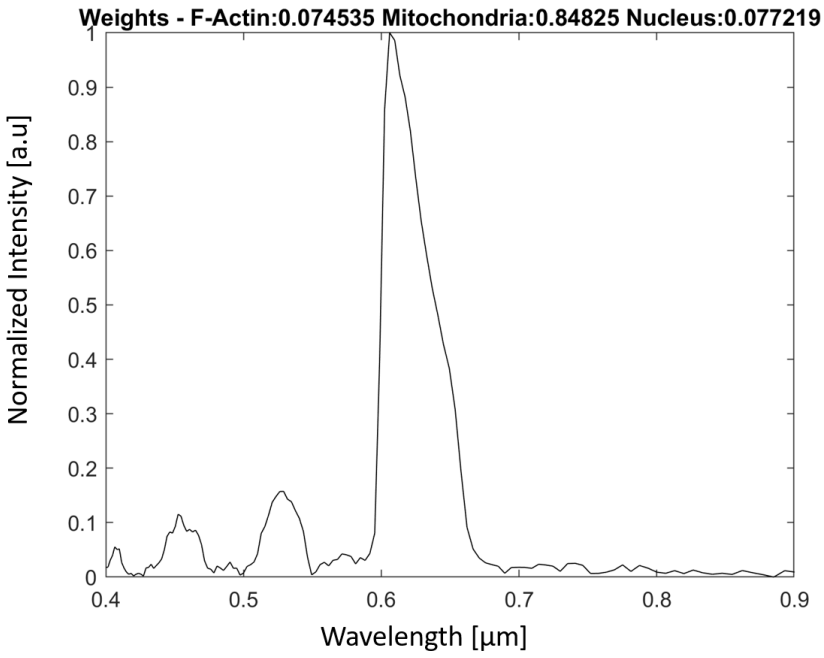

Figure 48: Mixed Spectrum

The randomly sampled spectra are multiplied by a set of randomly sampled weights, resulting in a mixed spectrum as shown in Figure 48. The known weight information is saved and is used later for training the neural network. From each mixed spectrum, the mixed interferogram is computed by the inverse Fourier transform. An example of a computed mixed interferogram is shown in Figure 49.

