# Automatic Interferometric Alignment of a Free-Space Optical Coherence Tomography System

by

Andrew Cenko

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2011

# Abstract

Optical Coherence Tomography (OCT) is a relatively new interferometric technology that allows for high-resolution and non-destructive tomographic imaging. One of its primary current uses is for *in vivo* and *ex vivo* examination of medical samples. It is used for non-destructive examination of ocular disease, dermatological examination, blood vessel imaging, and many other applications. Some primary advantages of OCT imaging include rapid imaging of biological tissue with minimal sample preparation, 3D high-resolution imaging with depth penetrations of several millimeters, and the capability to obtain results in real time, allowing for fast and minimally invasive identification of many diseases.

Current commercial OCT systems rely heavily on optical fiber-based designs. They depend on the robustness of the fiber to maintain system performance in variable environmental conditions but sacrifice the performance and flexibility of free-space optical designs. We discuss the design and implementation of a free-space OCT interferometer that can automatically maintain its alignment, allowing for the use of a free-space optical design outside of tightly controlled laboratory environments.

In addition, we describe how similar enhancements can be made to other optical interferometric systems. By extending these techniques, we can provide similar improvements to many related fields, such as interferometric metrology and Fourier Transform Spectroscopy. Improvements in these technologies can help bring powerful interferometric tools to a wider audience.

# Acknowledgments

It is my pleasure here to thank all the people that made this thesis possible.

I owe my deepest gratitude to Arsen Hajian, who has been a constant supporter and guide for my work and, without whom, I would not even have begun this endeavour.

I would also like the thank the rest of my thesis committee, namely Paul Calamai, Maud Gorbet, Glenn Heppler, Ed Vrscay, and James Fraser. Their comments, questions, and suggestions have greatly enhanced the quality of this thesis and forced me to think in directions I would not otherwise have considered.

It is an honor for me to thank my fellow graduate student and partner in crime, Jeff Meade. Without the many late night lab sessions, the immediately following early morning discussions, and the (often fruitless) food hunting expeditions, this work would have been infeasible and much more boring.

I am also grateful to my other fellow graduate students, Tom Haylock, Farnoud Kazemzadeh, Simon So, and Linda Vu, who both aided me in completing my research and provided an escape when I could no longer bear to look at my own.

I am also indebted to the aid and support provided by my collaborators at Tornado Medical systems, especially Andrew Berkeley, Peter Christensen, Annie Deng, Jan Hendrickse, Stefan Larson, Fred Sweeney, Paul van der Vecht, and Jonathan Vermette.

Additional thanks are due to the machinists at the UW Engineering Machine Shop and Mike Hintermeister, who helped turn system drawings into the components necessary to complete this work.

I also extend my gratitude to the Thunder Bay Regional Research Institute, a research partner of the Thunder Bay Regional Health Sciences Centre, for their financial support of this project. Further support was also provided by Tornado Medical Systems and Arjae Spectral Enterprises.

Finally, and most importantly, I wish to thank my parents, Alex and Martha Cenko, my brother, Lesyk Cenko, and all the rest of my extended family for their continued love and support. Without them, I would never have managed to reach this point.

# Dedication

This thesis is dedicated to my mother, who has shown me the value of perseverance, hard work, and self-esteem. It is also dedicated to my father, who has always pushed me to do more than I believed possible.

# Table of Contents

# List of Tables

# List of Figures

# List of Symbols

| | |
|---|---|
| $\bar{\lambda}$ | The mean wavelength of the source. |
| $\Delta\lambda$ | The bandwidth of the source. |
| $\phi(v)$ | The constant phase offset of a wave as a function of frequency. |
| $a(v)$ | The amplitude of a wave as a function of frequency. |
| $A_c$ | The coherence area |
| $c$ | The speed of light in the medium. |
| $d$ | The diameter of the diffraction limited spot. |
| $d_s$ | The distance from the 0 optical path delay to a scatterer. |
| $f$ | The focal ratio of the imaging system. |
| $I$ | The integrated intensity of light. |
| $L_c$ | The coherence length of the source. |
| $N$ | The number of spectroscopic samples in the detector plane. |
| $n$ | The index of refraction of the medium. |
| $R$ | The distance of the source to the coherence plane. |
| $S$ | The area of the source. |
| $t$ | The time of interest. |
| $V$ | The fringe visibility. |
| $v$ | The frequency of the waveform. |

| | |
|---|---|
| $z$ | The axial resolution. |
| $z(t)$ | The complex analytic signal associated with the real waveform at a given time. |
| $\mathcal{D}$ | The depth of focus. |
| $\mathcal{W}$ | The imaging window size. |
| InGaAs | Indium Gallium Arsenide. Often used for infrared light detectors in the 1-3 $\mu$m range. |
| ND | Neutral density. A type of filter used to reduce the intensity of a light signal uniformly at all wavelengths. Also can refer to the level of intensity reduction on a logarithmic scale (base 10). |
| NIR | Near Infrared. A range of light with a slightly longer wavelength than visible light. For optical components, this range often spans from about 650 nm to about 1000 nm [40]. |

# Chapter 1

# Introduction

Optical Coherence Tomography (OCT) is a quickly advancing interferometric medical imaging technology. It fills an imaging niche between low resolution, high penetration depth imaging techniques like Ultrasound (US) and Magnetic Resonance Imaging (MRI) modalities and high resolution, low penetration techniques like Confocal Microscopy (CM) techniques. OCT provides high resolution imaging ($\sim$1 $\mu$m) over 3D volumes spanning several millimeters with minimal sample preparation time. It is often used to study ocular, vascular, respiratory, dental, dermal, neurological, and gastrointestinal diseases [5, 9, 10, 13, 49].

Currently, there is sometimes a significant hurdle between state-of-the-art research systems and commercial implementations of OCT systems—many high end research systems use a free-space optical design [4, 25, 37, 46] and would need to be reimplemented using fiber optic components for a commercial product. This reconfiguration greatly improves the robustness of the system to external effects but comes at the cost of additional development time and can often reduce the overall system performance. In this document, we discuss the enhancement of a free-space OCT system by implementing an automatic alignment system to reduce external effects on the system.

## 1.1 Motivation

Free-space optical designs have some advantages and disadvantages compared with fiber optic based designs. I propose to maintain the advantages of a free-space optical design while reducing some of the limitations. This will allow free-space designs to be used in commercializable OCT systems.

### 1.1.1 Free-Space Optics Advantages

Fiber optics are primarily designed for conveniently transporting light long distances. While very useful for OCT systems, the available optical components and operating wavelengths are heavily limited compared to free-space choices. Current OCT systems implement significant portions of their design in free-space optics, such as the sample focusing system, because of these limitations. The fiber optics are primarily used in the interferometer body, where tolerances are most strict. In addition, by enclosing the light transport path inside a fiber, it can be difficult to modify and enhance an already designed system.

In addition to the limitations in component choices, there can be performance penalties for a fiber based design. Simple off-the-shelf fiber cables quote losses of 0.3 dB (∼6.7%) compared with coated off-the-shelf free-space optical losses of less than 1% [40]. In addition, coupling between free-space and fiber based systems, as performed in most current OCT systems, imposes additional losses that can become fairly severe with minimal misalignment [41, 48].

Free-space optics also maintain polarization better than fiber optics. In order to achieve a strong interference signal, maintaining proper polarization is important. While careful design and polarization controlling devices can mitigate some of this effect in fiber systems, a free-space optical design makes polarization control much simpler. This can be especially important in polarization sensitive OCT applications, such as Mueller OCT systems [13].

The chromatic variation in the index of refraction of fiber is different and more significant than that of air, making it important to closely match the length of fiber in each arm of the interferometer to minimize dispersive effects [10]. Because the path lengths in the interferometer already need to be closely matched in an OCT system, the removal of the fiber reduces this additional source of dispersion in the system.

With current trends in OCT technology, higher resolution systems are a large focus of research. This requires extremely broadband light sources to improve the axial resolution of the system [4, 37, 46, 49]. Because the wavelength range that can efficiently propagate through a single mode fiber is constrained by the physical parameters of the fiber, it can be difficult to design a fiber system that provides high throughput with a large bandpass. This can be especially difficult at short wavelengths, where high lateral resolution can also be achieved. By removing the fiber from the system, standard broadband optical coatings can be used to provide high throughput over large wavelength regions.

### 1.1.2 Free-Space Optics Limitations

One of the main advantages of fiber based designs is the ability to contain large optical paths in an easily manipulated fiber. It is relatively simple to encapsulate many meters of path length in a small coil that can be later stretched a long distance and then coiled again. A free-space optical system likely needs to be larger to accommodate the same path length requirements.

In a free-space system, efficiently travelling long distances can be difficult and can greatly magnify small alignment errors—a small tilt of a beam entering a fiber will cause a small light loss at the far end of the fiber while the same error in a free-space system could be magnified to a fairly large beam shear.

The discrete optics in a free-space system are also more sensitive to positional effects. If a lens moves by a small amount, the beam position and tilt can change by relatively large amounts. These effects are both seen in construction and in use and require special care in interferometric systems. Initial alignment of an interferometer built with free-space optics is significantly more difficult than a fiber based design. Temperature changes of a few degrees are sufficient to cause noticeable alignment changes and can occur simply from the body heat of an operator near the system.

In addition, certain OCT applications use a catheter to examine otherwise unreachable observation sites. In such cases, avoiding the use of a fiber is almost impossible—the flexibility of the fiber allows it to guide light along the frequently changing paths required by the catheter. While this does not remove all advantages of a free-space system, it does place limitations on the enhancements that a free-space system can provide. Because of dispersion matching considerations and the complexity increase, it can be preferable to use a full optical fiber based design for these types of applications.

### 1.1.3 Mitigating Limitations

With all the potential improvements of a free-space optical design, it is highly desirable to use in commercial settings. While not all applications are appropriate to a free-space design, significant improvements can be achieved in many applications. To use such designs in a commercial setting, we must overcome the primary limitation—maintaining alignment.

For an OCT system to be useful in a commercial setting, it needs to be available reliably with high performance. Frequently stopping to realign the system or requiring special technical expertise can severely disrupt productivity.

I will describe a free-space OCT system that actively maintains its alignment without disrupting the user's workflow. While my primary focus is on an OCT system, similar techniques are highly applicable to other interferometric technologies. This will expand the number of interferometric technologies that can be commercially viable outside of highly controlled laboratory environments and also improve some of the technologies already available.

## 1.2 State-of-the-Art

Optical alignment is a necessity of optical systems. Optics vendors sell alignment aids ranging from kinematic optic mounts up through auto-alignment packages[15, 40]. The more complex devices are designed for general R&D usage, though, and the simpler devices require specialized designs to maintain the alignment of a complex OCT system in a commercial setting. By leveraging products such as these we are able to develop and implement a working auto-aligning OCT system with minimal custom part development, enabling the work presented in this thesis.

Other teams have implemented auto-alignment systems for free-space interferometric technologies, such as in gravitational wave detection systems like LISA and VIRGO [1, 21]. These are expensive custom configurations designed for the extremely precise scientific requirements of the specific projects. They assume an experienced scientist will be involved with the system in addition to working with a heavily isolated instrument that is unlikely to become commercially viable. They also generally operate on relatively narrowband light (single frequency laser) rather than the broader bandwidths required for high-resolution OCT imaging—this narrow bandwidth allows easy interference phase measurement and enables different measurement techniques to be used.

In OCT systems, the accepted approach involves extensive use of fiber optics. As discussed above, this approach has many advantages and disadvantages. In this thesis, we describe a system that reduces some of the most serious disadvantages and enables system designs that stretch the current limits of commercial OCT designs.

## 1.3 Thesis Structure

The following chapters will expand on our discussion. Chapter 2 provides basic background information to assist in understanding the topic of discussion. Chapter 3 will introduce the base OCT system we designed and built, to be enhanced with an automatic alignment system. Chapter 4 describes the automatic alignment system, discussing the optical design, the component choices, the monitoring and control techniques, and shows results of the system in action. Chapter 5 explains the design method in more depth, showing how similar enhancements can be made for other optical interferometric systems. Finally, chapter 6 ties everything together and provides a summary of the accomplishments and future goals of this project.

# Chapter 2

# Background

I will now cover background material essential to the understanding of this thesis project. I will begin by reviewing the basics of interferometry, which are central to the technology under study. This will lead into a discussion of optical coherence tomography (OCT), a type of interferometry that will be used as a case study in this project.

## 2.1 Interferometry

At the most basic level, interferometry is the study of the interference patterns of waves. These waves can take any number of forms including water waves, sound waves, light waves, and many others. When multiple similar waveforms overlap, interference patterns emerge. These interference patterns can provide extremely precise information about the underlying waveforms. By studying these interference patterns, it is possible to extract information that would be much more difficult to measure through conventional means.

### 2.1.1 What is a Wave?

Simply described, a wave is a cyclic energy pattern propagating through a medium. For the purposes of this discussion we will mostly be interested in one dimensional steady-state electromagnetic waves, which can be described by [30]

$$z(t) = \int_0^\infty a(v) e^{i(\phi(v) - 2\pi v t)} dv, \tag{2.1}$$

where $z(t)$ is the complex analytic signal associated with the real waveform at a given time, $v$ is the frequency of the wave, $a(v)$ is the amplitude of the wave as a function of frequency (the amplitude spectrum of the wave), $\phi(v)$ is the constant phase offset of the wave as a function of frequency, and $t$ is the time of interest. While only the real portion of this waveform is usually measured, this formulation provides analytical advantages which are utilized below. The important thing to note is that the waveform has some amplitude and phase as a function of frequency and that these describe an electromagnetic wave that oscillates as a function of time. The amplitude as a function of frequency is known as the spectrum.

### 2.1.2 How Do Waves Interfere?

When two waveforms overlap, they add together and interference becomes possible [30].

$$\begin{aligned} z_1(t) + z_2(t) &= \int_0^\infty a_1(v)e^{i(\phi_1(v)-2\pi vt)} + a_2(v)e^{i(\phi_2(v)-2\pi vt)}dv, \\ &= \int_0^\infty (a_1(v)e^{i\phi_1(v)} + a_2(v)e^{i\phi_2(v)})e^{-i2\pi vt}dv. \end{aligned} \tag{2.2}$$

Unless there is overlap in the spectra ($a_1$ and $a_2$ are both non-zero for the same $v$), this equation reduces to a simple sum. If there is overlap in the spectra, the amplitude and phase determine the interference pattern. To see interference, we need to have waves of the same frequency in our system.

With current detector technology, we cannot directly measure visible light waveforms and must detect the intensity. The actual waveforms have frequencies of hundreds of terrahertz and the detection requirements are faster than currently available technology. By integrating the light on a detector, we can measure its intensity. The intensity can be easily described from the complex analytical waveform representation by

$$I(T) = \int_0^T [z(t)]^*[z(t)], \tag{2.3}$$

where $^*$ denotes the complex conjugate [30] and $T$ is the integration length. Note that $T$ will be much longer than the cycle period of the light, averaging over many cycles.

If we wish to measure the inteference above, we can calculate the intensity as

$$\begin{aligned}
z_1(t,v)+z_2(t,v) &= (a_1(v)e^{i\phi_1(v)}+a_2(v)e^{i\phi_2(v)})e^{-i2\pi vt}, \\
I_{z_1+z_2}(T) &= \int_0^T \left(\int_0^\infty z_1(t,v)+z_2(t,v)dv\right)^* \left(\int_0^\infty z_1(t,v)+z_2(t,v)dv\right) dt, \\
& \text{assuming the orthagonality of sines and cosines applies} \\
& \text{for our relatively long integrations, this simplifies to:} \\
&= \int_0^T \int_0^\infty (a_1(v)^2+a_2(v)^2+a_1(v)a_2(v)e^{i(\phi_2(v)-\phi_1(v))}+ \\
&\qquad a_1(v)a_2(v)e^{i(\phi_1(v)-\phi_2(v))})e^{-i2\pi vt}e^{i2\pi vt}dvdt \\
&= T\int_0^\infty a_1(v)^2+a_2(v)^2+a_1(v)a_2(v)e^{i(\phi_2(v)-\phi_1(v))}+ \\
&\qquad a_1(v)a_2(v)e^{i(\phi_1(v)-\phi_2(v))}dv \\
&= T\int_0^\infty a_1(v)^2+a_2(v)^2+2a_1(v)a_2(v)\cos(\phi_2(v)-\phi_1(v))dv. \qquad (2.4)
\end{aligned}$$

Note that we will ignore the scaling factor T (integration length) below.

In an OCT system, the return beam from each scatterer in the sample is phase shifted by the depth of the scatterer. This allows us to relate the phase of the interferring beams as

$$\phi_2(v) = \phi_1(v) + 2\pi v\frac{d_s}{c}, \qquad (2.5)$$

where $\frac{d_s}{c}$ is the time delay introduced by the position $d_s$ of the scatterer with the light travelling through the medium at a speed of $c$, ignoring dispersion. Note that the distance travelled by the light is twice the physical distance to the scatterer, since the light must travel there and back before interferring. This allows us to reduce our interference equation to

$$I_{\text{OCT}}(d_s) = \int_0^\infty \left[a_1(v)^2+a_2(v)^2+2a_1(v)a_2(v)\cos\left(2\pi v\frac{d_s}{c}\right)\right]dv. \qquad (2.6)$$

The first two terms in this equation reduce to the integrated intensity of the two beams while the third term is the cross-correlation function of the two beams as a function of the scatterer position.

Because the two beams are generated from the same source, this is the auto-correlation function of our source. The Wiener-Khintchine theorem states that the auto-correlation function of a complex electric field is the Fourier transform of the power spectrum [30]. This tells us that the signal visibility in OCT is directly related to the Fourier transform of our spectrum. Because of this, we know that a narrowband (high temporal coherence) source will have higher signal visibility than a broadband (low temporal coherence) source.

As an example, consider a source with a Gaussian spectral shape. It is well known that the Fourier transform of a Gaussian is itself a Gaussian with a width inversely proportional to the width of the original. If the width of the Gaussian spectrum of our source increases, the width of the Fourier transform will decrease and vice-versa.

Through the Wiener-Khintchine theorem, we know that the autocorrelation function of our source will follow the Fourier transform of our spectrum. This tells us that, for a Gaussian spectral source, the oscillation signal in equation 2.6 will show a delay based decay dependent on the bandwidth of our spectrum. As the spectrum increases in bandwidth, the interference signal will decay more quickly. By narrowing our spectrum, we can lengthen the range over which a meaningful interferometric signal can be obtained. While we specifically use a Gaussian spectrum in this example, other spectral shapes follow a similar pattern. If we wish to see a strong interferometric signal at long delays, we will want to use a narrowband source. If we instead wish to isolate the interferometric signal to a small region, we will want to use a much wider spectral source. We can also split a broadband detection into multiple narrowband detections, maintaining the broadband information of our source while allowing us to maintain high throughput.

### 2.1.3 What is an Interferometer?

An interferometer is a device that uses interference patterns to measure properties of waveforms. One of the simplest interferometers can be seen in Young's double slit experiment. A spatially coherent light source (often narrowband, such as a laser, to increase the coherence length) is projected through two parallel slits and focused onto a detector or imaging screen. An alternating pattern of light and dark "fringes" are visible on the screen, forming an interference pattern. This interference fringe pattern is specifically determined by the system configuration. Depending on the known system parameters (for example, the distance between the slits, the wavelength of the light, the focal length of the focusing lens, etc), it is possible to accurately determine some of the unknown parameters. The measurement and quantification of these inteference effects is the field of interferometry with the actual instruments known as interferometers.

### 2.1.4 Interferometric Tolerances

Interference is a very sensitive phenomenon. While this sensitivity provides a powerful tool, it also leads to very tight tolerances. Misalignment at fractional wavelength levels will significantly affect the resultant signal. A shift of one wavelength causes a full $2\pi$ phase rotation in the interference signal. In optical interferometers, the wavelengths are on the order of a micron. This can place tolerances well below the micron level.

### Fringe Visibility

The fringe visibility $V$ is a relative measure of the interference fringing power (i.e., the light fluctuations due to the interference effect) to the total power in the signal

$$V = \frac{I_{\max} - I_{\min}}{I_{\max} + I_{\min}}. \tag{2.7}$$

The maximum fringe visibility is 1.0 with the minimum being 0. Reduced fringe visibility can be due to actual signal properties (such as the coherence length) or to misalignment. Because the fringing signal contains all the interferometric information, maximizing fringe visibility is important for good signal measurement.

### Coherence Length (Temporal Coherence)

Many interferometric experiments are performed on broadband signals and, even when monochromatic signals are desired, perfectly monochromatic sources are unavailable. While perfectly monochromatic light will have a strong fringe signal at any path delay due to the perfect sinusoidal nature of the interference pattern (see equation 2.6), real signals will have a finite coherence length past which the fringing signal is difficult to discern; this delay range corresponds to a distance called the coherence length. The coherence length of a light source can be expressed as [30]

$$L_c = \frac{\bar{\lambda}^2}{\Delta\lambda}, \tag{2.8}$$

where $\bar{\lambda}$ is the mean wavelength of the source and $\Delta\lambda$ is the bandwidth. As the source increases in bandwidth, the coherence length becomes shorter. It is important to keep path lengths in the arms of an interferometer matching to less than the coherence length of the source. This also places a limit on the information that can be efficiently measured before the fringe visibility is too low.

If the coherence length of the source is too short, spectrally dispersing the signal over multiple channels can increase the coherence length of each channel. This can greatly improve an interferometric signal measurement.

### Coherence Area (Spatial Coherence)

In addition to the temporal constraint on interference, light exhibits greater coherence coming from similar spatial positions. Unlike temporal coherence, this spatial coherence does not depend on the bandwidth of the light but rather on the mean wavelength of the light and the size of the emitting source. The coherence area of a source can be expressed as [30]

$$A_c = \frac{R^2 \bar{\lambda}^2}{S}, \tag{2.9}$$

where $A_c$ is the coherence area, $R$ is the distance of the source to the coherence plane (equivalent to the distance from the source to the collimating lens in a Michelson style interferometer), $\bar{\lambda}$ is the mean wavelength of the source, and $S$ is the area of the source. This can also be expressed as a function of the solid angle subtended by the source $\Omega \approx \frac{S}{R^2}$, a measure of the angular extent of the source.

#### Relative Beam Intensity

Examining equation 2.6, the interference signal can be seen as a modulation of the mean level of the two interfering beams. Additionally considering equation 2.7, maximizing fringe visibility requires maximizing the relative level of interference. Careful examination shows that the maximum fringe visibility occurs when the beam intensities are equal [30]. As additional light will only contribute to noise characteristics in the signal, it is important to match the intensity of our interfering beams.