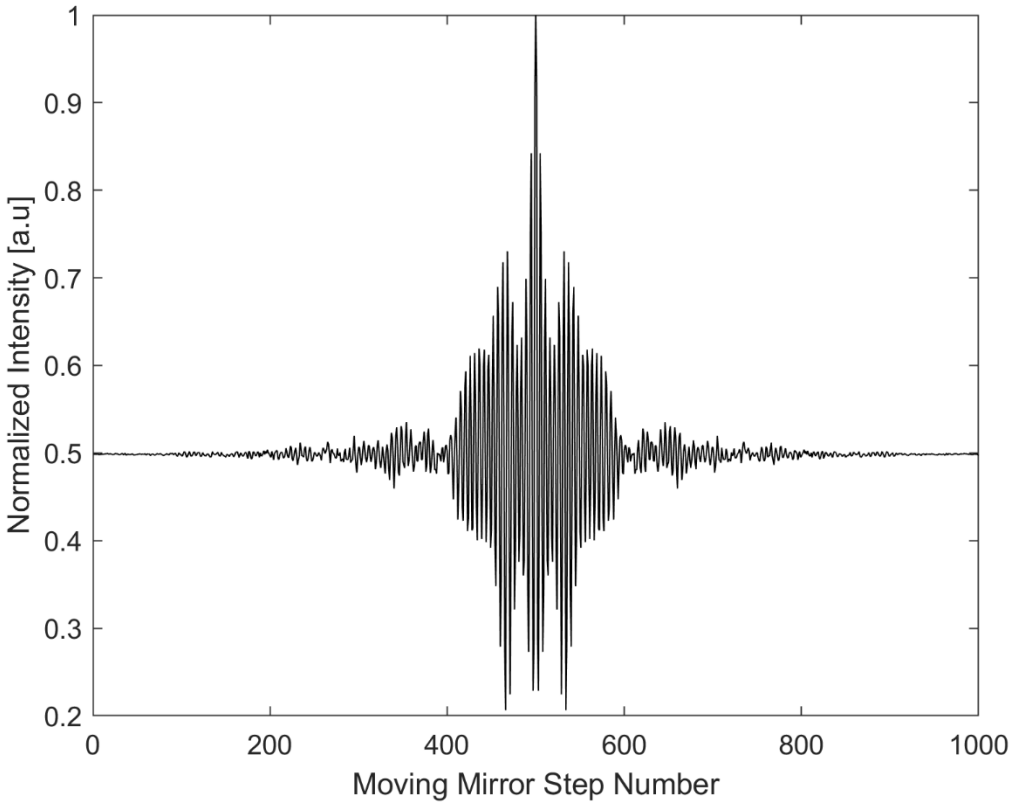

Figure 49: Interferogram of Mixed Sample

In order to create training data with labels from mixed single channel interferograms, the spectra were computed out of necessity requiring the data to be corrected by the ENC algorithm. As a result, the neural network trained with the ENC-corrected data cannot accurately classify the real data containing the experimental noise. To have a chance at successful test data classification results, the experimental noise needs to be re-added to the mixed interferograms in the training data set. To correctly include the experimental noise, we have measured translation stage error (TSE) using the ENC algorithm described in Chapter 2. Observing the experimental translation stage error in Figure 50, it appears to consist of two normal distributions. These may be from two sources of error, potentially from asynchronous mirror vibration and translation stage movement inaccuracy. The TSE to be added to the training data set was similarly generated from a bimodal distribution. Distribution 1 has a mean of 1.25 with a

standard deviation of 0.3 and a weight of 0.72, while distribution 2 has a mean of 0.2, a standard deviation of 0.4 and a weight of 0.28.

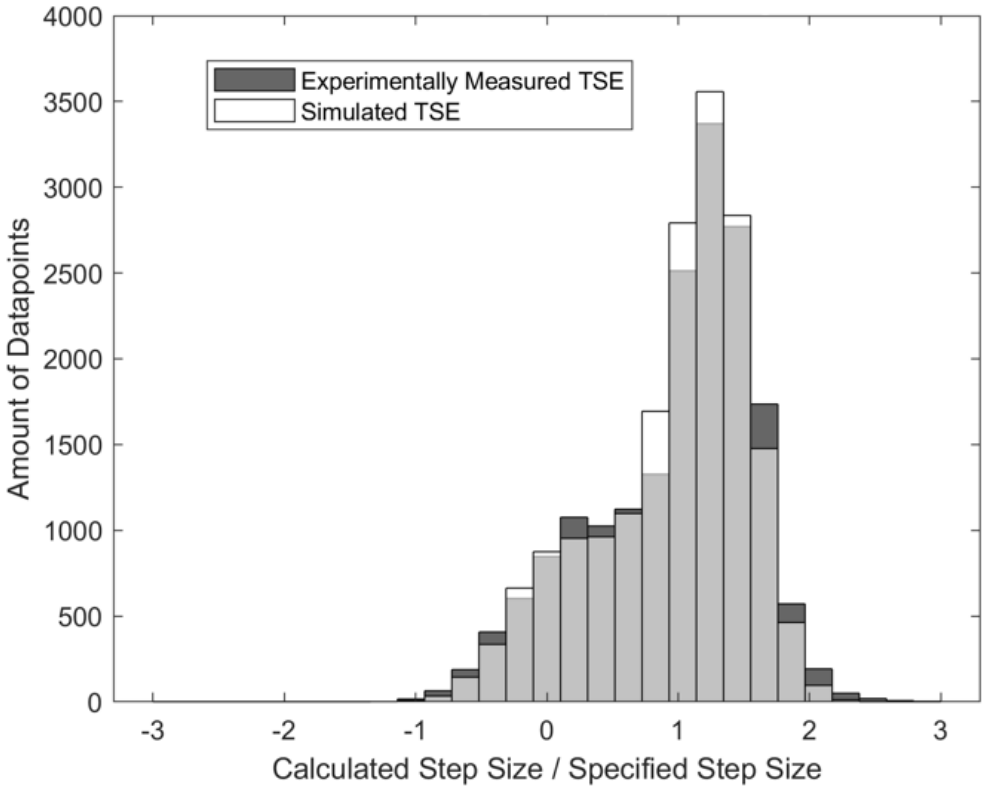

Figure 50: Histogram Comparison between Experimental Noise and Simulated Experimental Noise

We repeat the described weighting process several times to generate a dataset of over 2 million pieces of data. The generated data is split up into training and validation data where the training data is used to train the neural net, but the validation data is not. The validation data comes from the same generated data set, but it is separated and only used to check the performance of the neural net. The testing data set comes from the BPAE sample which is excited by all three LEDs. Since this data has to be manually labeled, one organelle region from each sample is manually selected, and manual labels are given, to check performance while the neural net is training. The dataset size used in experiments is shown in Table 2.

Table 2: Dataset Information

	Amount of Data
Training	2,000,000
Validation	25,000
Testing	9

4.3. Results

The neural network trains with the mixed interferograms and known weights from the training dataset. Several iterations through all of the datasets (epochs) are completed. Eventually the performance stops increasing, indicating that the neural net is fully trained. Once this has been done, the neural net is ready to be used on the data with unknown mixing weights.

4.3.1. Corrected Interferogram Neural Network Results

In Figure 51, we see the neural net training results from the traditional FTS experimental setup. The neural net is outputting values of organelle overlap per pixel from the corrected interferograms. We see that the final classification accuracy of the test data is 100% correct, and the loss value of the test data, which is the mean squared error between the predicted values and the manual labels, is very low. These two metrics indicate that the neural net successfully trained on the interferogram data.