This tells us that, for optimal interference, the intensity of our beams should be matched. This will give us minimum intensities close to 0 and maximal intensities close to double the mean level, assuming the system is otherwise properly configured. Adding additional intensity to either beam will keep the interference strength the same while increasing the mean level. As we are primarily interested in the interference signal, additional intensity is simply a bias level that adds additional photon noise to our detection.

### 2.1.5 Interferometric Devices

Interferometry is a powerful measurement technique with many applications beyond just OCT. Extremely precise positional and vibrational measurements can be made with optical metrology systems. Fourier Transform Spectroscopy is another interferometric technique allowing for precise determination of the spectrum of an input source. Aperture synthesis allows for simulating larger telescopes (or other imaging apertures) with a collection of smaller telescopes. Many other interferometric technologies exist in many fields, utilizing this powerful technique to make measurements that would otherwise be infeasible.

Because of the inteferometric basis of these technologies, some of the enhancements developed in one domain are applicable in many of the others. Developing an improvement in one application can have much farther reaching effects, potentially even opening new domains not previously feasible. The improvements implemented in this thesis have the potential to influence many other forms of interferometry.

### 2.1.6 Dispersed Interferometry

Dispersed Interferometry is a subset of interferometry that utilizes the increased coherence time of narrowband sources along with array detector technology to greatly enhance the efficiency of classical interferometric systems.

Because the useful interferometric signal is contained in the fringes on the optical signal, the signal from short coherence length (i.e., broadband) sources decays rapidly. Artificially narrowing the spectrum of the input source can increase the coherence length but this comes at the cost of ignored signal.

By instead dispersing the broadband signal with a diffraction grating and collecting multiple narrowband interferograms with an array detector, you can maintain the original source bandpass while also increasing the signal coherence length in any individual pixel. This simultaneous collection of multiple narrowband interferograms is known as dispersed interferometry.

This technology is the basis for such instruments as the Dispersed Fourier Transform Spectrograph (dFTS) as well as Fourier Domain Optical Coherence Tomography (FD-OCT). In both cases, sensitivity improvements of several orders of magnitude are possible over conventional Fourier Transform Spectrographs [19] and Time Domain OCT systems [26].

## 2.2 Optical Coherence Tomography

Optical coherence tomography (OCT) is a form of interferometric imaging often used in medical and industrial fields. OCT allows for non-invasive micron level resolutions over millimeter sized regions (in 3D) with high imaging rates (typically several Hertz). It allows for high resolution *in vivo* structural imaging of the human eye, arteries, and other tissue. OCT fills in a resolution gap in medical imaging modalities between *in vivo* techniques such as MRI and ultrasound and *ex vivo* techniques such as confocal microscopy. Figure 2.1 shows the imaging regimes covered by these technologies [11].

### 2.2.1 OCT Basics

OCT imaging is based on the scattering characteristics of the tissue under observation. Temporally incoherent light (generally infrared light [$\sim$1$\mu$m wavelength] with a wide bandwidth [$\sim$100 nm]) is split into a sample and a reference arm. The reference light is reflected off a mirror while the sample light is projected into the tissue. Light backscatters from the tissue and is interferometrically recombined with the reflected reference light. The

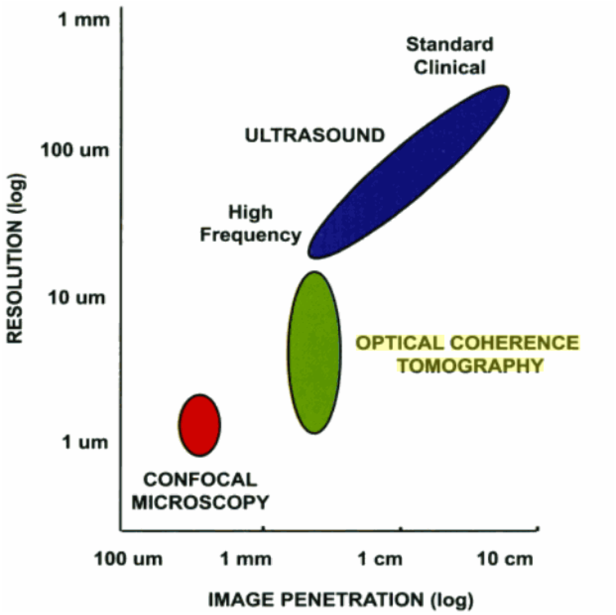


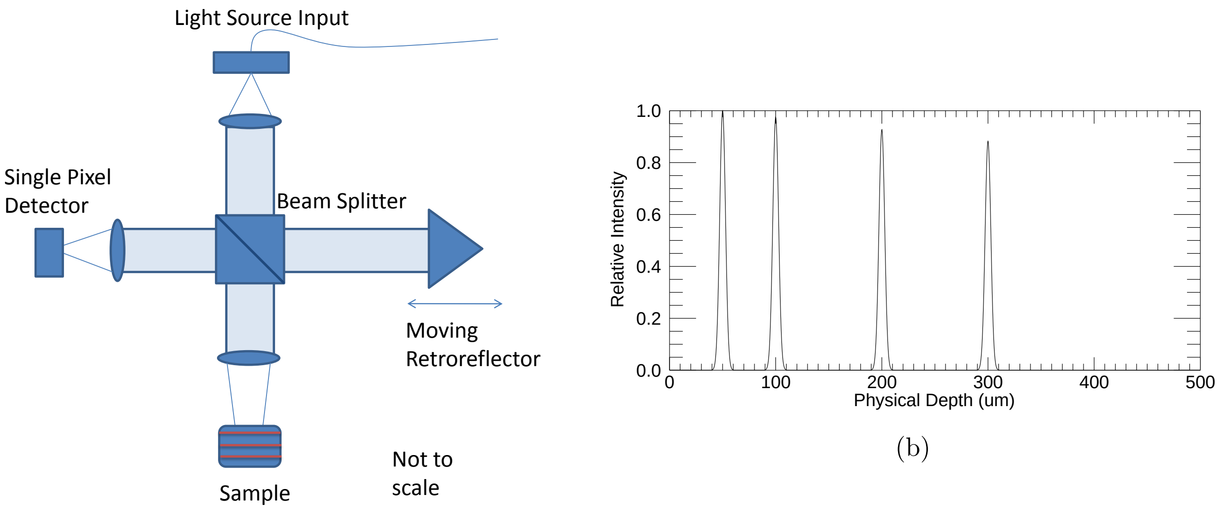
Figure 2.1: Cartoon representation of the depth penetration and resolution of several medical imaging modalities. OCT fills a gap in the resolution ranges. Image taken from [11].

interference pattern detected provides information about the structure of the tissue under examination.

Two main types of OCT systems are currently in use, time-domain OCT (TD-OCT) and Fourier-domain OCT (FD-OCT, also sometimes called spectral-domain OCT or SD-OCT). While the underlying principle is the same, they differ greatly in implementation.

### Time-domain OCT

TD-OCT exploits the low coherence source to scan the depth profile of the sample. The reference arm is scanned over a distance and the interferometric signal is recorded for the broadband source. Highly localized interference patterns show up when the reference arm is at the same distance from the beam splitter as a reflective surface in the sample. By recording the location of the envelopes of these interference patterns, it is possible to reconstruct a reflectivity profile as a function of depth.



(a)

(b)

Figure 2.2: A schematic drawing of a time-domain OCT system and the corresponding data collected.

### Fourier-domain OCT

FD-OCT splits the broadband low coherence interferometric signal into many narrowband high coherence signals. By collecting all the separate narrowband signals and analysing them together, it is possible to extract depth reflectivity profiles for the sample without scanning through delay. The efficiency of a FD-OCT system can be many times greater than that of a TD-OCT system [11, 26].

## 2.2.2 Uses of OCT Systems

OCT systems allow for noninvasive imaging of surface structures. While the penetration depth is relatively short (on the order of millimeters), resolution is relatively high (on the order of microns). This places OCT in an imaging regime between techniques such as MRI and ultrasound (which are low resolution with very high penetration depth) and confocal microscopy (with very high resolution but very limited penetration depth).

These characteristics make OCT ideal for structural examination of the retina and skin. It can be used to detect macular degeneration, tumors, blood flow issues, and many other

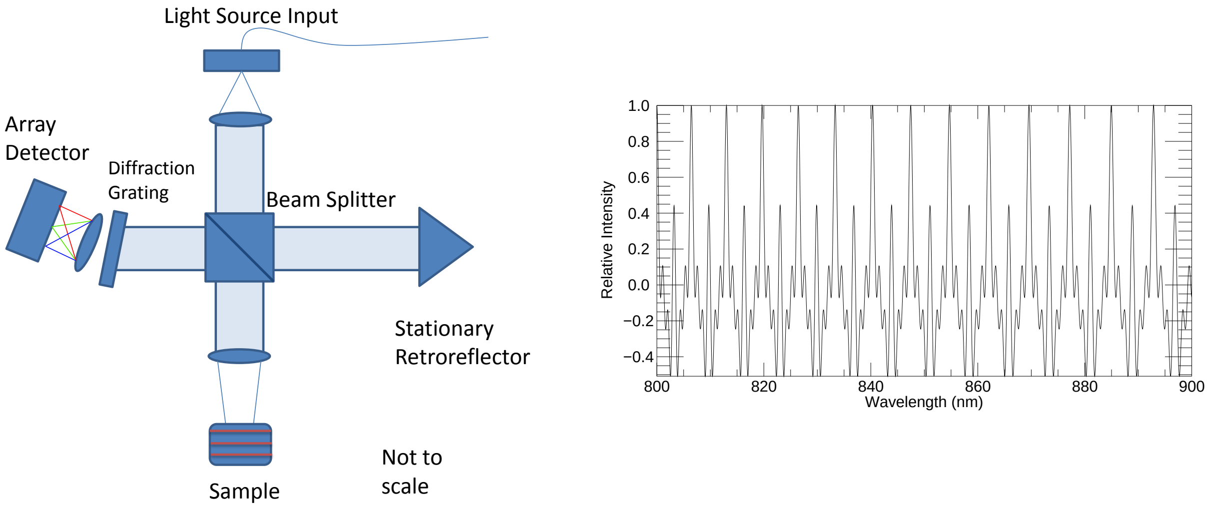


Figure 2.3: A schematic drawing of a fourier-domain OCT system and the corresponding data collected. The position of reflective surfaces is the same as those in figure 2.3(b).

serious problems. OCT can also be combined with endoscopy to view internal structure, such as blood vessel, gastrointestinal, and respiratory imaging [9, 10, 13, 49].

### 2.2.3 Limiting Factors of OCT Performance

OCT system performance can be described by several parameters. The axial and lateral resolutions define the imaging resolution of the system. The scanning range determines the maximal lateral size of the sample, while the scanning speed determines how quickly an image of the entire sample can be generated. The dynamic range of the system determines the minimum reflectivity necessary to detect a signal at different depths. The penetration depth determines the axial size of the image. The depth of focus determines the range over which the sample is in focus. The wavelength of imaging can greatly affect these other parameters.

#### Resolution

Somewhat uniquely, the axial and lateral image resolutions in OCT are decoupled. The axial information in OCT is obtained through the interferometric properties of the signal.

The lateral information is determined through standard imaging. Each axis of resolution has different constraints limiting performance.

Lateral resolution is limited by the optical diffraction limit without resorting to highly specialized optical technologies. The diffraction limited spot size can be approximated by [6]

$$d = \frac{2.44\bar{\lambda} f}{n}, \tag{2.10}$$

where $d$ is the diameter of the diffraction limited spot, $\bar{\lambda}$ is the mean wavelength of the light, $f$ is the focal ratio of the imaging system, and $n$ is the index of refraction of the sample medium. This limit indicates the smallest spot that can be focused in the sample. The smallest discernible feature is about half of the spot size ($\frac{1.22\bar{\lambda} f}{n}$). The main controllable factors influencing the diffraction limited spot size are the wavelength of light and the focal ratio of the imaging system.

Axial resolution is limited by the coherence length of the light source. A shorter coherence length provides higher axial resolution. The coherence length of the light source decreases as the relative bandwidth of the source increases. The axial resolution can be approximated by [11]

$$z = \frac{2 \ln 2 \bar{\lambda}^2}{n \pi \Delta\lambda}, \tag{2.11}$$

where $z$ is the axial resolution, $\bar{\lambda}$ is the mean wavelength, $\Delta\lambda$ is the bandwidth, and $n$ is the index of refraction of the sample medium.

The index of refraction of air is close to 1.0. Water (and biological tissue, which is composed largely of water) has an index of refraction closer to 1.3. Glass has an index of refraction of around 1.5. Passing through materials with a higher index of refraction acts the same as travelling a longer path length. This will effectively increase the resolution of the system.

### Scanning Range and Speed

Each depth reflectivity profile in a standard OCT system contains information from a small line profile of the sample. When a 3D image of the sample is desired, the sample beam needs to be scanned across the sample. The size of the scanning region determines how large a single image can be, generally in the several millimeter square range.

The scanning rate depends on many additional factors. The size of the scanned region and the desired resolution are a dominant factor. The exposure time for a single line scan and the delay between line scans also matter.

### Imaging Window

In addition to scanning across the sample, the design of the system will limit the useful depth imaging window in which data can be collected, $\mathcal{W}$. This is effectively equivalent to the depth scanning range of the system. In TD-OCT, this is determined by the scan range of the reference arm of the interferometer. In FD-OCT, this is determined by the number of spectroscopic samples across our bandpass ($N$, usually determined by the number of pixels on our detector). This can be approximated for FD-OCT by [11]

$$\mathcal{W} = \frac{N\bar{\lambda}^2}{4n\Delta\lambda}. \tag{2.12}$$

### Dynamic Range

The dynamic range of the system is defined as the ratio between the maximal detectable signal and the noise floor limit. It determines the weakest detectable signal from a sample in a relative sense (optimizing the integration time to almost saturate the detector). Because the amount of returning light decays exponentially with depth (due to scattering and absorption), this parameter affects the detectable reflectivity differently at different depths.

### Sensitivity

The sensitivity of the system is defined as the minimum sample reflectivity to obtain a signal-to-noise ratio of one in an image. This is an absolute measurement of the weakest detectable signal of the system, usually described in decibels (dB). This differs from dynamic range by focusing on the absolute limit of the system rather than the relative limit within a sample.

### Penetration Depth

Penetration depth gives an indication of the deepest return that can be reliably discerned. For TD-OCT, the scan range of the reference arm will be a significant limitation on the potential penetration depth. In FD-OCT, the backend spectral resolution provides a similar limitation. In both cases tissue scattering and absorption will also reduce the return signal from deep depths, limiting the useful penetration depth.

Biological tissue generally causes more scattering at shorter wavelengths and more absorption at longer wavelengths. There is an optical window for biological tissue that extends from approximately 700 nm to 1500 nm, where the penetration depth can reach several millimeters [49]. At shorter wavelengths, scattering will begin to limit the useful depth. At longer wavelengths, absorption of light becomes the dominant limiting factor.

### Depth of Focus

The depth of focus determines the axial region over which the lateral resolution doesn't degrade significantly. For a Gaussian beam profile focused into a sample by a lens with a focal ratio of $f$, the depth of focus $\mathcal{D}$ is [11]

$$\mathcal{D} = \pm \frac{1.22^2 \pi \bar{\lambda} f^2}{n}. \tag{2.13}$$

Outside of this region, the lateral resolution of the system will begin to degrade as the lateral slice of the cone angle of light becomes larger. Comparing this to the equation for the diffraction limited spot size (equation 2.10), we can see that a smaller spot size also comes with a smaller depth of focus. This is a tradeoff that needs to be optimized in the system design, providing for high resolution near the focus while not degrading too much at the edges of the image.

### Imaging Wavelength

As seen above, the imaging wavelength can have huge implications on system performance. The wavelength region selected directly affects the axial resolution, lateral resolutions, and the depth of focus. Different tissue scattering and absorption coefficients will also affect the efficiency of light returned from the sample. The optimal optical materials, coatings, sources, and detectors will also change for different wavelength regions. Different health considerations also need to be considered, with potential side effects from certain forms or intensities of radiation.

# Chapter 3

# Base OCT System

Back in October 2009, Jeff Meade and I designed and began implementation of an OCT imaging system. This system serves as the basis for the improvements discussed in chapter 4. Here we will discuss the basic design of our system, the implementation of this system, and present results obtained using this system.

## 3.1 OCT System Design Choices

Before building an OCT system, several important design choices needed to be made. The first design choice involved the goals of the final system. Knowing the goals of the system, we then needed to define the operating wavelength range of the system and the light source we would use to provide this.

Our design goals are relatively simple. First, we want a system that performs as well as (or better than) currently available commercial biological imaging OCT systems. Our system is to provide scanning across tissue samples extracted from a body in a surgical environment, to allow rapid feedback while the patient is still in the operating room. We want the system to be high resolution and capable of fast imaging. In addition, we want the system to be easily upgradeable with new technologies to enhance the overall performance.

Given these design goals, the wavelength range becomes reasonably well defined. We require a large bandwidth to obtain high axial resolution combined with a short center wavelength to allow for high lateral resolution. The goal of imaging biological tissue requires us to work in the near-infrared region to maintain good scattering properties with low light absorption. This optical window for biological tissue extends from approximately 700 nm to 1500 nm [49]. By choosing a shorter wavelength in this range, we can maintain our penetration depth while allowing for higher resolution imaging.

For our lightsource, we want a large bandwidth source with high optical power in the shorter wavelength region of the biological optical window. Preferring a commercially available source, we decided to use the Superlum D855. This superluminescent diode (SLD) source has a mean wavelength of 850 nm with a 100 nm FWHM bandpass and 12 mW output power. This should allow us to achieve an axial resolution of about 4 microns and a similar lateral resolution (determined by the focal ratio of our sample imaging subsystem).

Knowing the desired source, we are able to pick appropriate components for the rest of our system. We know our optical coatings will be chosen to work in the near infrared (NIR) range and that we can use a standard silicon detector rather than needing indium gallium arsenide (InGaAs).

For our detector, we desire high speed combined with high sensitivity and low noise. Because of our broad bandwidth light source, we also require a large number of pixels in the dispersion direction to provide a reasonable depth window (see section 2.2.3). The Basler Sprint spL2048-70km CMOS line scan camera fit our needs nicely while maitaining reasonable cost. It has a 2048 x 2 array of 10 $\mu$m pixels along with a 70 kHz readout rate.

Working back from the detector, we are able to determine the appropriate grating, lenses, and pinhole size for the rest of our system. These components are described in detail below.

The sample scanning configuration was designed by Tom Haylock to provide the high scan speeds necessary while providing a large scan area with the high precision sampling necessary for our high resolution system.

## 3.2 Baseline OCT System Layout

The OCT system is implemented in 3 main sections. The first section is the main interferometer body (figure 3.1), which splits and recombines the light and allows interference to occur. The second section is the sample scanning system (figure 3.2). This system takes the light from the sample arm of the interferometer and scans it across the sample under observation, allowing for a 3D reconstruction of the sample structure. The final section is the backend spectrometer (figure 3.3), which disperses the light from the interferometer and acquires the spectral inteference data.

The light from our SLD enters our system through a single mode fiber matched to the emitting diode. The FC-APC coupler on this fiber is designed to minimize back reflections into the SLD, which can damage the device. This fiber has a numerical aperture (NA) of 0.14 and is collimated by a Thorlabs AC254-75-B 75 mm NIR achromatic lens to provide a 21 mm collimated beam diameter. The collimated light is then sent into the main interferometer body.

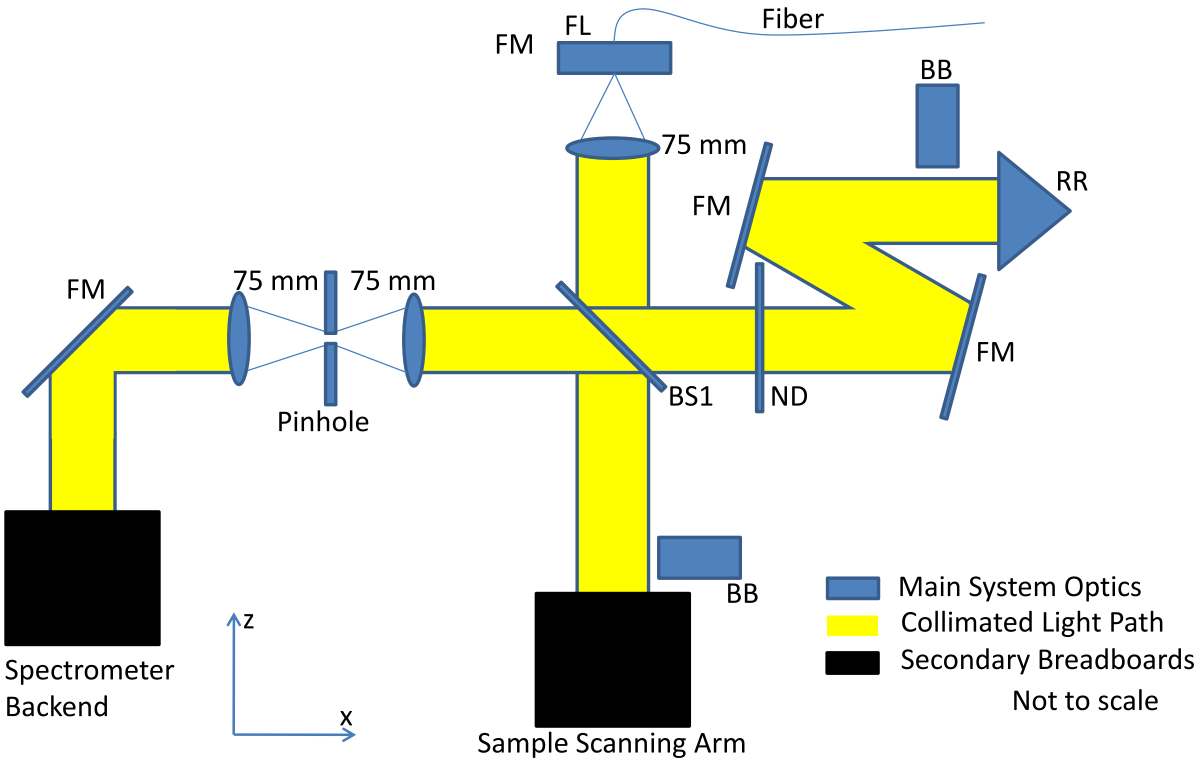


Figure 3.1: Main breadboard layout of the OCT system. Acronyms: BB beam blocker, BS1 beam splitter, FL fiber launcher, FM fold mirrors, ND neutral density filter, RR retroreflector. Lenses are identified by their focal length.

Inside the interferometer, we split the beam with a Thorlabs BSW17 non-polarizing 2" beam splitting plate and send the beam into our reference arm and our sample system. The reference arm primarily consists of a CVI Melles Griot CCH-25.4-1-LEBG 1" hollow retroreflector along with several beam steering mirrors (Thorlabs PF20-03-P01 are used for all our beam steering mirrors unless otherwise specified) to compress the beam path and neutral density filters to reduce the reference intensity. The light from the reference arm bounces off the retroreflector and returns to the beam splitter cube for recombination.

The sample scanning system consists primarily of a Nutfield QuantumScan-30 1" galvanometer scanning mirror system, a Thorlabs AC508-100-B 100 mm 2" NIR achromatic sample focusing lens, and a Nanomotion FB050 50 mm motorized translation stage. A pair of mirrors doglegs the beam up to the galvanometer, which allows us to scan laterally across our sample with micron level resolution by changing the angle we enter the sample lens. The light reflecting off the galvanometer enters the sample focusing lens and is focused onto a sample platform mounted on the Nanomotion translation stage. The Nanomotion

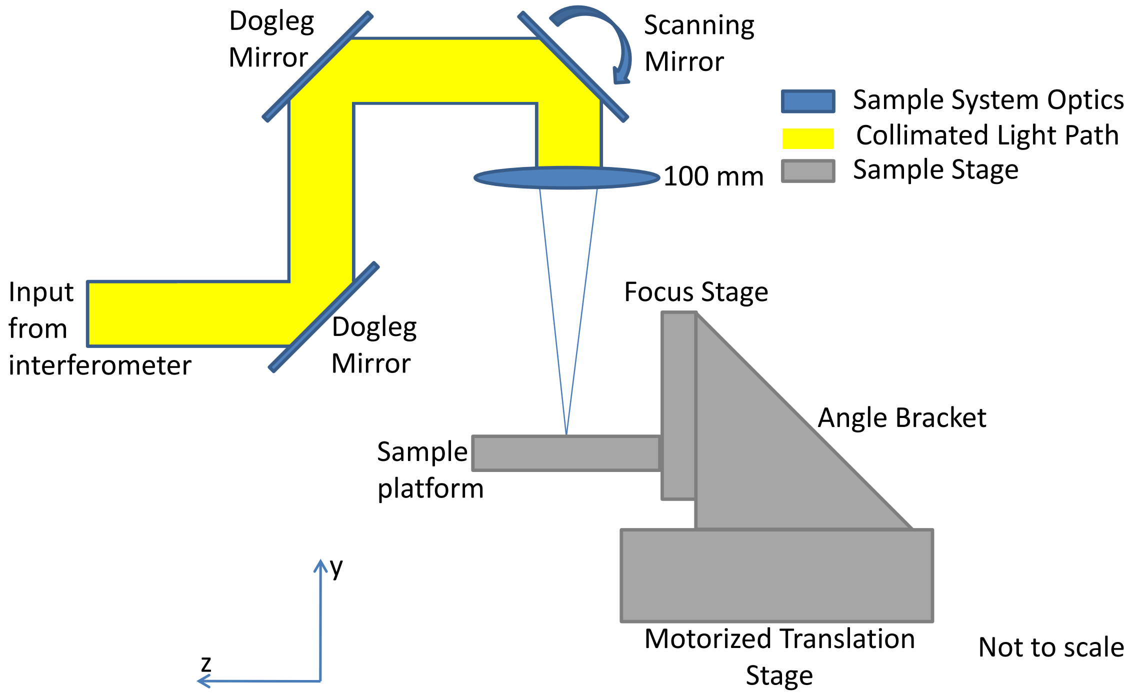


Figure 3.2: Sample Scanning System breadboard layout of the OCT system.

stage allows us to position our sample with 10 nm resolution and 50 nm repeatability and stability in a direction orthogonal to the galvanometer scan direction. Combined, the Nanomotion stage and Nutfield galvanometer allow us to scan across our entire sample, with the OCT interferometer providing depth information for a full 3D image. A New Focus 9064-X translation stage provides a sample focus adjustment of ± 14 mm. The light hitting our sample will scatter back into the sample focusing lens and return to our beam splitter for recombination.

When the light from both arms returns to the beam splitter, half the light is returned towards the fiber (and lost) while the other half is sent to a spatial filter system. The spatial filter system focuses the light with a Thorlabs AC254-75-B 75 mm NIR achromatic lens onto a Newport 910-PH10 10 $\mu$m pinhole. This pinhole ensures that only the light returning from near the diffraction-limited cone in the sample passes into the backend of our system. The light is recollimated with another Thorlabs AC254-75-B 75 mm NIR achromatic lens and enters our spectrometer backend.

In our spectrometer backend, we have a custom Kaiser Optical 1,200 lines per mm

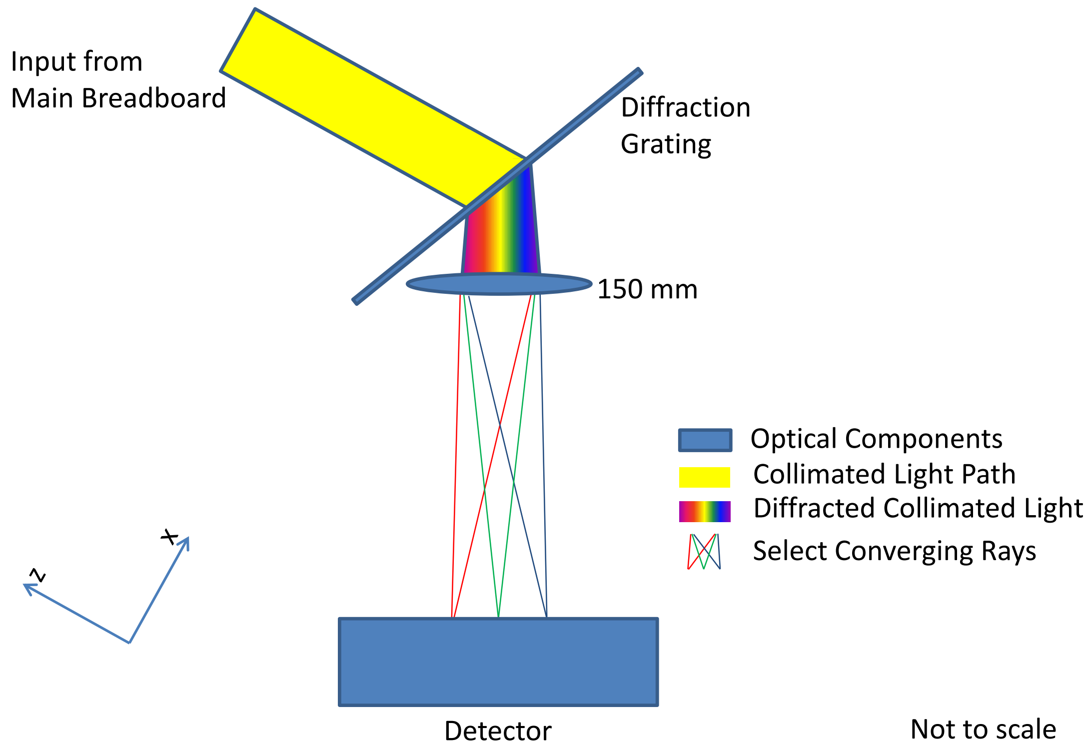


Figure 3.3: Spectrometer breadboard layout of the OCT system.

(l/mm) grating specially designed to maximize the spectral throughput from our light source. The collimated beam passes through our grating and the dispersed light is focused by a Thorlabs AC508-150-B 150 mm NIR achromatic lens. The light is focused onto a Basler Sprint spL2048-70km line scan camera and digitized into our computer by a National Instruments NI PCIe-1429 Camera Link image acquisition board.

## 3.3 Baseline OCT System Results

As configured, our baseline OCT system allows us to measure depth profiles, 2D slices, and 3D volumes of excised samples. Our standard imaging parameters provide us with cube voxels $\sim 3\mu$m per side. Table 3.1 provides the imaging parameter info of our OCT system. Our (very basic) reduction code, used to reconstruct all these images, can be found in Appendix A.3.

Figure 3.5(a) shows a sample image taken by our baseline OCT system of a fatty piece of pork. In this 2D slice, the surface layer and tissue/fat boundary are clearly visible

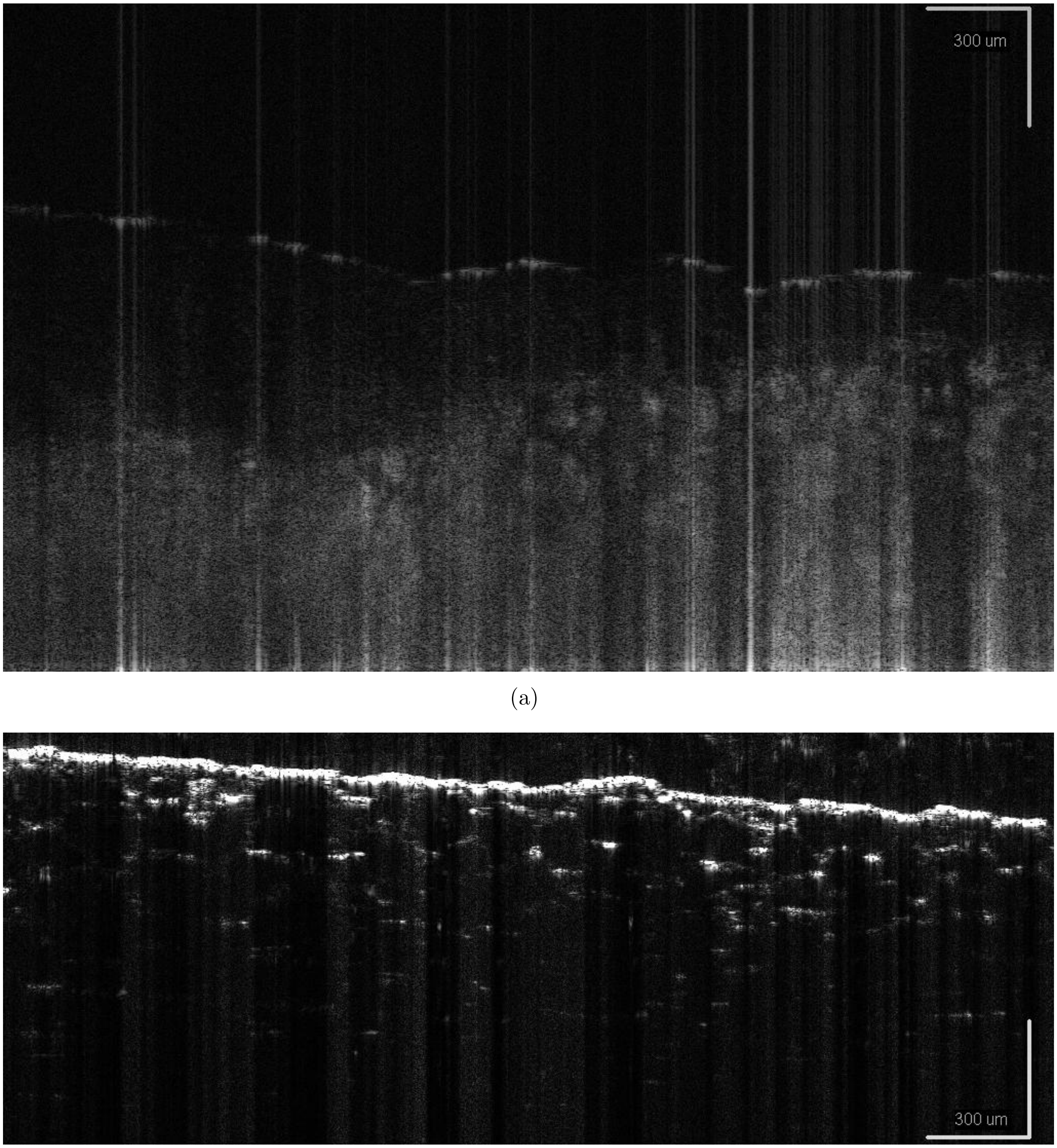
| Parameter | Value |
|---|---|
| Lateral Resolution in Air | 5.2 $\mu$m (theoretical) |
| Lateral Resolution in Tissue | 3.4 $\mu$m (theoretical) |
| Axial Resolution in Air | 3.7 $\mu$m (theoretical) |
| Axial Resolution in Tissue | 2.6 $\mu$m (theoretical) |
| Field of View in Air | 3.8 mm (measured) |
| Field of View in Tissue | 2.7 mm (theoretical) |
| Depth of Focus in Air | 198 $\mu$m (theoretical) |
| Depth of Focus in Tissue | 141 $\mu$m (theoretical) |
| Typical Galvonometer Induced Scan Step | 3 $\mu$m (measured) |
| Galvonometer Scan Range | 7 mm (theoretical) |
| Galvonometer Precision | 1 $\mu$m (theoretical) |
| Typical Stage Induced Scan Step | 3.5 $\mu$m (measured) |
| Stage Scan Range | 50 mm (measured) |
| Stage Precision | 10 nm (theoretical) |
| Sensitivity | 86 dB (measured) |

Table 3.1: The imaging parameters of our OCT system. For tissue, we assume an index of refraction $n = 1.4$.

with additional structure in between. Such an image allows us to determine structural information for our sample of interest.

Figure 3.5(b) shows a 2D slice from a sample of cucumber. Cell boundaries are visible with distinct differences between cell walls and cell interiors.

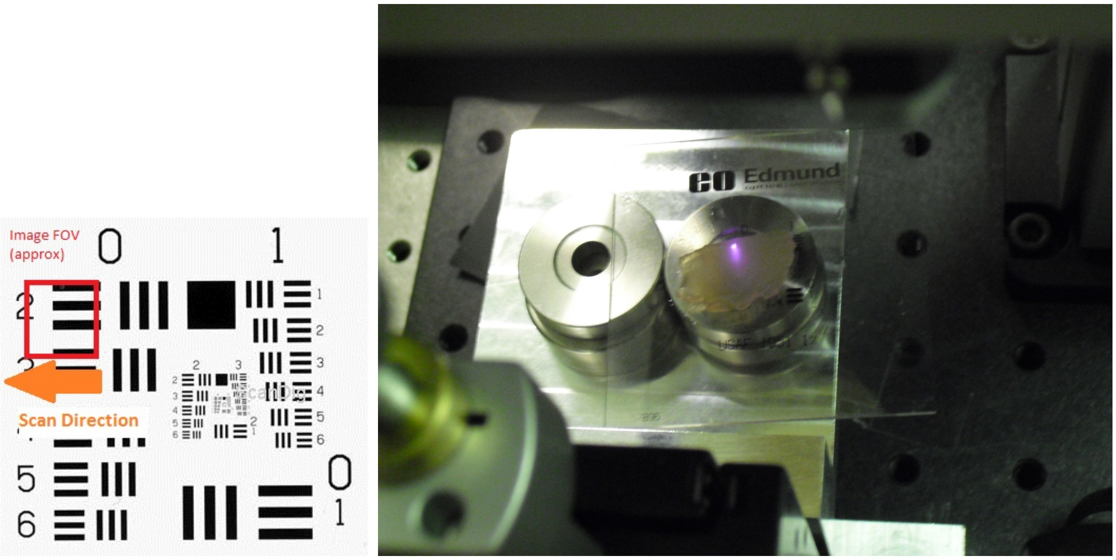
Our system also shows extreme depth penetration when compared to standard OCT systems. OCT normally sees ∼1 to 2 mm penetration depths in breast tissue [33]. Our system has identified structure under 3.3 mm of chicken breast tissue (figures 3.5, 3.6, and 3.7) and further testing indicates that this increase also allows tissue differentiation at depths significantly greater than 2 mm. While still under investigation (and not discussed further in this thesis), our free-space OCT system design already shows promise for enabling diagnostic capabilities previously not possible with OCT systems. Note that normal depth penetration measurements are different from the procedure used here—these are very preliminary results, but they show great promise.



(a)

(b)

Figure 3.4: 2D slice examples from our baseline OCT system. (a) shows an image of pork tissue over a layer of pork fat, clearly showing the boundary between the two layers. (b) shows a sample of cucumber with individual cells visible. Both images are plotted on a logarithmic intensity scale.



(a) USAF Resolution Target

(b) Experiment Setup

Figure 3.5: The experimental setup used to show the increased depth penetration possible with our system. (a) shows the airforce resolution target used to identify a point deep under our sample, with the region imaged indicated by the red square. (b) shows the chicken breast placed on top of the target with our OCT light focused into the sample. The chicken breast thickness is approximately 3.3 mm.

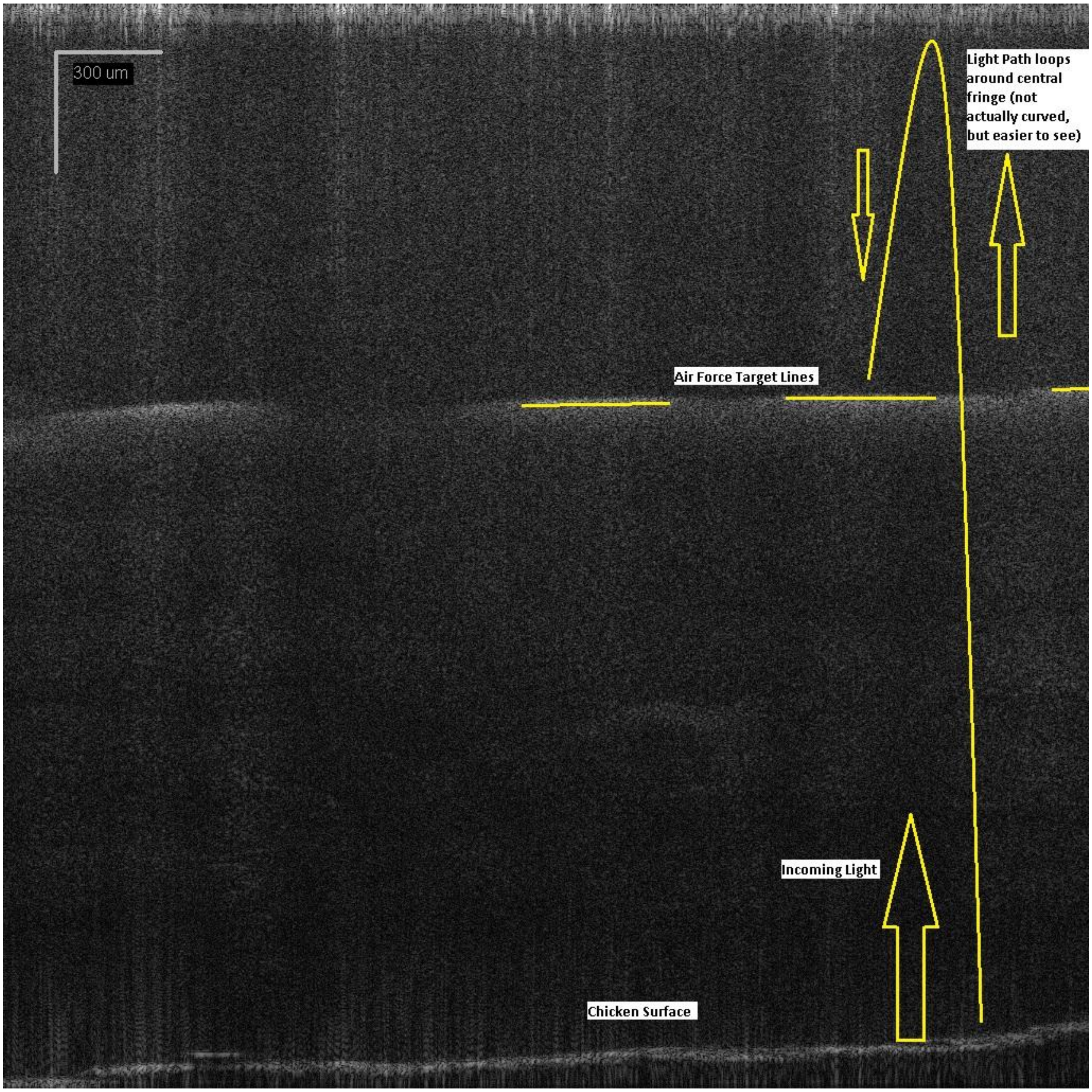


Figure 3.6: A representative 2D slice showing the path of light travel into our tissue in the experiment setup shown in figure 3.5. Note that the field of view of our system requires us to wrap around the central fringe in order to reach the target. This image is plotted on a logarithmic intensity scale.