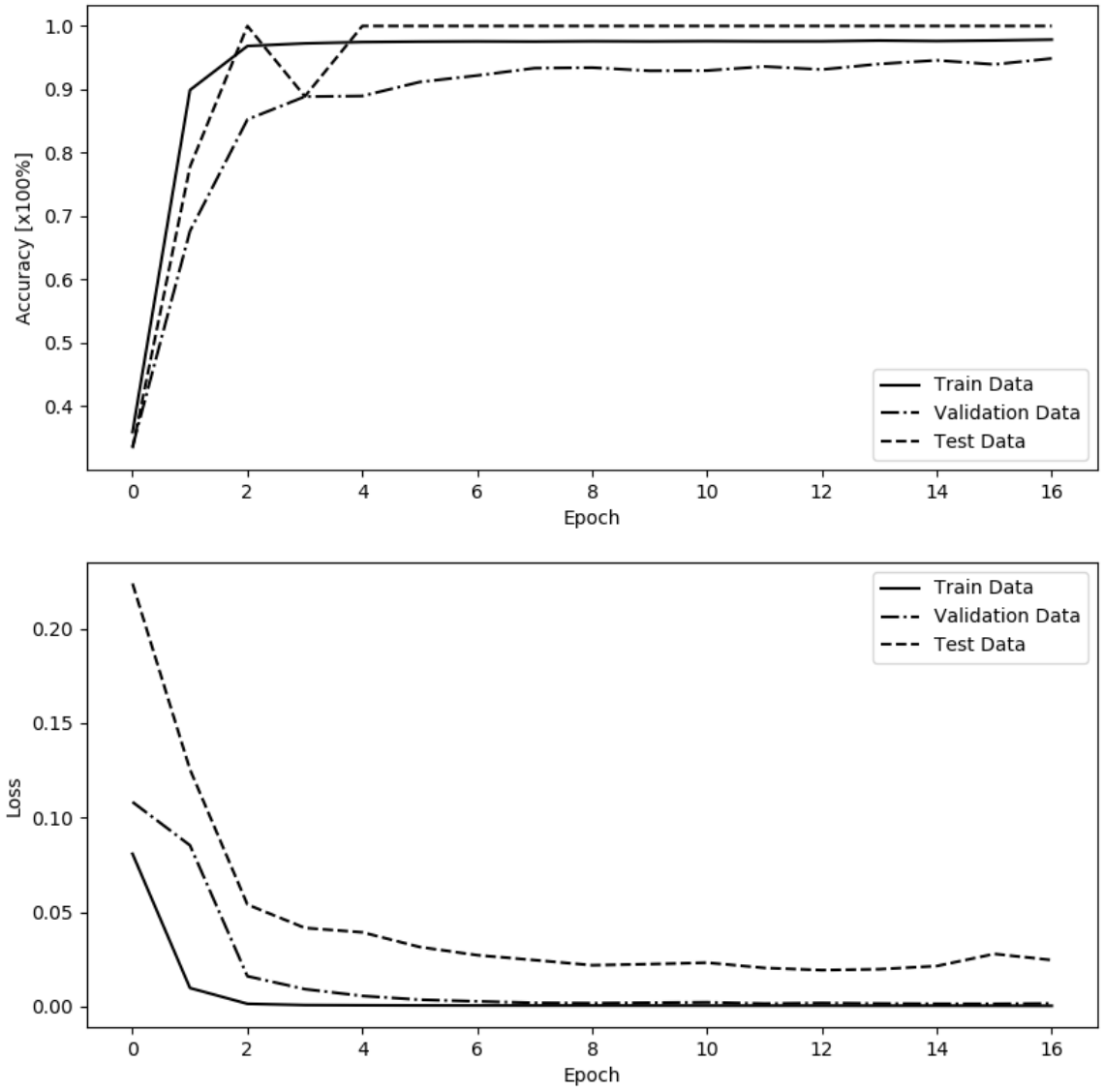

Figure 51: Neural Network Training Results from Experimental Data with Traditional FTS Components

As mentioned earlier, the test set comes from a sample which is excited by all of the LEDs, and is emitting signal from all of the fluorescent dyes simultaneously. Because of this, the exact value of overlapping is unknown, however three regions which are believed to have minimal overlapping are manually selected and labeled. By observing Table 3, we see that predictions by the trained neural network accurately match the manual labels. This confirms that the neural network has correctly found

patterns from the training interferograms which can be directly applied to the raw interferograms of a sample emitting several florescent signals.

Table 3: Results from Trained Neural Network on Experimental Test Data

	Test Sample 1			
	Organelle	F-Actin	Mitochondria	Nucleus
Manually Labeled Region 1	Prediction	0.99496	0.00084	0.00126
	Manual Label	1	0	0
Manually Labeled Region 2	Prediction	0.02262	0.97122	0.00768
	Manual Label	0	1	0
Manually Labeled Region 3	Prediction	0.00748	0.11526	0.88653
	Manual Label	0	0	1

4.3.2. Raw Interferogram Neural Network Results

To demonstrate the feasibility of eliminating the components traditionally used in FTS, we repeated the same process for the dataset without our experimental noise correction algorithm. We see that the neural net reaches 100% classification accuracy, and a mean squared error loss value of 0.11 on the test data. Figure 52 indicates that we can correctly classify multiple fluorophores while eliminating the components required for noise correction.