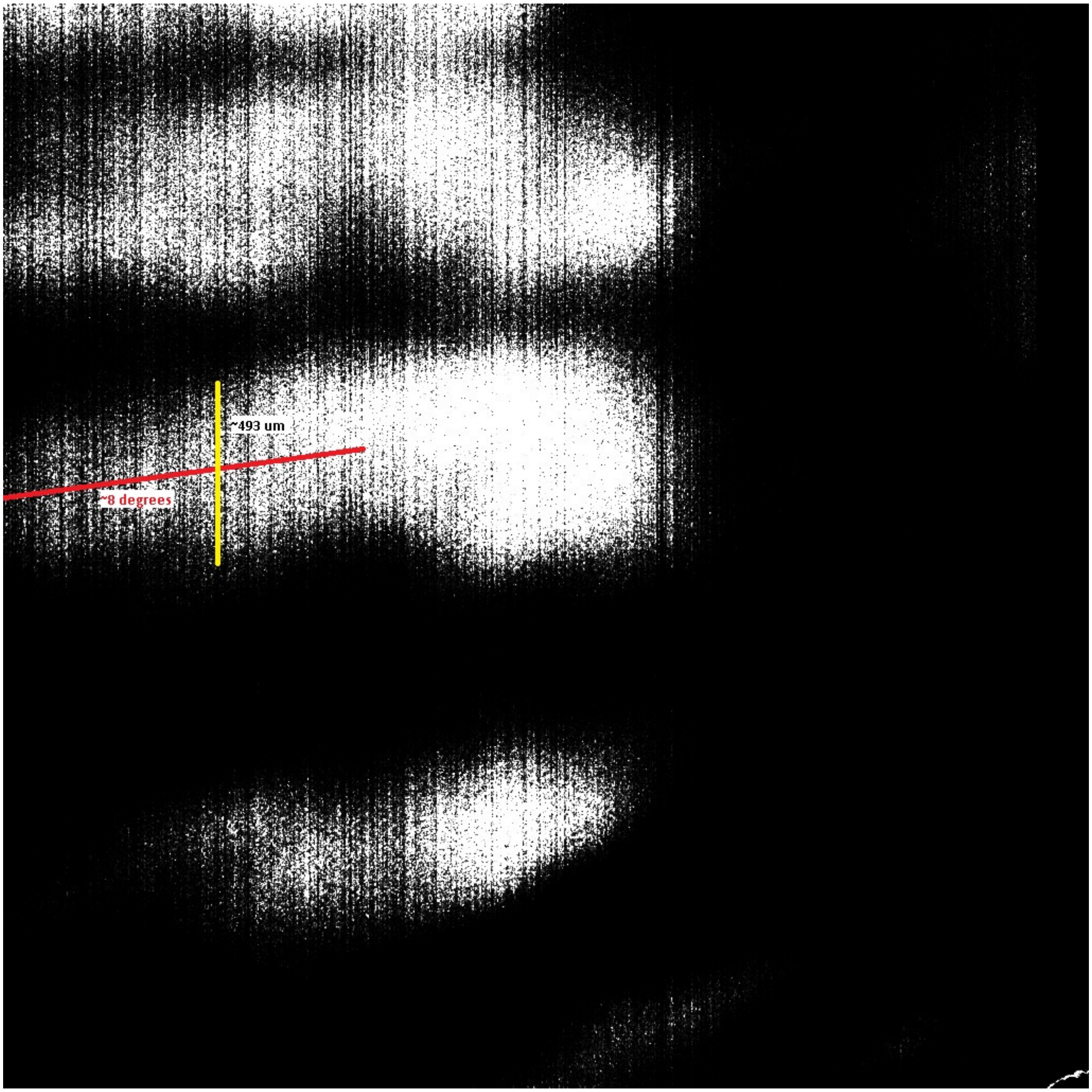


Figure 3.7: Slice from a 3D volume showing a constant depth layer coincident with the USAF target from the experiment setup shown in figure 3.5 (∼3.3 mm under the surface of the chicken). The yellow scale bar indicates a length of 493 $\mu$m while the red bar shows an 8 degree angle. This image is extracted from a 3D volume and thresholded to enhance detail.

# Chapter 4

# Alignment Enhancements

Given the OCT system described above, we wish to enable a user to use the system without needing to be an expert in the design and alignment. This section describes enhancements to the OCT system allowing a non-specialist user to align the system.

## 4.1 Expected Problems

With a fiber optic based design, most of the alignment is handled by the high precision couplers attached to the fiber optic components. This makes for simple assembly and a robust implementation but requires the use of fiber optics in the interferometer. Because we decided to use free-space optics for the advantages outlined in section 1.1.1, we need to worry about maintaining the alignment of our OCT system. This section describes the expected requirements of an automatic alignment system.

### 4.1.1 Alignment Effects on System Performance

In section 2.2.3, we discussed various parameters that a user cares about in an OCT system. The effect of alignment on these parameters is discussed below.

With serious misalignments, changes in the imaging wavelength, depth-of-focus, scanning range, imaging window, and imaging resolution can be seen. For reasons discussed below, these parameters are unaffected in our system–we see much smaller effects on system alignment than would be required to change these parameters.

The scanning speed and sensitivity, however, are heavily influenced by the efficient propagation of light through our system. If less light returning from the sample reaches the detector (for example, clipping on the pinhole), we will immediately see a loss of sensitivity and require longer integrations to reach the same image quality. By examining the mean level of light reaching our detector with a constant integration time, we can see the effect of system alignment on these parameters.

The dynamic range and penetration depth will be affected by changes in the fringe visibility or fringe contrast of our signal (see Equation 2.7). Most changes in fringe visibility will be due to relative intensity variations in the sample or reference beam intensity, which can be mitigated by adjusting neutral density levels. While alignment can play a role in some systems, the relatively high coherence of our system source (discussed more in Section 4.1.2) combined with our normal imaging setup limits the effects of alignment on our fringe visibility to a relatively small level.

### 4.1.2 Expected Tolerance Requirements

To obtain interferometric OCT fringes on our detector, we must maintain spatial coherence on our detector. We must also ensure that our light paths continue to propagate through our system.

Because of our FD-OCT based design, our temporal coherence constraints are limited by the bandwidth of a pixel in our backend spectrometer rather than by the bandwidth of our light source. Optimal use of our 2048 pixel detector with our 100 nm bandpass would give us pixel bandwidths of approximately 0.05 nm. Our light source has a central wavelength of 850 nm, giving us a coherence length of about 15 mm. Staying within the coherence length of our source should be relatively simple. Even with bandwidths many times the optimal, millimeter level offsets are acceptable.

The spatial coherence constraints of our system are determined by the angular size of our source and our pinhole. Both of these are of order $10\mu$m with a 75mm focal length focusing lens. This gives us a coherence area of about 45 mm$^2$ for our shortest wavelengths. This corresponds to a circular region with a diameter of approximately 7.5mm. This is about one third of our beam diameter and should also be relatively easy to maintain. Tests in the lab confirmed our minimal coherence effects.

While coherence is relatively easy to maintain, small tilt errors can greatly offset the position of the spots in our system. Assuming $10\mu$m spots with a 75mm focal length, an induced tilt of 30 arcseconds would be enough to move an entire spot width. A 30 arcsecond tilt would be induced by a $2\mu$m skew in a 1" diameter optic (and even less in our larger optics). A small fraction of this distance is enough to significantly affect our system. Such small errors are likely to occur and need to be corrected.

### 4.1.3 Expected Alignment Timescales

A Thorlabs LMR1 lens mount has an aluminum base height of about 10mm [40]. The coefficient of thermal expansion of aluminum is about 23 $\times 10^{-6}$ m/m °C near room temperature [34]. A 1°C temperature change would induce a shift of 0.2$\mu$m in this mount. When the combined effect of many such mounts is considered along with the hardware to affix these mounts in the system, a temperature change on the order of 1°C can have a relatively large effect on the efficiency of our system. With no thermal isolation, a person's body heat near the instrument could be enough to disrupt alignment. Without significant thermal isolation, alignment will drift as the system temperature changes.

Because all the components in our system are attached to a fixed breadboard and our integration times are very short (less than one millisecond), a small amount of vibrational isolation should place most of the alignment concerns on the temperature variations. Because we expect the system to be used indoors, it is likely that the temperature variations will occur on long timescales. In our laboratory environment, we are able to use the system with people in the room for several hours without significant image degradation but alignment improves system throughput, especially when performed before beginning any data collection.

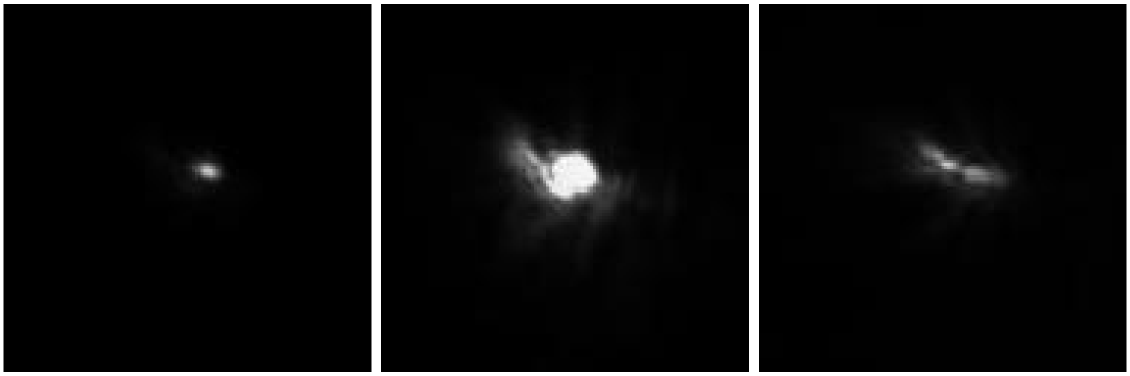
### 4.1.4 Mitigating factors

Because of the nature of OCT imaging, several mitigating factors present themselves, reducing the tolerances placed on the alignment system.

First, the light from our sample returns with a much larger effective spot than the specular reflection off a mirror surface (see figure 4.1). While some of this is multiply scattered light, the majority of the signal near the center of this spot is useful singly scattered light [47]. While we still wish to isolate a small portion of this light to focus on a specific lateral point in the sample, a small misalignment will primarily shift the point of interest rather than significantly reducing our returned signal.

On a similar note, the light in our reference arm needs to be significantly reduced (using neutral density [ND] filters) to provide an appropriate signal level to mix with our sample light. The primary result of a misalignment in the reference arm is a reduction in signal strength, with a secondary spectral shift due to a non-perfectly achromatic lens. The signal strength reduction is easily compensated by a change in ND value and experiment calibration data mitigates the spectral shift effect.

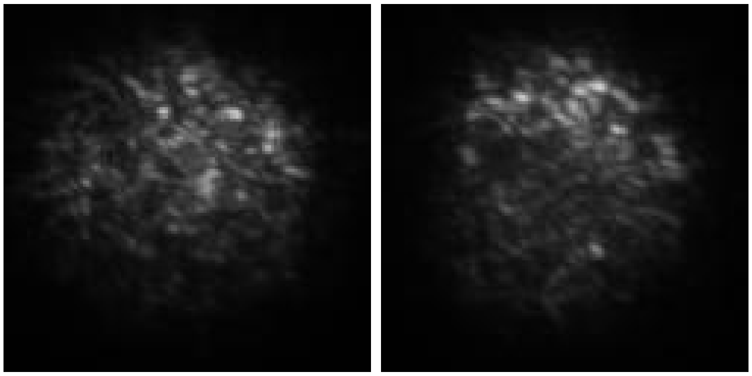
These two effects combined limit the instantaneous requirements on system alignment, moving the more stringent requirements towards long term stability of the system.



(a) Focused spot from mirror, reduced exposure time to avoid saturation

(b) Focused spot from mirror

(c) Focused spot from mirror through pinhole



(d) Focused spot from sample

(e) Focused spot from sample through pinhole

Figure 4.1: A comparison of the returned signal from a mirror in the focal plane of the sample arm and a representative scattering sample. Note the greatly increased size of the spot returning from the sample and the residual light from a mirror spot that does not pass through the pinhole.

### 4.1.5 User-level Requirements

A long-term goal of our alignment system is to maintain alignment in a commercial setting. In this environment, an important consideration is the impact on the user. While the specific requirements are outside the scope of this thesis (focusing on making the base system work), we must account for the presence of a user in a final design. We need to avoid requirements for highly specialized training to use our system. We need to ensure that the system can correct alignment errors in a relatively short period of time. Anything that will require user interaction will increase user requirements past the current standard—minimizing our impact on the user is a consideration in the design described below, although further work will be necessary in this area in the future.

## 4.2 Alignment System Degree of Freedom Reduction

Each component in an optical system has 6 degrees of freedom: translation and rotation axes for the x, y, and z dimensions. Aligning every possible axis of the components in a complex system is infeasible—we'd require well over 50 axes of control to accomplish this. A critical component in the design of our auto-alignment system is an analysis allowing us to reduce the required control axes. This requires both identifying insensitive degrees of freedom and combining complementary degrees of freedom into a smaller number of controls. In all cases we assume the errors we wish to correct are reasonably small, such as those caused by moderate temperature fluctuations or by small shocks to the system.

First, we must identify the degrees of freedom that cause a noticeable effect for our various types of components. As an example, all of our optics are rotationally symmetric, immediately removing one degree of rotational freedom from consideration. Table 4.1 ennumerates the effect of the various degrees of freedom on our optical components. This table makes assumptions based on our design—for example, all of our main OCT system mirrors operate on collimated light.

With small errors, the optical effects in our system compound. As an example, if a mirror is expected to induce tilt then the mirror tilt will be added to any original beam tilt. As long as the errors remain small, this allows us to correct an error in the system by adjusting a single component with the opposite effect. This principle allows us to greatly simplify our correction requirements.

We now need to identify how these noticeable degrees of freedom will affect our system. For this, we will break the system into five smaller subsections: fiber collimation, the reference arm, the sample arm, recombination, and the spectrometer.

Fiber collimation primarily consists of our fiber launcher and a collimating lens. From table 4.1, we need to worry about focus, shear, and tilt. The depth of field of our collimation

| Degree of Freedom | Fiber Launcher | Pinhole | Lens | Mirror/Beamsplitter | Retroreflector |
|---|---|---|---|---|---|
| Translation X | Tilt | Tilt | Tilt | — | Shear |
| Translation Y | Tilt | Tilt | Tilt | — | Shear |
| Translation Z | Focus | Focus | Focus | Shear & Path Length | Path Length |
| Rotation X | Shear | — | Focus | Tilt | — |
| Rotation Y | Shear | — | Focus | Tilt | — |
| Rotation Z | — | — | — | — | — |

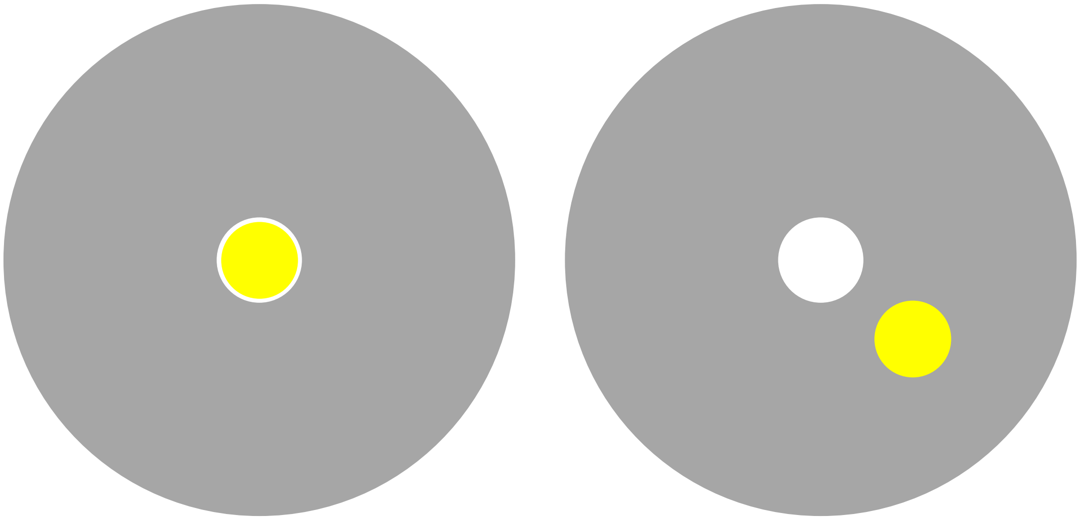
Table 4.1: The effect of degrees of freedom on our various optical components. The degrees of freedom are referenced to the centers of our optical components.

lens is large enough that we should not notice most focus misalignments—as an example, the thermal expansion of aluminum gives us a 15°C window before we exceed our depth of field. In addition, the focus of the sample arm compensates for a defocus entering the sample arm and an adjustment of ND power in the reference arm can compensate for lost light passing through the pinhole. Any shear introduced at this point will be small relative to our pupil diameter and will affect both arms of the interferometer equally, making any effect small. Tilts introduced here are very significant, though, with degree level temperature fluctuations shifting the spot location by large fractions of the spot size. Because of the sensitivity to tilt here, we introduce tilt corrections through the fiber launcher.

For the reference arm, light bounces off our beam splitter and a pair of fold mirrors and then enters our retroreflector. Because of the design of our retroreflector, light entering the retroreflector is reflected with the same tilt (with less than one arcsecond error) but offset in shear by double the original amount. The long path length in the reference arm also converts any tilts into a small shear. By reflecting off our fold mirrors twice, any residual tilt effect is removed but they can still induce additional shear. Overall, only the tilt induced by the beam splitter will affect the tilt of our reference arm output. Significant shear can be induced, though, and we add motorized shear control to our retroreflector to correct for this, allowing us to ensure overlap of our reference and sample beams.

In our sample arm, light reflects off several mirrors and is then focused by a lens onto our sample. In OCT, we are primarily concerned with the light that singly backreflects from our sample. This is light that is reflected back the same way it enters, which ensures that light entering the sample arm returns along the same path it enters. Any alignment errors in the sample arm correct for themselves as the light travels back along the path it enters. It then reflects off of our beam splitter and gains the same tilt induced before light entered the reference arm.

After passing through the reference and sample arms, our light must be recombined through our spatial filter pinhole. At this point in our system, we have an initial tilt and shear introduced by our fiber collimation, tilt induced by our beam splitter, and shear from the reference arm. The shear in the reference arm can be corrected through motorized shear control in the reference arm. This leaves a tilt and small shear to our beams. The residual shear will be a small fraction of our collimated beam diameter and should cause little issue. The tilt of our beam will determine our spot location—we must ensure that the spot location and pinhole location coincide (see Figure 4.2). By manipulating our tilt through the fiber launcher, we can correct any misalignment and pass our OCT signal into our spectrometer backend.


(a) Aligned Pinhole

(b) Misaligned Pinhole

Figure 4.2: Figure showing the difference between an aligned and misaligned pinhole plane. In the aligned case, the central lobe of the beam cleanly passes through the hole in the pinhole plane. In the misaligned case, the focused beam hits the pinhole plate and does not make it through to the backend of the system. Note that the airy rings are ignored in this drawing.

Experience with the spectrometer in our system in the lab suggested that alignment would not be necessary—months of sitting in the lab environment and needing tilt and shear correction in the interferometer had not required spectrometer adjustment. Temperature testing shows a need for alignment with large temperature changes, and primarily involves vertical position on the focal plane which can be adjusted by tilting one axis of our fold mirror. Because of the small vertical height of our detector, this is the most sensitive degree of freedom in the spectrometer. Horizontal position is relatively insensitive due to the large focal plane width (assuming spectrometer calibration is performed), the depth of field is large enough that focus should not be an issue, and any shear induced should be small.

Despite all the potential locations for misalignments, our analysis suggests that two axes of tilt control and two axes of shear control should be enough to adequately maintain the alignment of our system. With large temperature variations (larger than those seen in our laboratory environment under normal conditions), an additional axis is required to control the vertical position of our spectrum in our spectrometer.

## 4.3 Alignment System Design

In order to maintain system alignment, additional hardware to monitor and adjust the alignment is required. To minimize the cost and complexity, the number of alignment components needs to be minimized. This requires identifying the unique degrees of freedom in the system and providing monitoring and control devices for them.

Examination of the system shows that four degrees of alignment freedom should be enough to monitor and maintain interferometer alignment (see section 4.2). The tilt of the beam entering the interferometer needs adjustment to ensure the spots in the system pass through the pinhole. In addition, the retroreflector position needs adjustment to ensure the two interferometer arm beams are coincident. By monitoring and controlling these four degrees of freedom (vertical and horizontal tilt and shear), it is possible to correct for most system drifts. By aligning our system at the pinhole, we can ensure a clean interferometric signal enters the backend with both the reference and sample beams coincident.

Implementing system tilt control is possible by moving the position of the input fiber relative to the collimating lens. Using a New Focus 8051 pico fiber launcher allows for positioning the fiber with 30 nm step sizes over a ±3 mm range. With our 75 mm collimating lens, this allows for tilt adjustments of approximately 80 milliarcseconds over a ±2° range. This is plenty to maintain alignment at a high level.

Mounting the retroreflector on two orthogonal translation stages (New Focus 9067-COM) with two attached New Focus 8302 picomotors allows for shear adjustment of the returning reference beam. The New Focus 8302 picomotors provide for ±0.5” of translation with 30 nm step sizes, allowing us to maintain coincidence at a tiny fraction of the beam diameter.

In addition to adjusting the system alignment, we need some way to measure the deviation from proper alignment and determine the required corrections. Ideally, the system should be able to monitor alignment at all times while being minimally invasive. Because we expect our alignment drifts to have a long time frame relative to the acquisition rate of

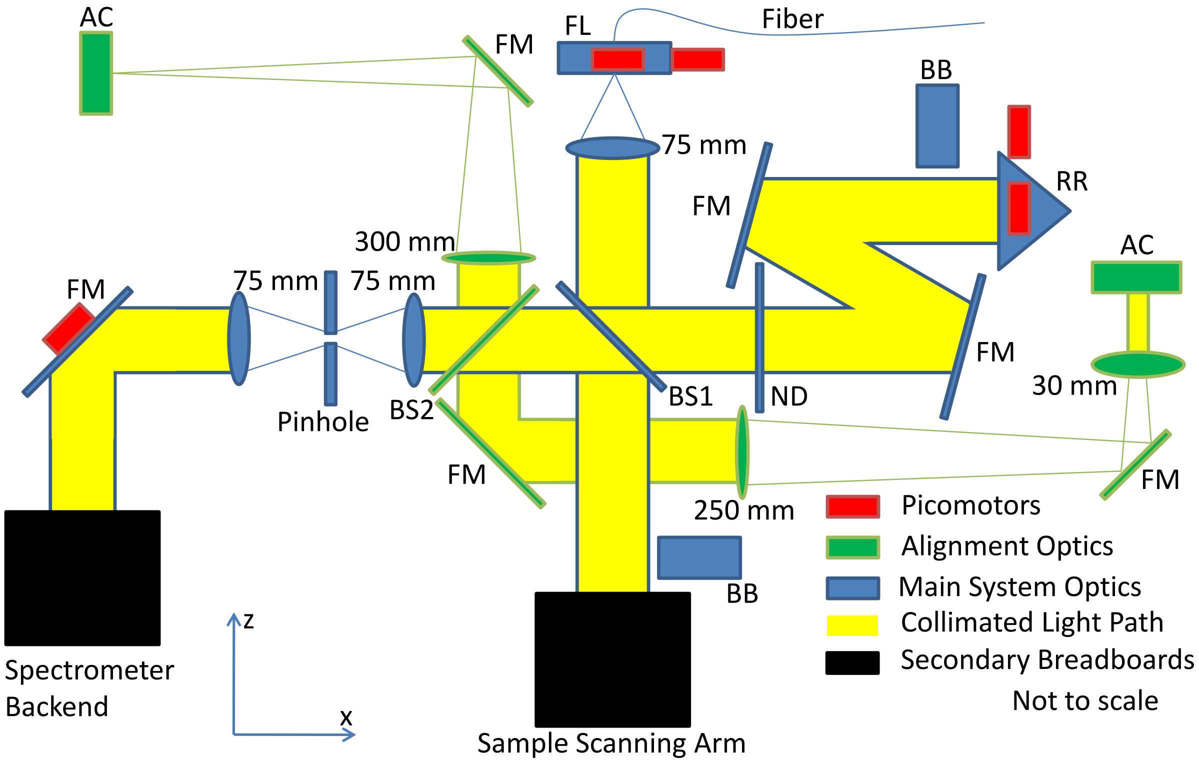


Figure 4.3: A cartoon drawing indicating the alignment hardware required to monitor and maintain system alignment in the OCT system. The alignment optics provide the primary alignment monitoring system while the picomotors provide the primary alignment control. Acronyms: AC alignment camera, BB beam blocker, BS1 beam splitter, BS2 beam sampler, FL fiber launcher, FM fold mirrors, ND neutral density filter, RR retroreflector. Lenses are identified by their focal length.

our system, a tiny fraction of the light from our system is sufficient to monitor the alignment. Using a 0.2% anti-reflection (AR) coated beam sampler, we are able to maintain sufficient frame rate for our alignment system while maintaining the vast majority of our system throughput.

In order to monitor the presence of a tilt offset, we use a reflective pinhole and implement a reimaging system. Placing the beam sampler before the pinhole focusing lens but after the beam splitter cube sends an image of the pinhole plane out of the beam path of the interferometer. By focusing this light using a Thorlabs AC254-300-B 300 mm focal length achromatic lens onto a IDS UI-1225LE-M detector we image the pinhole with a 7 pixel diameter. This allows us to measure our tilt offset at the sub arcsecond level. Adjusting the focal length of this imaging system allows us to trade off measurement accuracy for measurement speed.

Because the beam sampler reflects the light reflecting off the pinhole and the light entering the spatial filter system in opposite directions, we can use the same beam sampler to also allow imaging of the pupil offset of our reference and sample beams. Imaging this beam through a beam reducer with another IDS detector allows us to measure the coincidence of our reference and sample beams. By adjusting the parameters of our beam reducer, we can optimize the imaging speed versus the measurement accuracy.

For environments with expected large temperature fluctuations, an additional axis of control is necessary on the fold mirror feeding our spectrometer. A single Picomotor attached to the vertical axis of a Thorlabs KM200 kinematic 2” mirror mount gives us the control flexibility we need for this axis. With the goal of maintaining light on a detector with large system variations, simply using the final system detector to correct for offsets in this axis provides the necessary alignment monitoring.

## 4.4 Additional Considerations

Implementation of our alignment system in our OCT system has led to several changes in the overall system design. The most significant ones are detailed in this section.

### 4.4.1 Beam Splitter Choice

Our original OCT design used a beam splitter cube to split and recombine our light. In implementing the alignment monitoring system, the cube was responsible for several significant issues. These issues were solved by replacing the beam splitter cube with a beam splitter plate.

A beam splitter plate has several advantages and disadvantages over a beam splitter cube or a pellicle as the primary beam splitter in a free-space OCT system designed with auto-alignment. For the current design, the advantages of a beam splitter plate outweigh the advantages of the other choices. This section details the advantages and disadvantages of beam splitter plates, cubes, and pellicles.

#### Displacement

A beam splitter plate causes a shear displacement in the beam. Because we enter a thick plate with a higher index of refraction at an angle, our beam is refracted. Upon exiting the plate, we have the same output direction but will have a shear. Figure 4.5(a) shows this effect.

Because we enter a cube face on, refraction (and any resulting shear) is minimized (see figure 4.5(b)). A pellicle also sees minimal offset due to its extremely thin (∼2 $\mu$m) size (see figure 4.5(c)).

If the beam splitting plate and beam sampling plate are at reverse angles, the displacement of one plate offsets the displacement of the other plate. Matching the beam sampler and beam splitter allows us to significantly reduce the displacement effect—by choosing both a beam splitter plate and a beam sampling plate, this effect is greatly reduced. Additionally, the system can be built knowing that a fixed shear is induced by these components by shearing any following components appropriately.

### Dispersion

Because a beam splitter plate has glass on only one side of the reflective surface, one of the two interferometer arms will travel through additional glass in the beam splitter plate (approximately twice the thickness of the glass plate for the arm farthest from the reflective surface). This additional glass will cause dispersion, leading to a dispersion mismatch in the two arms.

A beam splitter cube, on the other hand, causes both arms to travel through a similar amount of glass (reasonable cube tolerances would give ±0.2 mm path length deviation [40]). This causes a similar amount of dispersion in both arms of the interferometer.

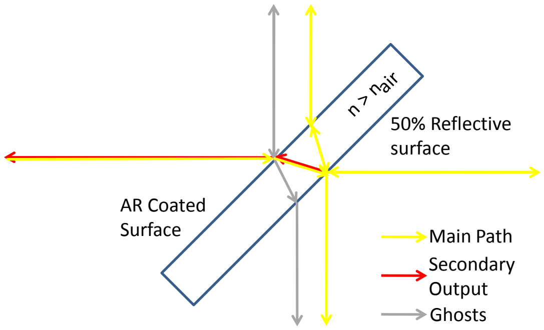
Similarly, the thinness of a pellicle causes very little dispersion at all (relative or absolute).

But, given the design of our system, we expect more dispersion in the sample arm than the reference arm due to the sample focusing lens. Combined with the neutral density filter in the reference arm, a beam splitter plate can help minimize a dispersion mismatch between the two arms without additional dispersion matching optics.

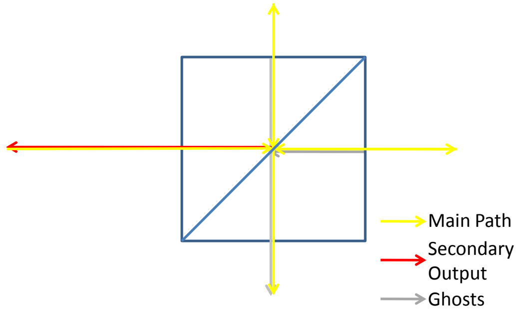
### Ghosting

As seen in figure 4.5(a), plate ghosts follow the main beam. Standard plate beam splitters also have the antireflection coated surface tilted by 30 arcmin. If these ghosts propagate through the system, they should see similar path lenghts and attenuation to the main beam but will be offset by a several millimeter shear and a 30 arcmin tilt. If they don't propagate, we don't care about them.

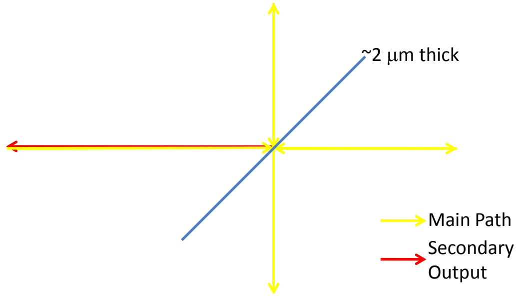
Cube ghosts will be caused by the reflection off the flat output surfaces and will travel opposite the desired direction. In a single cube interferometer, a significant part of the ghosting will exit the interferometer directly (instead of travelling fully through the two



(a) Plate



(b) Cube



(c) Pellicle

Figure 4.4: Displacement of the beam and ghosts due to a beam splitter plate (a), cube (b), and pellicle (c).

arms). These ghosts will not have a similar path length to the main interferometer, and any attenuation (such as caused by travelling through tissue or passing through a neutral density filter) in the interferometer serves only to increase the relative intensity of the ghosts. In addition, given the closely matched cube surface distances, a significant interference effect can be induced.

Pellicle ghosts are almost coincident, due to the ∼2 μm thickness of the membrane. While this effectively removes any ghosting effect, it causes an interference based sinusoidal spectral profile, discussed more below under the throughput heading (page 40).

### Size

Using standard off-the-shelf optics, a 1” BSC can be used in our system whereas a 2” plate or pellicle beam splitter would likely be needed. The size of the plate facing the beam at 45° would be $\sqrt{2}$”, an increase of about 40% per side over the cube and double the height. This will usually require more space in the system layout.

Using an elliptical plate with a $\sqrt{2}$” length and 1” height removes most of this difference if necessary.

### Cost

Buying off-the-shelf, a cube is slightly more expensive than an appropriately larger plate. A pellicle is in between the cost of a plate and cube. The total cost range is approximately 25% [40] and a relatively small portion of the total system cost.

### Durability

Both a beam splitter cube and plate are durable optics. Pellicles are extremely delicate, being ∼2 μm thick membranes—simply brushing across the surface of a pellicle is enough to destroy it. In a system designed for use outside a lab environment with non-expert users, the delicate nature of a pellicle can be a large disadvantage.

### Throughput

The ThorLabs plate beamsplitter (BSW17) has a throughput graph showing around 98% throughput in our wavelength range. The Newport cube (10BC17MB.2) has a throughput graph showing closer to 90% throughput. The ThorLabs plate beamsplitter has much larger deviations between S and P polarization than the Newport cube (sometimes as much as a 2x deviation for the plate) but our system operates on unpolarized light.

Due to the extremely thin nature of pellicles ($\sim$2 $\mu$m thick), there is an interference fringing effect seen in their spectral throughput [40]. Because our system uses a similar fringing effect to measure the OCT depth profiles, a significant noise signal might be present in our data when using a pellicle beam splitter. While the throughput is high (there is no ghost reflection loss and absorption is very low in the thin membrane), 50/50 pellicles are generally only available for a narrow wavelength range. 45/55 pellicles work over a much larger range (such as that required by our system) but the uneven splitting causes approximately 1% loss in our output beam (with that 1% extra light being "lost" in the secondary output, with a mismatched reference and sample power). This is further amplified by the spectral reflection profile, often having worse (or better) performance at specific wavelengths.

### 4.4.2 Beam Reducer vs Pupil Projection Imager

To measure the beam shear offset, we need to measure the pupil position of the two beams of our interferometer. The simplest method of accomplishing this measurement involves projecting the pupil onto a screen and then imaging the screen—this is analogous to a user inserting a viewing card into the beam and inspecting by eye. This proved inadequate for our system and was replaced by a beam reducer to allow direct measurement of the pupils on our detector. This section discusses the tradeoffs between the two methods.

#### Component Cost

For the pupil projection imaging scheme, we had anticipated using an inexpensive projection screen (such as a small square of poster board) combined with a lens, a fold mirror, and our detector. Changing to the beam reduction scheme required replacing the projection screen with a mirror and adding a second larger lens to the system. While a significant cost increase for this specific subsystem, the subsystem is still a relatively inexpensive component of the full alignment system.

#### Image Quality

The primary driver for replacing the pupil projection imager with a beam reducer was low pupil image quality. While the system provided good image quality in the visible region, pupil images in our system appeared washed out and diffuse for all the projection screens we had easily available. This is likely caused by the deeper penetration of infrared light into the screen materials—shorter wavelengths (in the visible) should not have this problem. Figure 4.5 illustrates the issue. Changing to the beam reduction scheme provides crisp pupil images irrespective of the wavelength.

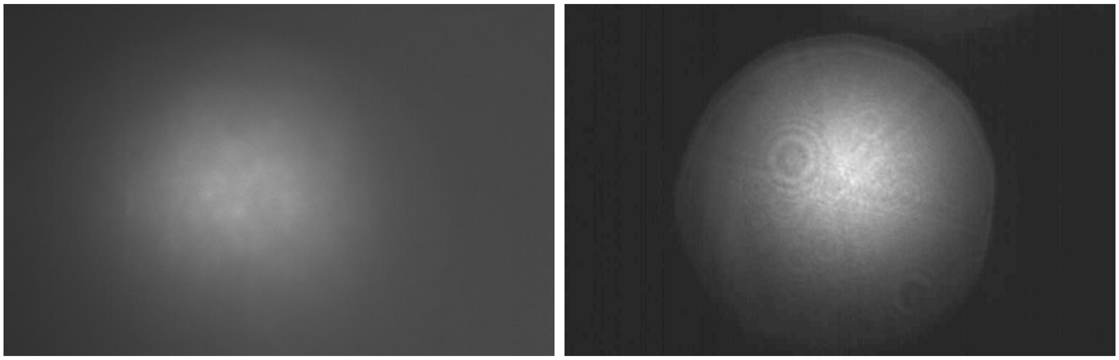

(a) Pupil Projection (b) Beam Reduction

Figure 4.5: The pupil images obtained through a pupil projection imager and a beam reducing imager in our system. Note the much sharper image quality in the beam reduced image. The images have been enhanced to increase contrast when printing.

### Throughput

In addition to cleaner images, the beam reduction scheme provides significantly higher throughput. Projecting the pupil onto a screen relies on scattering the signal and imaging the scattered light. A significant fraction of this scattered light does not reach the detector. While this scattered light shouldn't contribute significantly to background levels (our projected pupil is a tiny fraction of the primary system signal), the beam reduction scheme avoids throwing this signal away and is more sensitive as a result.

### Ghost Reduction

Because of the design of our beam sampler, two similar intensity pupils (one signal, one "ghost") offset with a 30 arc minute tilt and sheared by several millimeters are output. With the projection screen, unless a large distance is inserted between the splitter and the screen, both of these ghosts will hit the screen and are detected by the camera. The beam reduction method increases the offset of the ghost in the field of view and allows for blocking the ghost in the focal plane of the beam reducer.

## 4.5 System Implementation

In this section, we discuss the hardware used to implement the alignment monitoring and control of our OCT system.

### 4.5.1 Auto-alignment Components

An important point to note is that the required alignment hardware is available off-the-shelf, other than the mounts used to attach the hardware to a breadboard (these are largely the same custom mounts used in the main OCT system and could be replaced by posts and pedestals in preliminary testing). While the design of the alignment system can change for different optical systems, there is no need for a long development and manufacturing period to work with custom components. This allows significant flexibility in the design and testing phase, reduces costs, and makes the extension to other systems significantly easier.

### 4.5.2 Alignment Hardware

The alignment system is designed to be a minimally invasive upgrade to the baseline system. Control of the system alignment is provided by pairs of New Focus Picomotors mounted on our fiber launcher for tilt control and our reference retroreflector for shear control.

We use a New Focus 8051 motorized fiber positioner as the tilt control mount—this is a two axis fiber positioner with Picomotors already attached. We glue a Thorlabs S05FCA fiber adapter plate to this mount to allow us to attach the fiber from our SLD source. The light exiting the fiber enters the main OCT system and follows the same beam path described in chapter 3. Using small changes in the position of the fiber at this point, we induce small tilt corrections through the rest of the system.

For shear control, we replace the micrometers on our retroreflector mounting setup (two New Focus 9067-COM translation stages connected with a New Focus 9161 angle bracket, see chapter 3) with New Focus 8302 1” Picomotors. This setup allows us to move the retroreflector perpendicular to the beam travel direction—the design of our retroreflector converts this movement into a beam shear.

To control the vertical position of our spectrum on our detector, we use a New Focus 8354 Tiny Picomotor to replace the tilt control on one axis of the Thorlabs KM200 kinematic mount used to feed light into our spectrometer. This allows us to ensure that our output spectrum is vertically aligned with the 20 $\mu$m pixel height on our Basler detector. Simply providing small tweaks to this motor while examining the response on the detector is sufficient to align this axis.

To monitor the alignment of the main interferometer, we have placed a Melles Griot W2-IF-2025-C-670-1064-45UNP plate into the beam at a 45° angle. This plate reflects a tiny fraction ($\sim$0.2%) of the incident light towards a shear monitoring system. In addition, light reflected back from our pinhole also reflects off this plate and is sent to our tilt monitoring system. This plate is positioned between the plate beam splitter and the pinhole focusing lens and operates on collimated light.

The shear monitoring system beam-reduces our sample and reference arm beams using a Thorlabs AC508-250-B lens and a Thorlabs AC254-030-B lens in series, forming a 8.33x beam reducer. The reduced collimated beam is sent to an IDS Imaging UI-1225LE-M-GL camera, allowing us to monitor the reference and sample pupil locations with high precision. Two beam redirection mirrors allow us to position this setup in a convenient location in the OCT system (see figure 4.3).

The tilt monitoring system focuses the light reflected back from the pinhole using a Thorlabs AC254-300-B lens onto an IDS Imaging UI-1225LE-M-GL camera. Using this system, we are able to monitor the position of our focused spots on our pinhole to submicron accuracy. As the position of our spots on the pinhole is a direct result of the tilt of the beams, this system provides very precise tilt monitoring. As above, we also use a beam redirection mirror to more conveniently place this monitoring setup in our OCT system.

Figure 4.6 shows the OCT system with alignment monitoring and control hardware inserted. The baseline OCT system continues to perform as before.

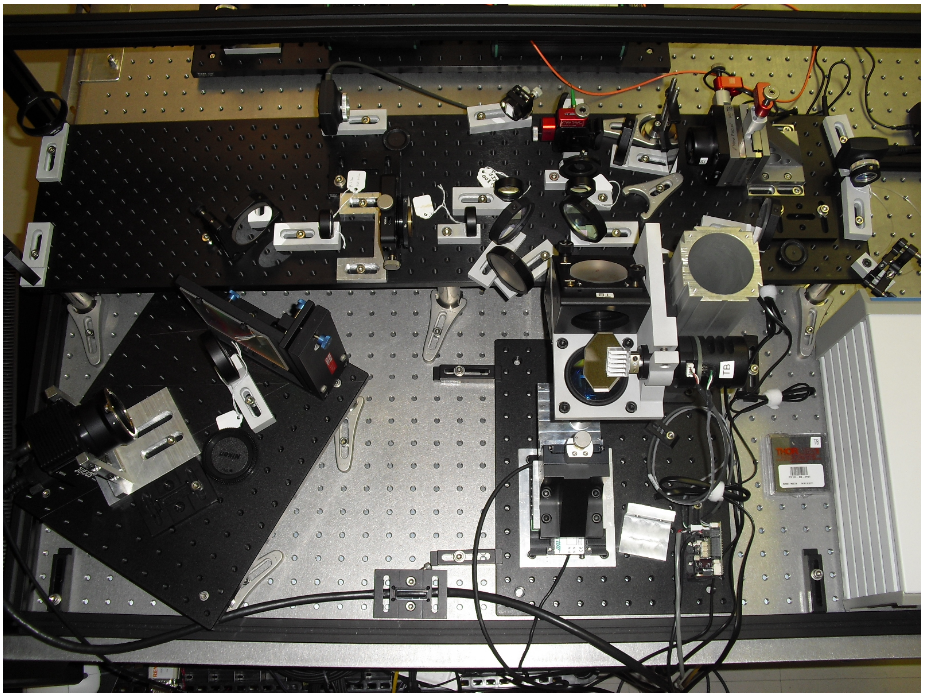
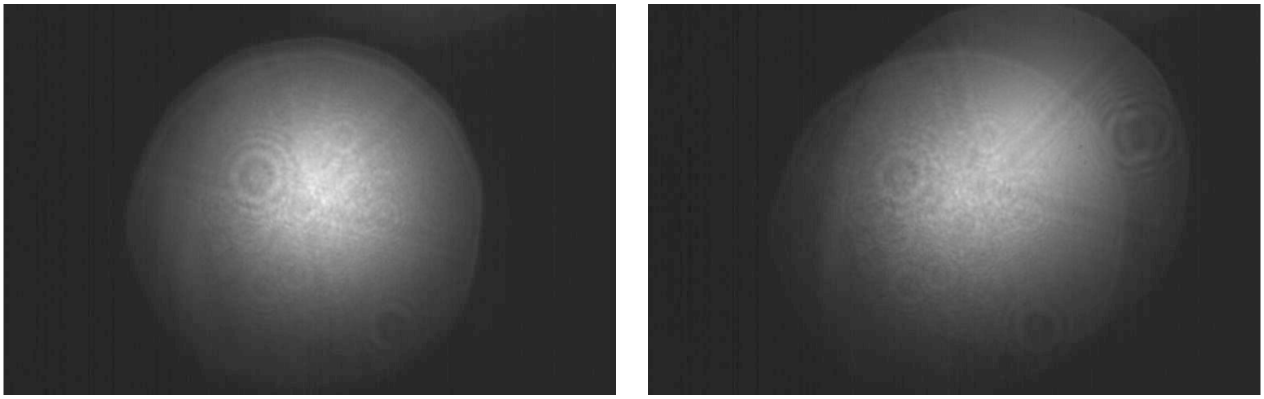


Figure 4.6: The OCT system as currently configured.

Figures 4.7 and 4.8 show the ability of our shear and tilt monitoring systems to see offsets in shear and tilt alignment. The system was first aligned and images from both systems were recorded (figures 4.8(a) and 4.9(a)). After inducing tilt and shear errors in the system, additional images were captured (figures 4.8(b) and 4.9(b)). The systems both clearly show large deviations between the aligned and misaligned images.



(a) Aligned

(b) Misaligned

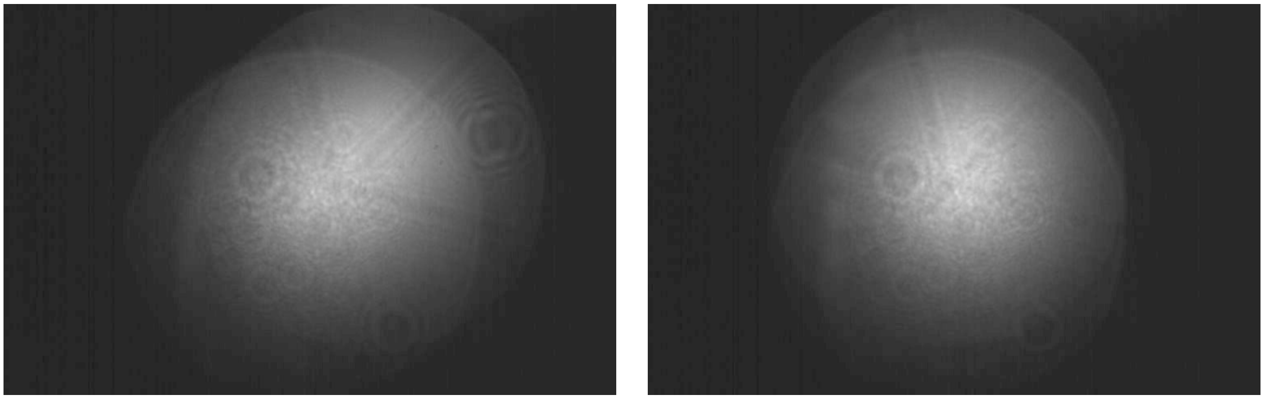
Figure 4.7: Images from the shear alignment monitoring camera. Shown is a view of the system properly aligned as well as a view of the system severely misaligned. The images have been enhanced to increase contrast when printing.