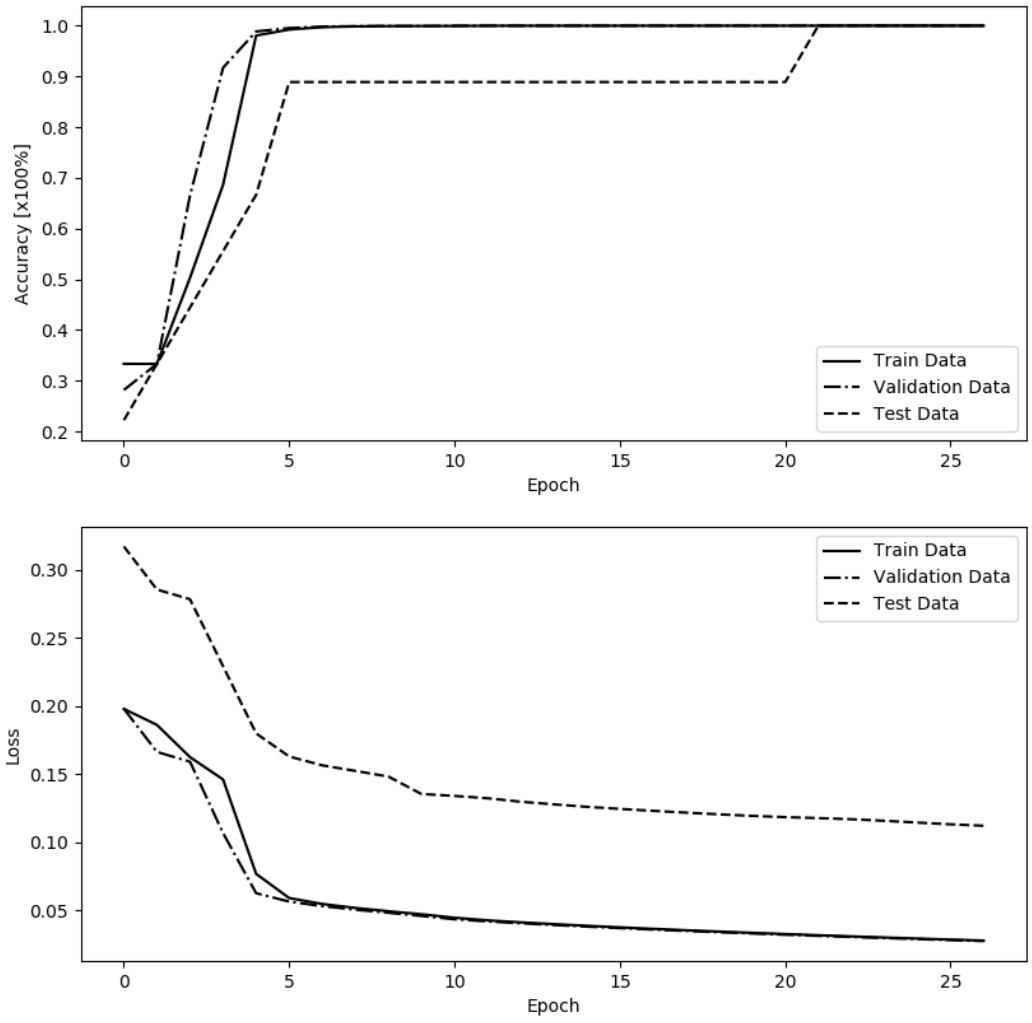

Figure 52: Neural Network Training Results from Experimental Data with Eliminated FTS Components

In Table 4 we see the table of results for the first test sample with the eliminated FTS components. We see that the highest predicted organelle matches the manual label for each organelle and fluorescent dye type.

Table 4: Results from Trained Neural Network on Experimental Test Data with Eliminated FTS Components

	Test Sample 1			
	Organelle	F-Actin	Mitochondria	Nucleus
Manually Labeled Region 1	Prediction	0.56103	0.12287	0.16463
	Manual Label	1	0	0
Manually Labeled Region 2	Prediction	0.17274	0.71203	0.09694
	Manual Label	0	1	0
Manually Labeled Region 3	Prediction	0.18078	0.15592	0.58112
	Manual Label	0	0	1