(a) Aligned

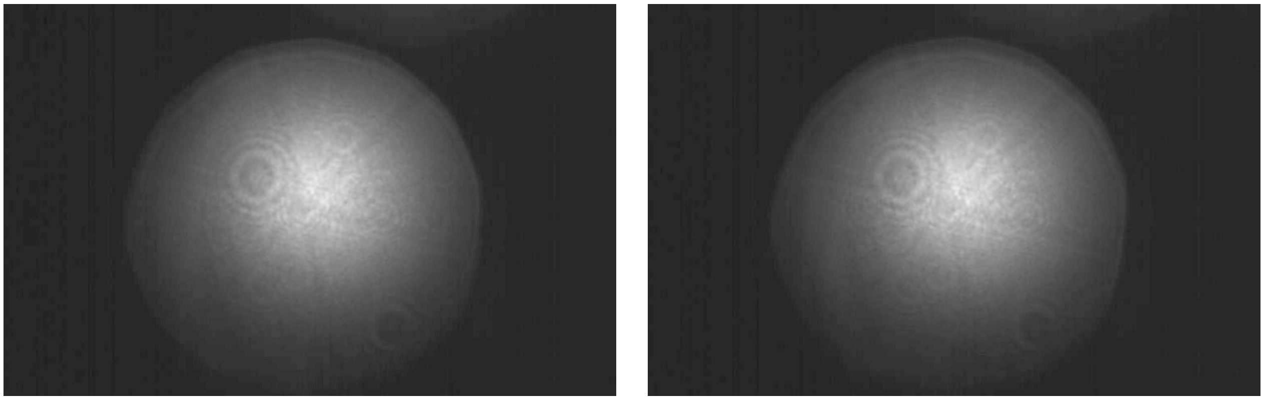
(b) Misaligned

Figure 4.8: Images from the tilt alignment monitoring camera. Shown is a view of the system properly aligned as well as a view of the system severely misaligned.

Figures 4.9 and 4.10 show the system's capability to return the misaligned system back to an aligned state. Using only the alignment motors, these figures demonstrate correcting the errors introduced previously (figures 4.10(a) and 4.11(a)). In each case, we correct one axis by moving the motor in a single direction until the system appears aligned for that axis (figures 4.10(b) and 4.11(b)). We then adjust the second axis in a similar fashion, bringing the system back into proper alignment (figures 4.10(c) and 4.11(c)). Comparing the realigned images to the original state (figures 4.10(d) and 4.11(d)), close agreement is seen.



(a) Misaligned (b) Axis 1 Corrected



(c) Axes 1&2 Corrected (d) Original

Figure 4.9: Images from the shear alignment system showing the progression from an unaligned to an aligned state. (a) shows the system significantly misaligned. (b) uses one axis of the shear control to improve alignment. (c) adds the second axis of shear control to further improve alignment. (d) is the original aligned image for reference. The images have been enhanced to increase contrast when printing.

### 4.5.3 Extra Benefits

In addition to the obvious benefits of this alignment hardware, a few additional features are also present.

(a) Misaligned (b) Axis 1 Corrected

(c) Axes 1&2 Corrected (d) Original

Figure 4.10: Images from the tilt alignment system showing the progression from an unaligned to an aligned state. (a) shows the system significantly misaligned. (b) uses one axis of the tilt control to improve alignment. (c) adds the second axis of tilt control to further improve alignment. (d) is the original aligned image for reference.

First, the alignment monitoring setup can aid in the initial alignment of the system. Initial alignment, especially of the pinhole, can be daunting but the alignment monitoring systems provide additional feedback that can greatly assist in this task. The tilt camera shows a view of the pinhole plane and can help locate the focused spot position, the pinhole position, and give an indication of the focus of the spot on the pinhole. The shear monitoring system can then help ensuring the sample and reference arms are coincident.

Additionally, the tilt camera can provide visual feedback on the sample under examination. The sample focus plane is reimaged onto the pinhole plane. Because our tilt camera reimages the pinhole plane, we are able to see the sample reimaged. This allows us to see the light returning from the sample and analyze features such as the sample focus, basic structural features in the sample, and the intensity of light returned from the sample.

Furthermore, a significant amount of system alignment control can be accessed directly through the computer. Even if the automatic features of the system fail, a technician can remotely connect to the system and potentially correct the error without needing an on-site visit. At worst, additional diagnostic information would be available to aid in planning an on-site visit. Further enhancements may be needed to take full advantage of this feature (such as additional software) but the basic functionality for remote examination and manipulation is present.

Finally, the alignment system can allow us to deterministically misalign the system. While this might sound like an odd feature of an alignment system, there are occassions where this ability can be useful. As a specific example, slightly misaligning the reference arm shear can reduce the reference intensity reaching our detector—as discussed in section 2.1.4, matching the reference and sample intensity provides increased interferometric signal. Exploiting this feature could allow for improved imaging quality or have other unconsidered effects.

While we currently take advantage of some of these features in the lab, these are not built into the alignment software. Additional work could enhance these benefits and discover additional features but they are not further explored in this work.

## 4.6 Software Control of Alignment Hardware

The alignment system requires computerized control of systems unique to the alignment hardware and is aided by access to several of the main OCT subsystems. The various subsystems and the necessary controls are outlined below.

### 4.6.1 Alignment Motors

Our system uses five New Focus Picomotor motors to control the alignment. We require the ability to move each of these motors, commanding them to move by a specific amount relative to their current location. The computer communicates with the Picomotors by sending commands through an ethernet connection to the Picomotor controller. Figure 4.11 shows what a Picomotor looks like. These motors have less than 30 nm step sizes (depending on loading), providing the small adjustments necessary to keep our system aligned, and maintain position when inactive (removing the need to constantly power the motors to maintain position). They do have large ($\sim$ 20%) repeatability errors, different forward and reverse movement performance (dependent on loading), and slow large-scale motion (1.2 mm/minute) but these effects are compensated by our monitoring and control software and the requirements of our system.

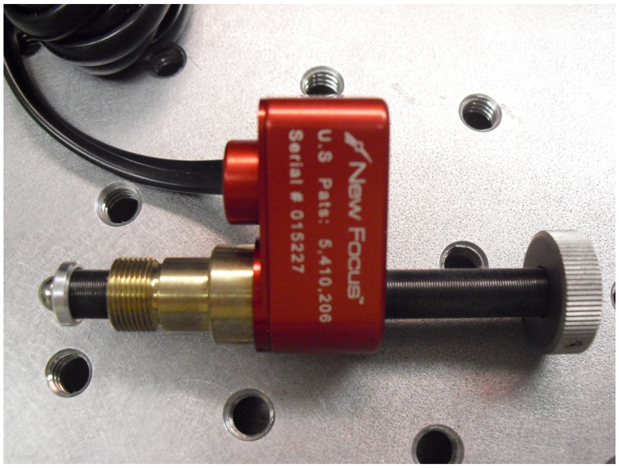


Figure 4.11: A New Focus Picomotor.

### 4.6.2 Alignment Cameras

Two IDS uEye cameras provide imaging capabilities for the primary alignment monitoring of our system. These cameras will provide images from our shear and tilt alignment monitoring subsystems. Figure 4.12 shows what these cameras look like. We communicate with these cameras through a USB connection. These cameras provide 752 x 480 pixels, each 6 $\mu$m square, providing us with the necessary resolution to measure alignment positions accurately. In addition, they provide a large exposure time range (80 $\mu$s to 5.5 s), providing us with the ability to monitor the system at both high and low flux levels. Section 4.7 will discuss these cameras and the alignment offset measurement in more depth.

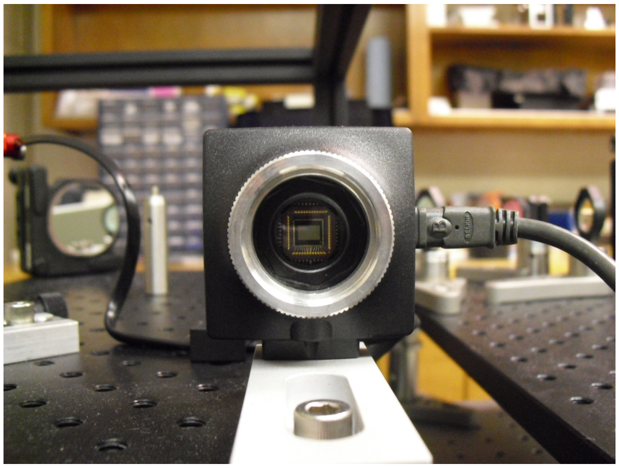
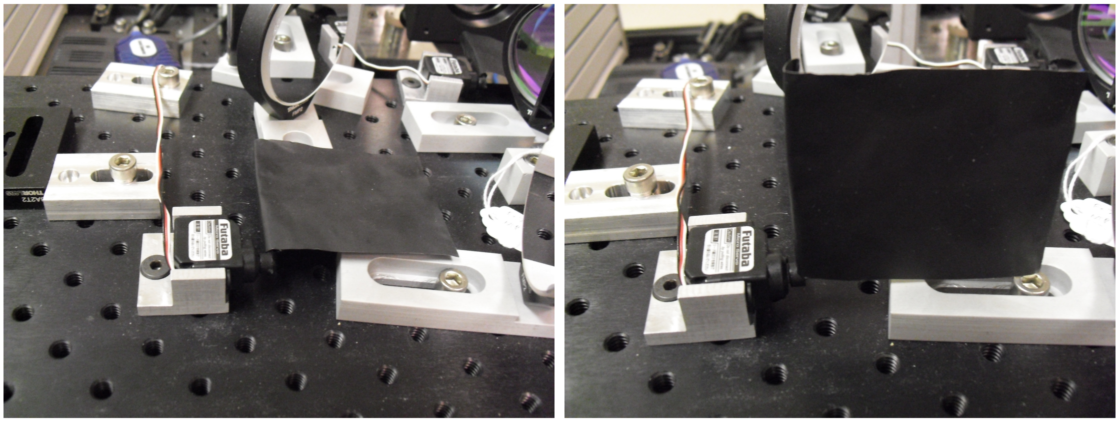


Figure 4.12: An IDS uEye USB detector.

### 4.6.3 Servo Blockers

The two servo blockers in our system allow us to isolate the light returning from the reference and sample arms. This allows us to inspect the signal from each arm individually. Using these, we can determine where the light on our detector is coming from (either the reference arm, sample arm, or some background signal). Our system communicates with the servo blocker controller using a serial port connection. While low precision, these servos provide sufficient repeatability for beam blocking and are inexpensive devices. Figure 4.13 shows what these servo blockers look like.



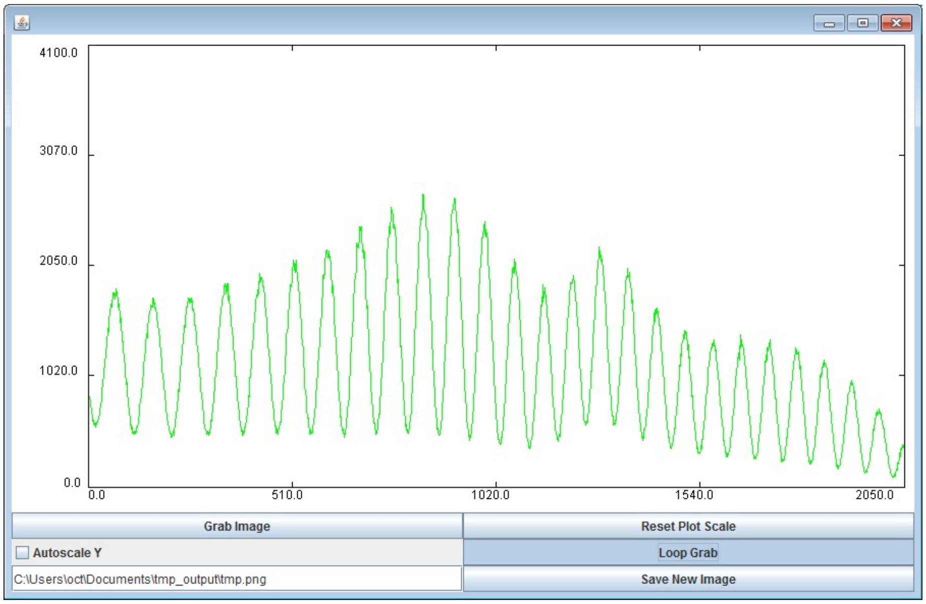
(a) Blocker Down

(b) Blocker Up

Figure 4.13: The servo blockers in our system allowing us to isolate the signal returning from our reference arm, sample arm, and any background light.

### 4.6.4 Linear Detector

The Basler Sprint linear detector in our system captures our final data product. Ensuring we have strong, useful, and representative signal on this detector is the end goal of our OCT and alignment systems. While our alignment cameras will be primarily responsible for maintaining alignment, information from this detector can confirm the system is operating as expected and can be used to aid in initial system setup or to recalibrate the alignment system when large fluctuations have occured. Figure 4.14 shows what our OCT signal looks like on this detector.



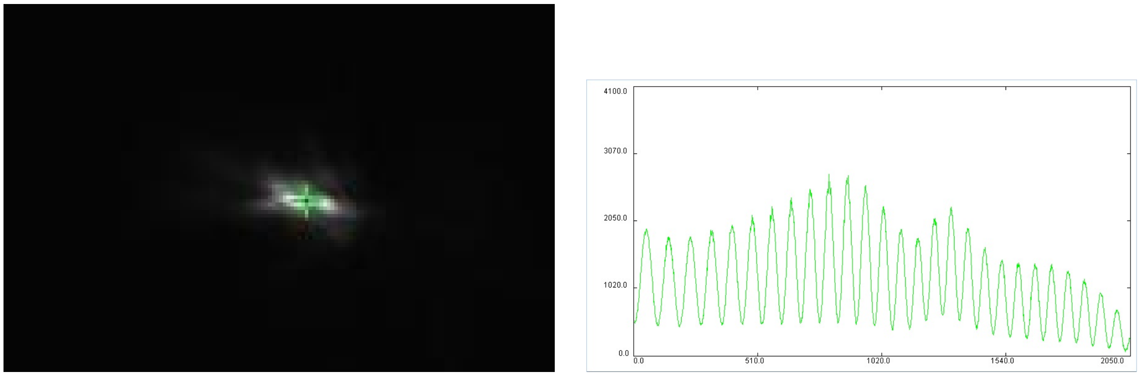
☐ Autoscale Y

C:\Users\oct\Documents\tmp_output\tmp.png

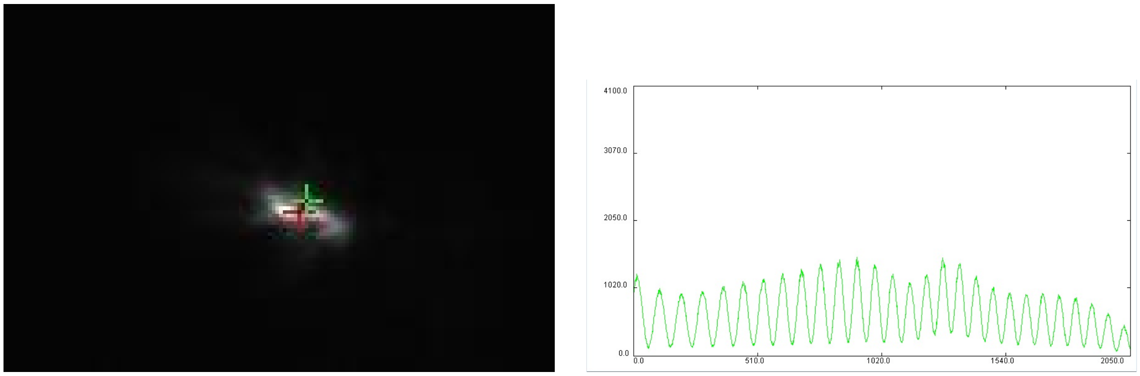
Figure 4.14: A simple interface allowing us to monitor the signal reaching the final detector in our OCT system. It currently shows an OCT spectral interference signal hitting our detector.

## 4.7 Software Offset Measurement

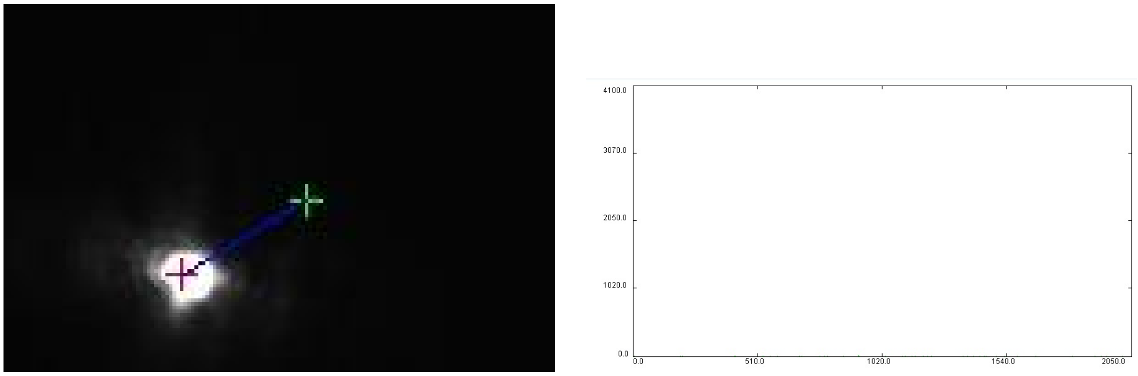
One of the main requirements of our system is the ability to determine alignment errors. These errors show up as offsets from the expected positions of our beams on our alignment cameras. By quantifying these offsets, we can determine the required corrections necessary to return to optimal alignment. Figures 4.15 and 4.16 show our ability to detect alignment offsets and the effect they have on our final signal.



(a) Aligned

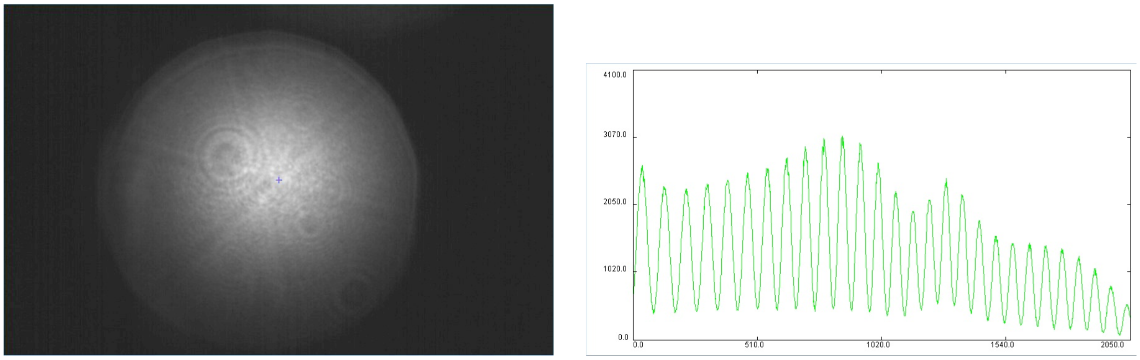


(b) Slightly Misaligned

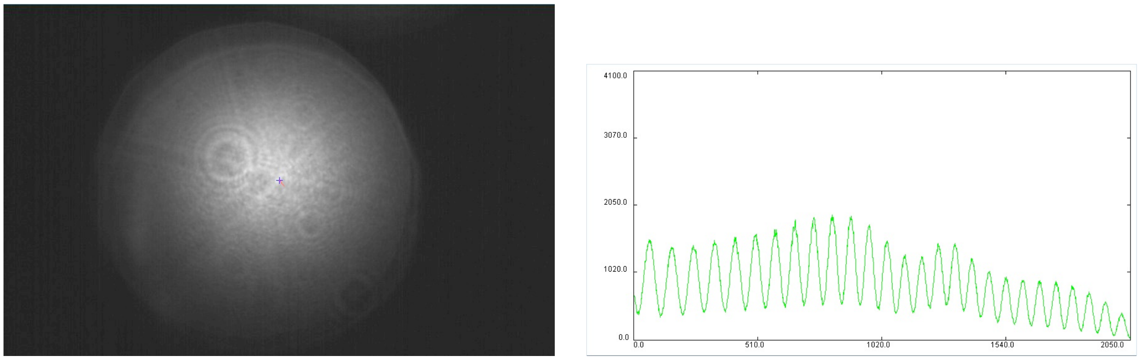


(c) Misaligned

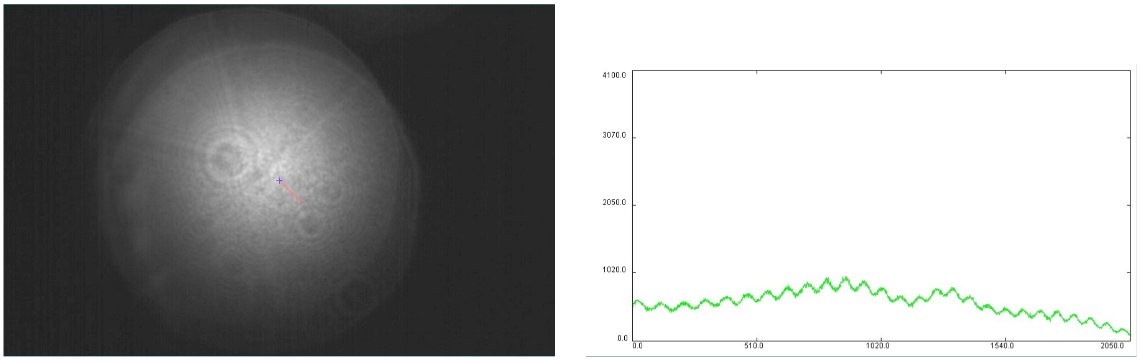
Figure 4.15: These images show the ability to measure tilt misalignments using our system. The images on the left show the measured offset of our tilt while the plots on the right show the signal hitting our detector. The green cross indicates where the spot should be while the red cross centroids the actual spot. The images on the left are zoomed in views of our tilt sensor and do not show the full field of view.



(a) Aligned



(b) Slightly Misaligned



(c) Misaligned

Figure 4.16: These images show the ability to measure shear misalignments using our system. The images on the left show the measured offset of our shear while the plots on the right show the signal hitting our detector. The red line coming out of the blue cross in the center of our images indicates the direction and magnitude of misalignment. Note that this axis of control only affects the reference arm of our interferometer and so the signal from the sample arm is always present at the same intensity in our plots. The images have been enhanced to increase contrast when printing.

### 4.7.1 Tilt Offset Measurement

The system tilt manifests as a positional offset of the focused spot on the pinhole plane. An offset of this spot from the pinhole has two main effects: the centroid of the reflected light off the pinhole plane shifts and the intensity of the reflected light increases (due to less light passing through the pinhole). Our goal is to determine the required correction to correct for any tilt offset induced in our beam.

If we have a perfectly focused spot reimaging our input fiber on the pinhole, it will resemble an Airy disk, the diffraction pattern caused by our finite aperture optics. It will have a very bright core (the signal we want passing through the pinhole) along with much dimmer rings. If further imperfections from a diffraction limited spot occur, they will pull light from the core into the wings—the light outside the core is the light we wish to block with our pinhole. The core of the Airy pattern contains approximately 84% of the light with the first ring containing approximately 7% and the third ring containing approximately 3%—even in the ideal case, a significant fraction of our light will hit the pinhole and provide us with useful signal to monitor alignment. Despite this, the required dynamic range for monitoring the entire Airy pattern is large—the peak intensity of the first ring is less than 2% of the peak intensity of the central core. Because our detector has only 8 bits of discrimination (256 levels), obtaining decent contrast on the rings will cause saturation in the core if it fails to pass through the pinhole.

Assuming the system begins in an aligned state, we wish to maintain the position of the focused spot on the pinhole plane. We need to be able to identify the desired position and maintain that position. To do this, we need to determine an appropriate direction and magnitude of corrective motion for any offset. With a fixed sample in the system, the pattern of light on the pinhole plane stays constant. Changing the tilt of the system shifts this pattern in a deterministic direction. The centroid of this pattern provides a good indicator of the offset from the desired position.

In calculating the centroid, two different methods can be useful. When a bright clean spot illuminates the pinhole (such as with the reflection off a mirror in the sample arm, see figure 4.2(a)), weighting the centroid by the intensity of the pixel value enhances the accuracy by accounting for the brighter center of the spot. When a more irregular sample is placed in the sample arm (providing a reimaged spot similar to that in figure 4.2(d)), intensity weighting can greatly skew the centroid location. Simply thresholding the image and centroiding the thresholded pixels without weighting provides a superior response in this case—the reduced information per pixel is offset by a larger number of illuminated pixels. Appendix A.1.1 provides the code used to centroid our images.

Despite the potential for saturation when the core misses the pinhole, the exposure time should be set to properly image the position when the light passes through the pinhole. The IDS cameras still behave well when saturated by the core, allowing accurate enough measurements to move the core into the pinhole. As the core moves into the pinhole, the light diminishes and ends the saturation and we are able to still measure the correct offset. If the exposure time is set to properly image the core, the signal will be too dim for proper measurement when the core enters the pinhole. An adaptive exposure time method could also work but is unecessary. Appendix A.1.3 provides code that measures the tilt offset.

### 4.7.2 Shear Offset Measurement

The shear offset measurement system images the collimated beams in our system. We need to ensure that both beams in the system go through the system together and pass through the focusing lens to land on the pinhole.

Identifying the two separate beams can be easily, but invasively, performed by using the beam blockers. Fortunately, the pupils do not change significantly with small shears. By storing the individual pupil images, we can compare shifted summations to a combined image to extract the position of each pupil. The required shift to generate the combined image gives us the offset of the pupil from the original position. Appendix A.1.4 provides code that measures the shear offset.

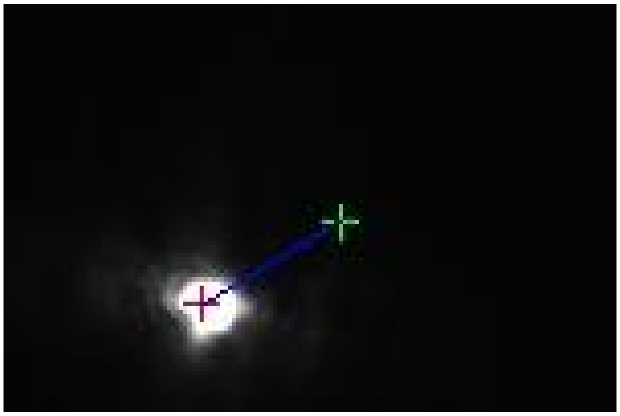
## 4.8 Computer Controlled Alignment

By combining computer control of the hardware described in section 4.6 with the offset measurements described in section 4.7, we have sufficient access to monitor and control the alignment of our system entirely through a computer. Figures 4.17 and 4.18 give a progression of images showing improving alignment obtained entirely with computer control but using human interaction to make alignment decisions.

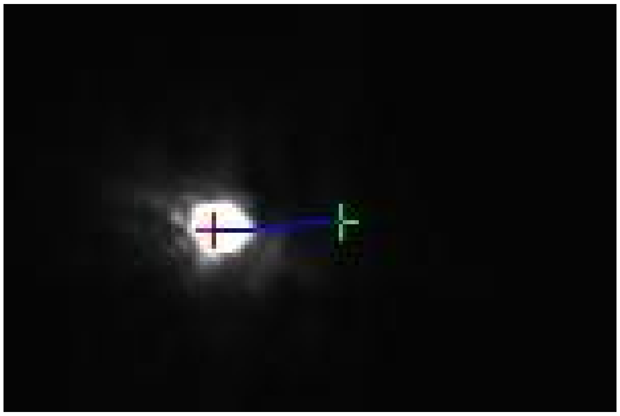
## 4.9 Optimizing Alignment

Our offset measurement algorithms above begin by assuming a good alignment point is known and maintaining that alignment. While this will be true when the system is first constructed, the alignment system will undergo similar misalignments to the main system—we need some method of determining a good alignment position.

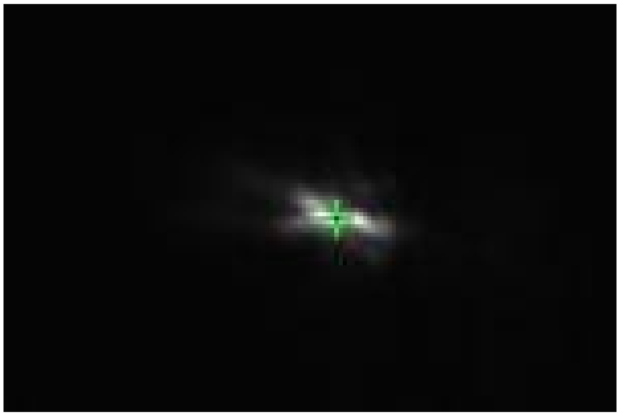
As we are primarily concerned with obtaining good signal on our final detector, this detector can be used to assess the alignment. One limitation of this is that the alignment must already provide sufficient light to this detector—the light must already be at least partially passing through the pinhole. The large field of view of our alignment cameras



(a) Misaligned

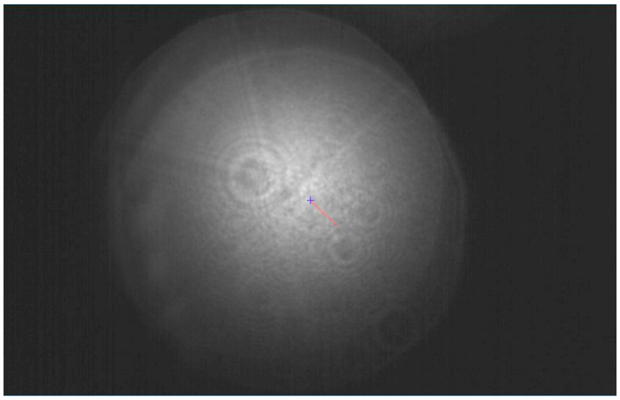


(b) Axis 1 Corrected

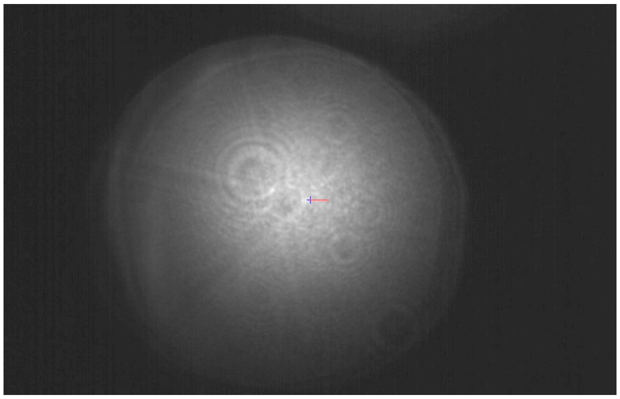


(c) Aligned

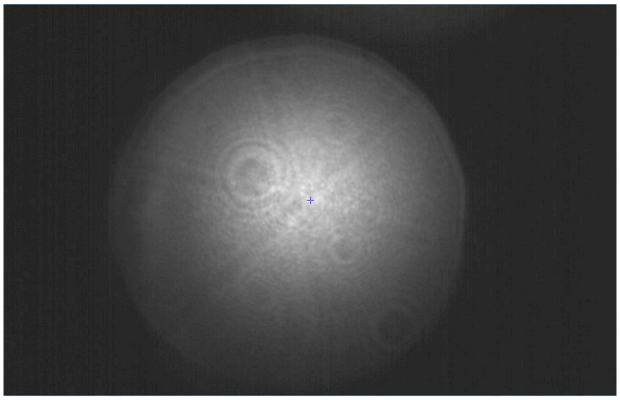
Figure 4.17: These images catalog a computer controlled (with human decision making) tilt realignment. We first improved the alignment on one axis and then completed the alignment on the second axis. The final realigned system behaves similar to the “aligned” configuration in figure 4.15.



(a) Misaligned



(b) Axis 1 Corrected



(c) Aligned

Figure 4.18: These images catalog a computer controlled (with human decision making) shear realignment. We first improved the alignment on one axis and then completed the alignment on the second axis. The final realigned system behaves similar to the “aligned” configuration in figure 4.16. The images have been enhanced to increase contrast when printing.

allows us to sufficiently align the system for signal to reach the final camera even if corrections are needed for better alignment. Once we have signal on the final camera, we can use this signal to improve our alignment and calibrate out any accrued alignment system errors.

By focusing on a mirror in the sample arm, we obtain a focused spot resembling an Airy pattern—we have a very bright core with fading light as we move farther from the center (see figure 4.2(a)). Blocking the reference arm with the beam blocker and adjusting the tilt motors, we are able to adjust the amount of light returning from the sample arm mirror that makes it through the pinhole without interference effects caused by the reference arm. Because the spot has a smooth profile, a simple gradient following algorithm with reducing step sizes is sufficient to maximize our sample signal.

Once the sample arm is maximized, we can adjust the shear control to optimize our reference arm as well. To avoid interference effects affecting the measured signal, we can block the sample arm. Again, a simple gradient following algorithm with reducing step sizes is sufficient to maximize our reference signal.

In a similar fashion, we can adjust the vertical position of the light hitting our spectrograph. As this control affects both arms equally, we can use the signal from both arms and simply maximize the total throughput. The interference between the two arms should stay mostly constant at this point and so blocking one arm or the other is not necessary (although slight fringe washout can be seen when the mirror moves, this is a smaller effect than in the other two cases).

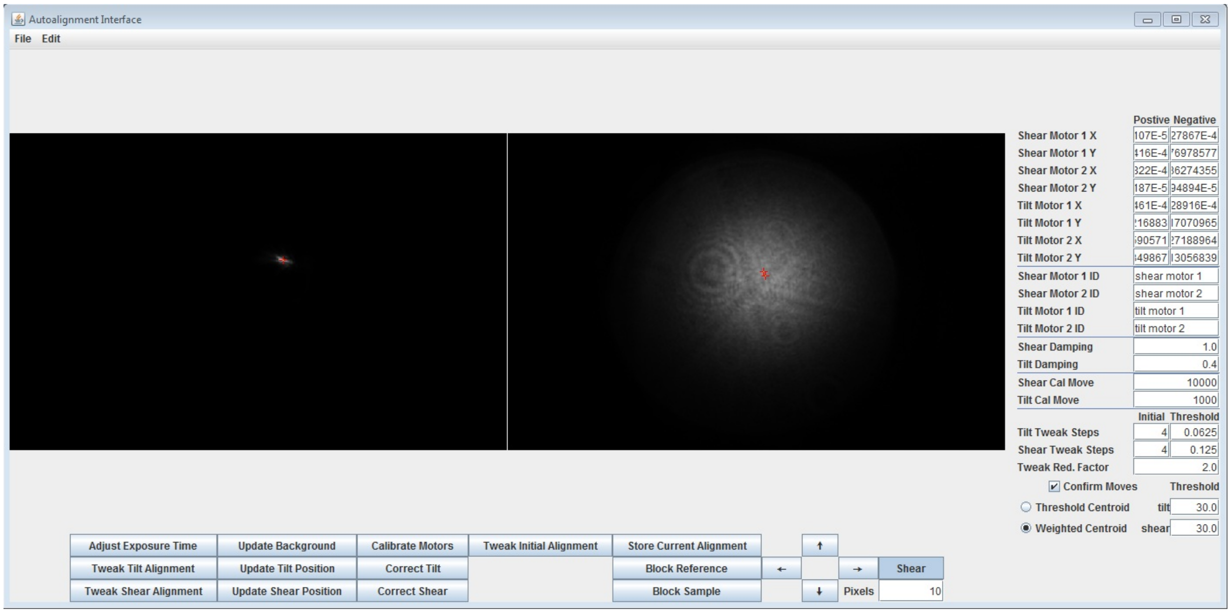
Appendix A.2.2 provides the code used to optimize our alignment.

## 4.10 Automatic Alignment Interface

Figure 4.19 presents the current automatic alignment interface. With a deterministic series of button presses, a user can optimize and maintain system alignment. This system currently requires human verification of the issued commands to prevent system damage during our initial testing. All alignment measurements and correction commands are determined and issued by the interface.

## 4.11 Automated Alignment Correction

The main system works by monitoring changes from an initial good alignment state. When the system is properly aligned, the system state is saved in a series of variables that allow us to determine offsets from this state. Even with large system changes (including alignment



| | Postive | Negative |
|---|---|---|
| Shear Motor 1 X | 107E-5 | 27867E-4 |
| Shear Motor 1 Y | 16E-4 | 6978577 |
| Shear Motor 2 X | 22E-4 | 6274355 |
| Shear Motor 2 Y | 187E-5 | 94894E-5 |
| Tilt Motor 1 X | 61E-4 | 28916E-4 |
| Tilt Motor 1 Y | 16883 | 7070965 |
| Tilt Motor 2 X | 90571 | 7188964 |
| Tilt Motor 2 Y | 49867 | 3056839 |

| | |
|---|---|
| Shear Motor 1 ID | shear motor 1 |
| Shear Motor 2 ID | shear motor 2 |
| Tilt Motor 1 ID | tilt motor 1 |
| Tilt Motor 2 ID | tilt motor 2 |
| Shear Damping | 1.0 |
| Tilt Damping | 0.4 |
| Shear Cal Move | 10000 |
| Tilt Cal Move | 1000 |

| | Initial | Threshold |
|---|---|---|
| Tilt Tweak Steps | 4 | 0.0625 |
| Shear Tweak Steps | 4 | 0.125 |
| Tweak Red. Factor | 2.0 | |

☑ Confirm Moves

| | | Threshold |
|---|---|---|
| ○ Threshold Centroid | tilt | 30.0 |
| ● Weighted Centroid | shear | 30.0 |

Pixels 10

Figure 4.19: The alignment interface. While the interface currently uses human verification to prevent system damage, the user only needs to press a deterministic series of buttons to align the system.

offsets that render the system completely unuseable), this allows us to quickly return close to our previous alignment.

Because of our reliance on an initial good state for system monitoring, we also need a method to optimize our initial state. Our interface is able to do this but requires access to primary system components (such as the final OCT detector) to accurately determine the best alignment state. This optimization is an intrusive process and also requires decent initial alignment to run—it is not suitable for constant system monitoring but gives us a good initial state and can correct for errors accruing in our alignment system. Combined with the primary alignment scheme, we are able to maintain high quality short and long term system alignment.

Given the calculated offsets, we must correct the errors by moving our Picomotors. Before we can determine the appropriate corrections, we must calibrate our system to calculate the relationship between camera offsets and Picomotor movements. This calibration is performed automatically by our software interface but requires operation of the system to be suspended. By moving each axis of the system individually by a known amount and computing the apparent movement, we can determine the effect each axis has on our system. It is worth noting that the mount loading forces cause forward and reverse picomotor movement commands to react differently, requiring a different calibration for each direction. Appendix A.2.1 provides the code used for motor calibration.

With the motor calibration and our measured offsets, we can calculate appropriate Picomotor commands to improve our current system alignment. By iteratively measuring the offset and correcting the offset, we have a feedback loop to maintain alignment. Using a small amount of damping will increase the response time slightly but allows for small errors or drifts in the motor calibration. Appendix A.2.3 provides the code used to convert pixel offsets to motor movements.

Because this system is still in testing mode, our alignment feedback loop currently requires human verification of each step. Using the above interface, a human controller commands the system to determine the alignment offset, determine the required correction, and verifies the correction is reasonable before allowing the system to implement the correction. By repeating this process for both tilt and shear, the system is able to recover and maintain system alignment. All the analysis and command decisions are determined and issued by the alignment software. Simply looping these commands without human verification is sufficient to correct alignment automatically. Figure 4.20 provides a flowchart showing the normal alignment procedure.

## 4.12 Alignment Limits

The alignment system is designed to correct for small and moderate alignment errors. This section quantifies the limitiations outside which the system will have problems.

The tilt sensor has a field of view of about 3000 x 2000 arcseconds. Our pinhole size is about 27.5 arcseconds in diameter with a diffraction limited spot size diameter of about 20.4 arcseconds. Our camera pixel size is about 4.1 arcseconds per pixel and our centroiding precision is better than 0.1 pixels (0.41 arcseconds). In the spot plane, a movement of 1 arcsecond corresponds to 0.36 $\mu$m. A 1°C temperature change near room temperature corresponds to about 0.88 $\mu$m (2.44 arcseconds) in a 1.5" (25.4 mm) tall aluminum mount. A 1" (25.4 mm) diameter mirror needs about 0.05 $\mu$m positional offset between opposite edges for a 1 arcsec tilt. We expect to see a significant performance degradation in our system with a several degree temperature change but we should be able to compensate for tilt errors even with much larger temperature changes (10's of degrees Celsius).

Our shear sensor has a field of view of about 37.6 x 24 mm. Our pupil diameter is about 21 mm. Each pixel in the field of view corresponds to about 50 $\mu$m. For efficiency, our pupil position measurement only determines offsets at the single pixel level, or 0.2% of the pupil diameter. This axis is less sensitive than the tilt axis, with a sizable portion of the error being directly due to tilt changes—a 10 arcsec beam tilt causes a 0.1 mm shift over a 1 m path. We can easily identify pupil shifts that stay fully on the camera ($\sim$1.5

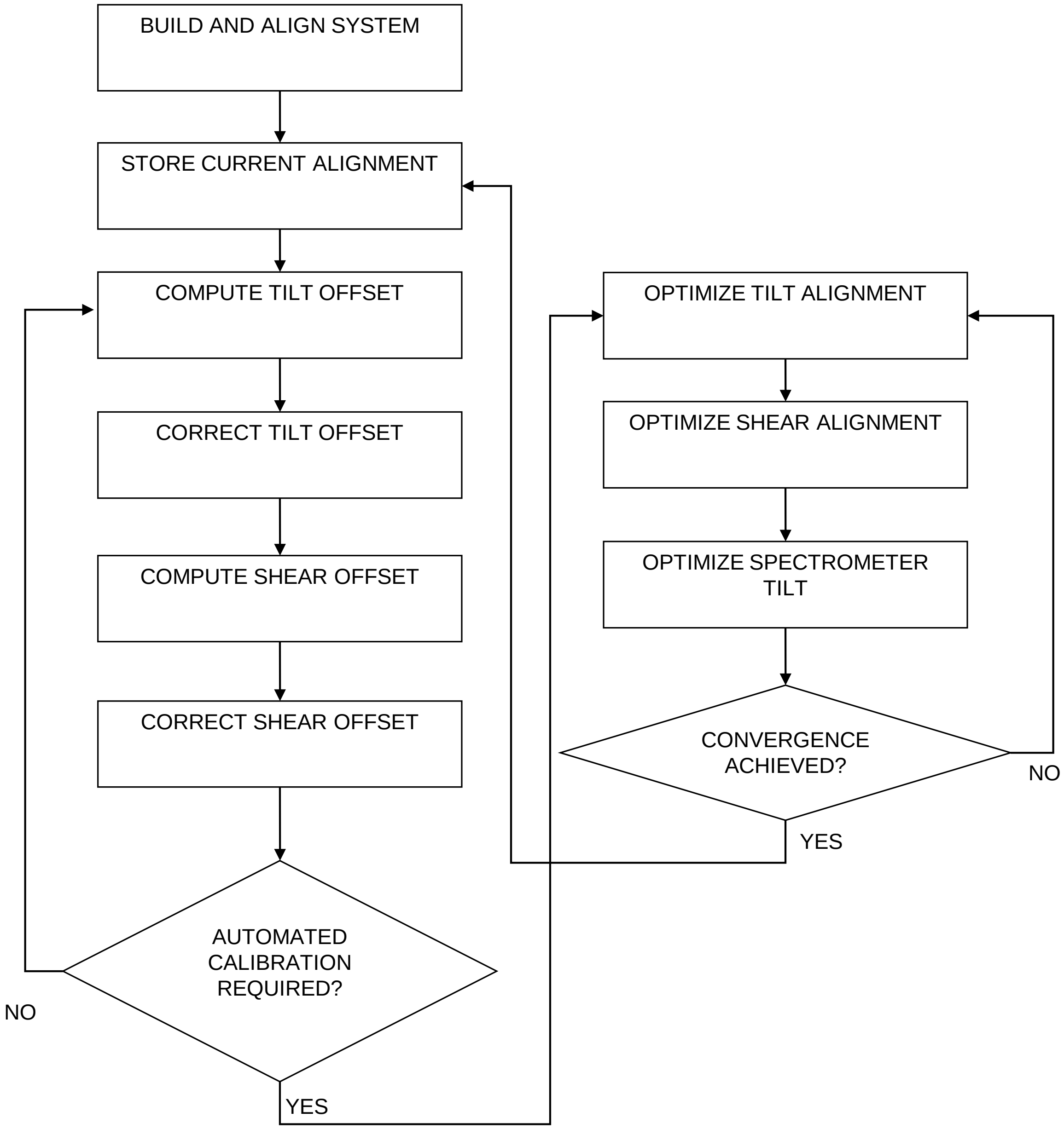


Figure 4.20: A flowchart describing our alignment procedure.

mm in the smallest direction) and can determine an appropriate direction of movement for significantly larger offsets—the pupils will still barely be on the detector for over 20 mm shifts. The limitations on this camera should not restrict the useable range of the system past what the tilt sensor requires.