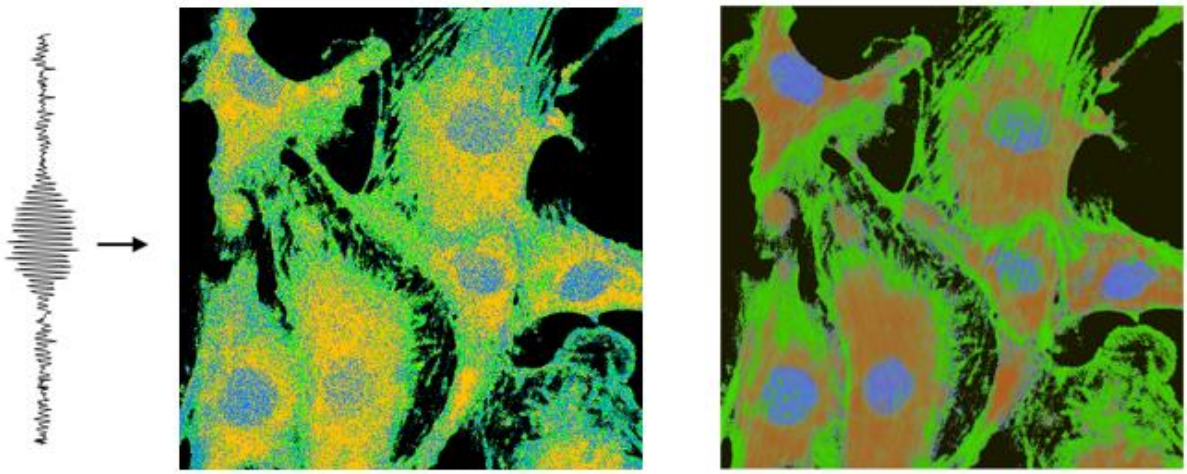

Figure 53: Deep Learning Synthesized Fluorescent Image Compared to Fluorescent Image of Computed Spectrum

On the left of Figure 53, the trained deep learning model is used to synthesize a fluorescent image from the raw interferogram data. The spectrum was not computed with the Fourier transform for each pixel, these are only the outputs from the neural net, helping to visualize the neural net results. Specifically, an RGB value of each fluorescent signal was found, attempting to match the emission spectra of the fluorescent dyes. The neural net classified each pixel, and the RGB values were assigned. On the right of Figure 53, the spectrum was computed at each pixel to compare to the image synthesized by the deep learning model. RGB values of each wavelength were weighted by the computed spectra value at each

wavelength. We see that the output from the deep learning model is very close to the computed spectrum although some additional steps are being taken to improve the results.

Chapter 5 – Summary and Future Works

In summary, the method presented in this thesis combines FTS and deep learning to improve multi fluorescent imaging by removing physical optical components, reducing cost, size, and computational time, as well as increasing the number of fluorescent probes which can be used in a single sample. This method could compete with, and surpass the performance of the standard FISH method for multi-fluorescence imaging.

In the near future, a few improvements are planned with the DL-FTIS BPAE cell procedure. When this is complete, the procedure will be tested with 20+ fluorescent dyes in the same sample. Next the DL-FTIS system will be integrated with some of the other optical systems in the lab, including the SPOT system which stands for Snapshot Projection Optical Tomography. The SPOT system allows to collect a 3D tomographic image of a sample with unprecedented speed (Sung, 2020). By adding the FTS-deep-learning system, we would have a 4D image allowing us to see the spectral fingerprint of the sample at each pixel in 3D space.

The combined system would be able to be used for a vast amount of biological applications. For example, fluorescence in situ hybridization (FISH) uses roughly 20 fluorescent dyes for karyotyping and detecting diseases. Although FISH has been extensively used and improved since it was introduced (DeLong, Wickham, & Pace, 1989), it does have several disadvantages. Using FTS, the data acquisition is slow. Also, it requires experienced personnel to interpret the results (Frickmann, et al., 2017), which is also prone to human errors. We envision that our deep-learning-assisted method can increase the imaging throughput and bypass the human interpretation. This research can be used to open new doors in biological experimentation by allowing more simultaneous fluorescent dyes to be classified in a single sample.

References

Aggarwal, C. C. (2018). *Neural Networks and Deep Learning.* Cham: Springer.

Bagchi, S., & Mitra, S. K. (1999). *The Nonuniform Discrete Fourier Transform and its Applications in Signal Processing.* New York: Springer.

Chollet, F. (2018). *Deep Learning with Python.* Shelter Island: Manning.

Christiansen, E. M., Yang, S. J., Ando, M. D., Javaherian, A., Skibinski, G., Lipnick, S., . . . Finkbeiner, S. (2018). In Silico Labeling: Predicting Fluorescent Labels in Unlabeled Images. *Cell, 173*(3), 792-803.

DeLong, E. F., Wickham, G. S., & Pace, N. R. (1989). Phylogenic stains: ribosomal RNA-based probes for the identification of single cells. *Science, 243*(4896), 1360–1363.

Frickmann, H., Zautner, A. E., Moter, A., Kikhney, J., Hagen, R. M., Stender, H., & Poppert, S. (2017). Fluorescence in situ hybridization (FISH) in the microbiological diagnostic routine laboratory: a review. *Critical Reviews in Microbiology, 43*(3), 263-293.

Goodfellow, I., Bengio, Y., & Courville, A. (2017). *Deep Learning.* Cambridge: MIT Press.

Goodman, J. W. (2015). *Statistical Optics.* Hoboken: Wiley.

Griffiths, P. R., & de Haseth, J. A. (2007). *Fourier Transform Infrared Spectrometry.* Hoboken: Wiley-Interscience.

Haugholt, K. H., Lacolle, M., Bakke, K. A., Tschudi, J., Honne, A., & Storstrom, O. (2011). Low Cost "Laserless" FTIR Spectrometer with Resolution Better Than 0.5cm-1. *Imaging and Applied Optics.* OSA Technical Digest (CD) (Optical Society of America, 2011).

Haugholt, K. H., Lacolle, M., O'Farrell, M., Honne, A., Bakke, K. A., & Lundon, A. (2013). Low cost "laserless" FTIR spectrometer on the farm for real-time nitrous oxide soil emission measurements. *Applied Optics, 52*(4), B93-B101.

Ishikawa, M., & Biju, V. (2011). Luminescent Quantum Dots, Making Invisibles Visible in Bioimaging. *Progress in Molecular Biology and Translational Science, 104*, 53-99.

Kubitscheck, U. (2017). *Fluorescense Microscopy From Principles to Biological Applications.* Weinheim: Wiley-VCH.

Lanoue, E., Genest, J., & Gibeault, M. (2006). Effects of Reference Laser Intensity Noise in a Fouierier Transform Spectrometer. *IEEE Transactions on Instrumentation and Measurement, 55*(3), 860-868.

LeCun, Y., Cortes, C., & Burges, C. J. (2010). *The MNIST handwritten digit database*. Retrieved 2019, from http://yann.lecun.com/exdb/mnist/

Niehorster, T., Loschberger, A., Gregor, I., Kramer, B., Rahn, H.-J., Patting, M., . . . Sauer, M. (2016). Multi-target spectrally resolved fluorescence lifetime imaging microscopy. *Nature Methods, 13*, 257–262.

Orth, A., Ghosh, R. N., Wilson, E., Doughney, T., Brown, H., Reineck, P., . . . Gibson, B. C. (2018). Super-multiplexed fluorescence microscopy via photostability contrast. *Biomedical Optics Express, 9*(7), 2943-2954.

Ounkomol, C., Seshamani, S., Maleckar, M. M., Collman, F., & Johnson, G. R. (2018). Label-free prediction of three-dimensional fluorescence images from transmitted-light microscopy. *Nature Methods, 15*, pages917–920.

Sung, Y. (2020). Snapshot projection optical tomography. *Phys. Rev. Applied, 13*(5), 054048.

Wadduwage, D. N., Singh, V. R., Choi, H., Yaqoob, Z., Heemskerk, H., Matsudaira, P., & So, P. T. (2017). Near-common-path interferometer for imaging Fourier-transform spectroscopy in wide-field microscopy. *Optica, 4*(5), 546-556.

Yang, C., Hou, V., Nelson, L. Y., & Seibel, E. J. (2013). Mitigating fluorescence spectral overlap in wide-field endoscopic imaging. *Journal of Biomedical Optics, 18*(8), 86012.