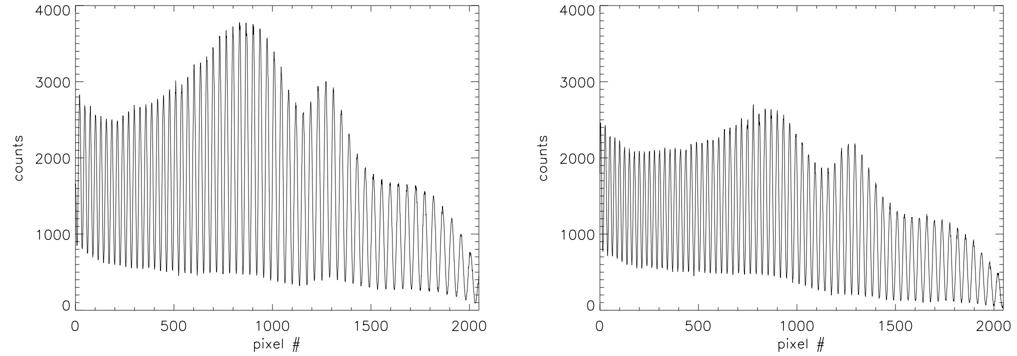
## 4.13 Temperature Testing

In order to show the viability of our system to correct alignment despite large fluctuations, we subjected our system to large temperature changes in the lab. We describe this experiment here, referencing Figure 4.21 to show the results. Figure 4.22 converts the original plots to show the intensity and fringe contrast (discussed in Section 4.1.1), which can be more easily related to system performance.

We performed this experiment by covering the OCT system with a cloth and using a small space heater near the OCT system to produce heat. We placed a mirror at the focus of the sample arm to provide a steady signal and aligned the system at 22° C. This gave us a strong signal on the detector with good fringe contrast (a).

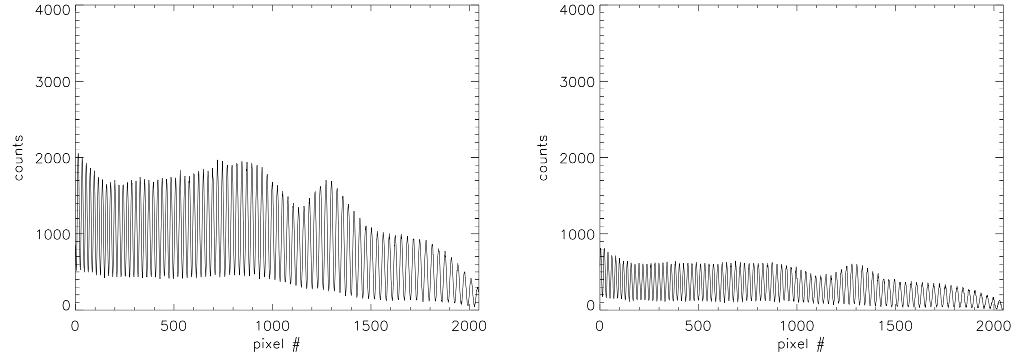
We slowly increased the temperature over 4.5 hours, recording the signal change seen on our detector as the temperature increased (b-d). With no alignment performed, a clear decrease in measured signal is seen. At our maximum temperature (32° C), over 80% of our signal has been lost (slightly under 20% remaining throughput).

With the temperature stabilized approximately 10° C above the initial temperature (at 32° C), we used the automatic alignment system to realign the system (e). Alone, this returned us to about 85% of the original throughput. By adjusting the focus of the sample arm (a standard part of the process of imaging a sample), this further improved to over 90% (f). The remaining loss might be explained by a slight reduction in reference arm throughput, comparable to that seen in the sample arm, combined with a slight shift in the focused position on the sample mirror—we use a slightly damaged mirror in our sample arm, because we handle this mirror often and wish to avoid accidentally damaging another, and a shift in position could easily cause a small decrease in reflected intensity. Also recall that the reference arm intensity is generally greatly reduced with ND—a slight adjustment in the ND value used will easily compensate for this loss. In addition, we notice that the fringe contrast has improved slightly–it's likely that the relative intensity between the sample and reference arms improved due to the non-uniform throughput loss (supporting a reduction in reference arm throughput). Both these effects would not decrease our actual signal in an experiment but could explain the losses seen here. Even if this loss carries over to a true signal loss, we have recovered the vast majority of our signal despite a very large temperature change from the initial setup of our interferometer—more normal changes should be even easier to correct.



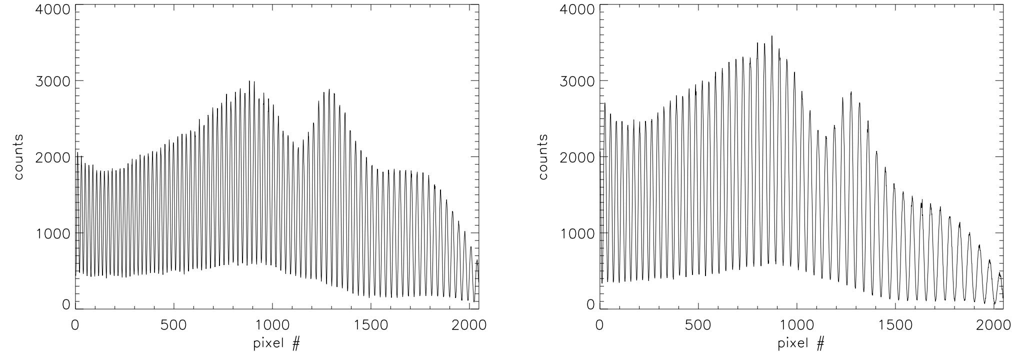
(a) 22° C

(b) 24° C



(c) 28° C

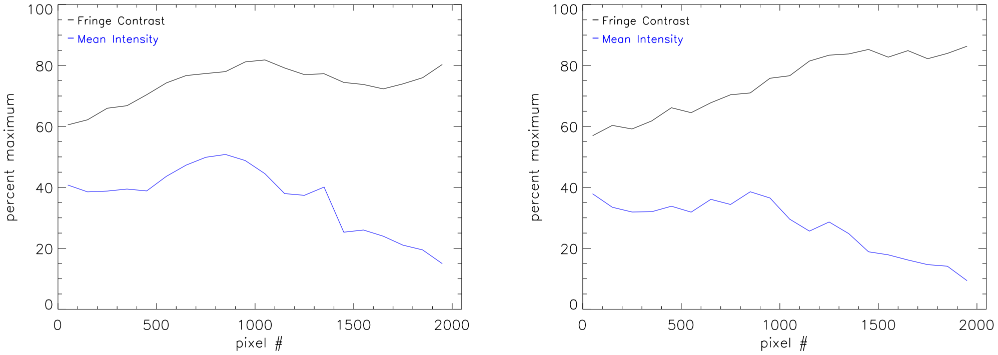
(d) 32° C



(e) 32° C Realigned

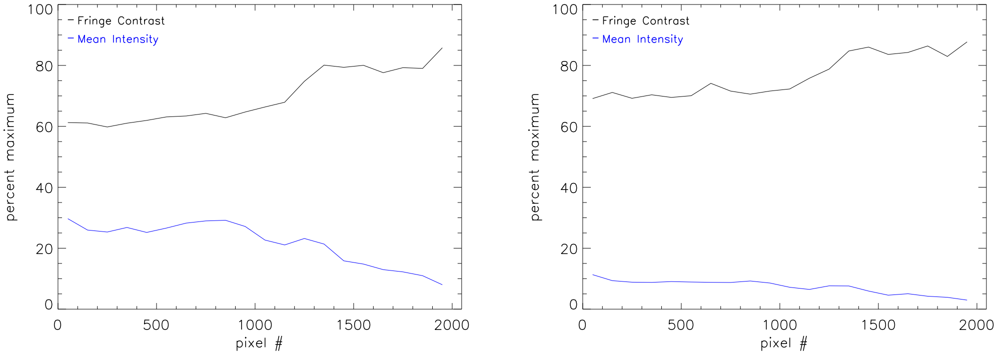
(f) 32° C Realigned & Refocused

Figure 4.21: These plots show the interferometric spectrometer signal from a mirror placed in our OCT system. The system was aligned at 22° C (a). As the temperature increases, a significant signal drop occurs (b-d). Automatic alignment greatly enhances the returned signal (e). A small sample focus adjustment after realignment returns us close to our original system throughput (f).



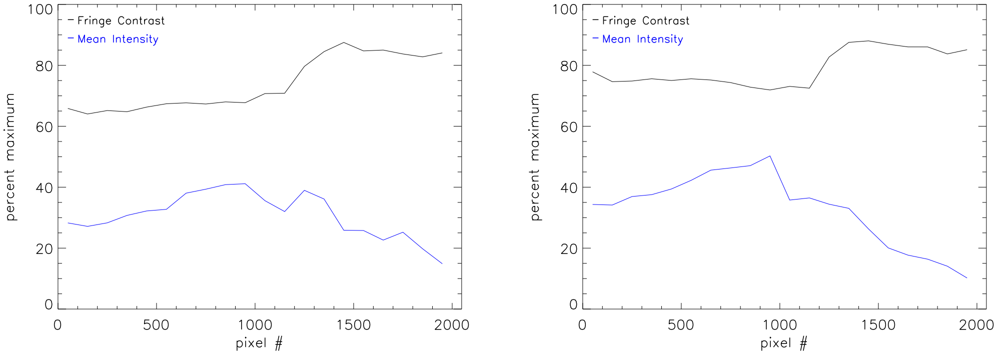
(a) 22° C

(b) 24° C



(c) 28° C

(d) 32° C



(e) 32° C Realigned

(f) 32° C Realigned & Refocused

Figure 4.22: These plots show the fringe contrast and mean intensity of our measured interferometric signals as a function of temperature (100 pixel bin sizes). The system was aligned at 22° C (a). As the temperature increases, a significant signal drop occurs (b-d). Automatic alignment greatly enhances the returned signal (e). A small sample focus adjustment after realignment returns us close to our original system throughput (f).

# Chapter 5

# Extension to Other Systems

The techniques used to enhance our OCT system can also be extended to other systems. By isolating the necessary degrees of freedom and implementing measurement and correction hardware, we can design an alignment system. This chapter goes through some of the processes involved and gives examples of how this can work for other systems.

## 5.1 Degree of Freedom Identification and Reduction

In section 4.2, we discuss the various degrees of freedom in our OCT system and how we can correct for errors induced by each of them. Here we expand on the actual process.

### 5.1.1 Identification

The first objective is to identify all the independent and significant degrees of freedom in our system. These are the degrees of freedom that will actually have some effect if changes occur. While the specifics will heavily depend on the actual system configuration, we provide some general guidelines here.

First, we need to know the characteristics of the light interacting with each optic. If the light is converging, diverging, collimated, or a focused spot, we will see different effects from different components. In our system, we primarily dealt with focused or collimated light but different systems can have different configurations.

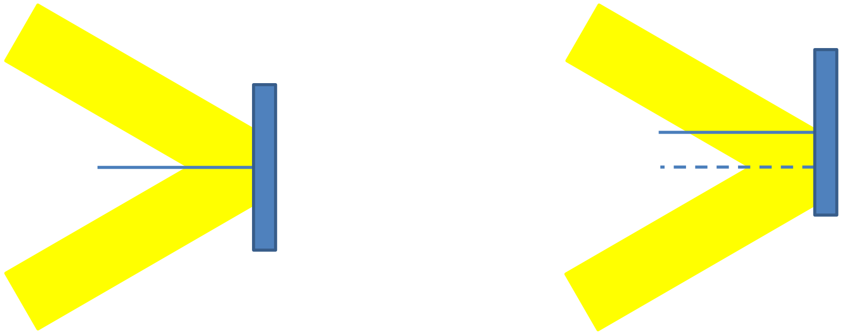
Next we need to look at the effect each individual component will have on the light path. Light hitting a flat mirror, a lens, a curved mirror, or other optical surfaces will all behave differently. The initial characteristics of the light at that surface will also matter. For systems with mostly simple surfaces (like flat mirrors, circularly symmetric lenses operating on collimated light, and similar), a geometric analysis is usually sufficient. When more complex optics are used, optical simulation software such as ZEMAX can be necessary (especially if the effect of one optic is expected to cause significant changes to the operation of another optic). We will briefly consider some specific examples in the next several paragraphs.

A flat mirror operating on collimated light is one of the simpler optics to deal with. Light reflecting off a flat mirror is reflected about the normal of the mirror surface. For collimated light, all the beams are travelling in the same direction and will see the same reflection. Four degrees of freedom (rotation about the normal, translation in two orthogonal dimensions perpendicular to the normal, and translation along the normal) have no effect on the direction of the normal—movement in these directions will not affect the reflection angle. The two remaining degrees of freedom cause a rotation of the normal, which leads to a different reflection angle of the beams. In addition, translation along the normal, while not affecting the direction of reflection, will change the incident point, potentially changing the path length and shear of the beam. With large movements, it is also possible for any degree of freedom other than rotation about the normal to cause the incident light to miss the mirror. Figure 5.1 illustrates these effects.

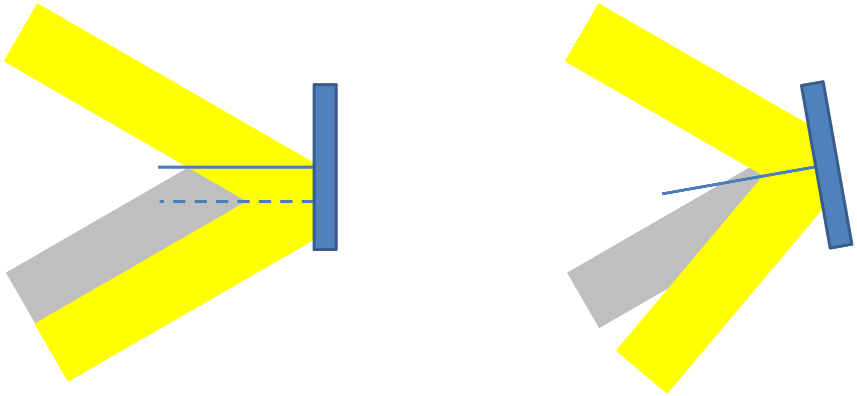
A standard lens converts a collimated beam into a focused spot over a specific focal length and vice versa. In the ideal case, tilts in the collimated beam are converted to positional shifts in the focal plane while shears simply tilt the cone angle. In reverse, a positional shift in the focused spot causes a tilt in the collimated beam while the incoming angle of the light from the spot determines the location of collimated beam (i.e., its shear). If the lens rotates about the optical axis, nothing changes. If the lens shears along the optical axis, the focal point of the lens shifts. If it shears perpendicular to this axis, the effect will vary depending on the direction of light propagation—if the lens is collimating light then a tilt will be seen in the collimated beam, while if the lens is focusing collimated light then the light will focus to a different point. If the lens tilts, this will rotate the focal plane and change the focused light position. As with the mirror, large enough shifts or rotations can cause the beam to completely miss the lens but this is an extreme case. Figure 5.2 illustrates these effects.

A single mode fiber can be approximated as a point source emitting light in a specified cone. If this light is to be collimated by a lens, the effects are related to those caused by a lens. If the position of the fiber changes on the focal plane of the lens, we will see a tilt in the collimated beam leaving the lens. If the exit of the fiber leaves the focal plane of the lens, we end up with a defocus. If the exit cone of light from the fiber tilts, we will see the collimated beam shear.

A corner-cube retroreflector consists of three reflective surfaces forming a shape similar to the corner of a room where ceiling or floor meets two side walls. This optical layout

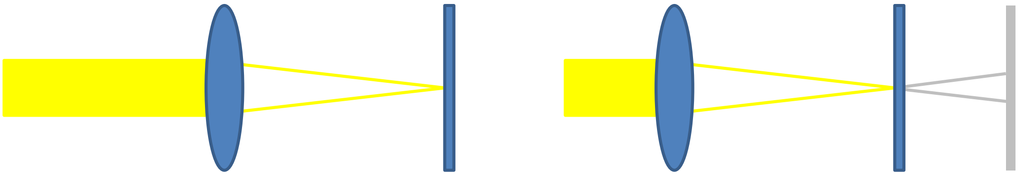


(a) Initial Configuration (b) Translation Perpendicular to Normal



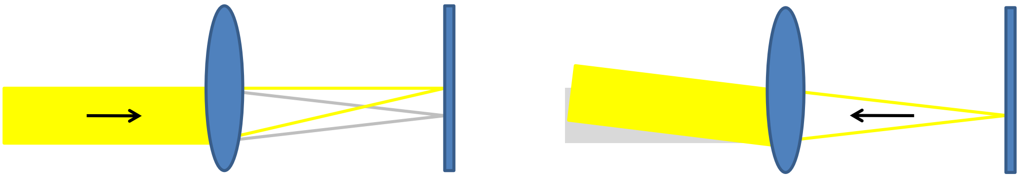
(c) Translation Along Normal (d) Tilt

Figure 5.1: The effect of mirror shifts on a collimated beam. (a) shows an assumed initial configuration while (b) through (d) show the effects of offsets from this configuation. Solid yellow indicates the collimated beam. The blue rectangle shows the mirror position and orientation while the solid blue line shows the mirror normal from the center of the mirror. If needed, a dotted blue line shows the mirror normal at the incident point. Where appropriate, equivalent objects in grey highlight differences from the initial configuration.



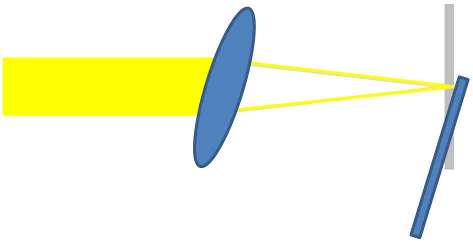
(a) Initial Configuration

(b) Translation Along Optical Axis



(c) Focusing Lens Lateral Shift

(d) Collimating Lens Lateral Shift



(e) Lens Tilt

Figure 5.2: The effect of various lens shifts. (a) shows an assumed initial configuration while (b) through (e) show the effects of offsets from this configuation. Solid yellow indicates collimated beams and yellow lines show focusing light. The blue oval shows the lens position and orientation while the blue rectangle shows the focal plane of the lens. Where appropriate, equivalent objects in grey highlight differences from the initial configuration.

has several nice properties, a primary one being strong tilt insensitivity—a beam entering the retroreflector exits with the same tilt as the incoming beam, as if bouncing off a flat mirror with a normal closely aligned to the optical axis. Unlike a flat mirror, though, any beam shear (or, equivalently, a shear in the retroreflector) is flipped about the center of the retroreflector. This effect has both advantages and disadvantages—while the sensitivity to shear can cause beam position errors, it can also be used to accurately cause an offset in beam position with no change in tilt.

Similar analyses can be performed for other optical components. After identifying all the possible degrees of freedom, we need to reduce the required number of controls to a more practical number. This is discussed in further depth in the following section.

### 5.1.2 Reduction

Once we have identified the effect of errors in each optic, we need to determine the required controls to correct for these errors.

First, we need to identify where any errors will cause noticeable problems. This might be the final detector (where an error changes the detected signal) or it could be something like a pinhole plane (where an error can cause the light to no longer propagate through the system). These are the positions where we will need to monitor the alignment to correct the errors before they cause a problem.

We then need to identify what components and subsystems contribute to detectable errors at this point. These are the places where corrections may need to be performed to correct the errors.

Once we have identified where errors can be corrected, we try to combine degrees of freedom into a smaller number of controls. For example, a tilt caused by a mirror can be corrected by a tilt in the beam hitting the mirror. This is true even for multiple mirrors in series, allowing a single tilt correction to handle many different tilt contributions.

It is also important to note at this point that, if the light returns along the same path it originally followed, many misalignments will correct themselves. This can be seen, for example, by considering a beam that reflects off the same mirror twice from opposite directions—when the beam first hits the mirror any error term is added in but, on the return trip, the reverse error is added (effectively subracting out the original error). By identifying locations where this occurs, significant reduction in the number of required control surfaces can be achieved.

Once all possible consolidations have been identified, we are left with a minimal number of necessary correction axes. We are now simply left with implementing monitoring and control hardware to measure and correct errors with these axes. While the implementation choice can vary for different system, hardware and algorithms similar to those described in chapter 4 are suitable for many different system configurations.

## 5.2 Simplified Example

We now will discuss a simplified example system (see figure 5.3) and show how these techniques can be applied to other systems.

Figure 5.3 shows an offset beam interferometer, such as might be used in a Fourier Transform Spectrometer (FTS) or optical metrology system. The offset layout allows easy access to the complementary outputs of the interferometer, collecting additional signal over a single cube design (like in our OCT system). Collimated light enters the interferometer (in this case, collimating the output of a fiber with a lens) and is split by beam splitter cube BSC1. The beam splitter cube acts as a mirror for half the light (sending light towards Retroreflector RR1) and passes the other half of the light directly through. Two corner-cube retroreflectors (RR1 and RR2) are used to offset the beams and return them to beam splitter cube BSC2. Half the light from RR1 passes through BSC2 and joins half the light from RR2 reflected from BSC2 to form Output A. The other half of light from RR1 reflects off BSC2 and joins with half the light from RR2 transmitting through BSC2 to form Output B. Complementary interference effects due to phase shifts caused by different path lengths for the two arms of the interferometer provide our signals in Output A and Output B.

Examining the system, we have 6 different optics that can have an effect on the alignment: the fiber, the collimating lens, two beam splitter cubes, and two retroreflectors. Treating the beam splitter cubes like mirrors for the reflective path and ignoring them for the transmissive path, we can use the descriptions in section 5.1.1 to determine the effect of the various optics on our system. In addition, it is easy to identify the primary alignment points as Output A and Output B.

Considering Output A, we primarily have a focus effect from the fiber/collimating lens pair, an overall tilt and shear from the same, a tilt in one beam from BSC1 and a tilt in the other beam from BSC2, and a shear in one beam from RR1 and the other from RR2. In Output B, we have a focus effect from the fiber/collimating lens pair, an overall tilt and shear from the same, a tilt in one beam from both BSC1 and BSC2, and a shear in one beam from RR1 and the other from RR2.

If analysis indicates that collimation is a significant factor, our system has only one place to affect the collimation and corrections here propagate through the rest of the system equally. For the rest of the system, we may need to worry about: (1) that the two beams at Output A have the same tilt, (2) the two beams at Output A have the same

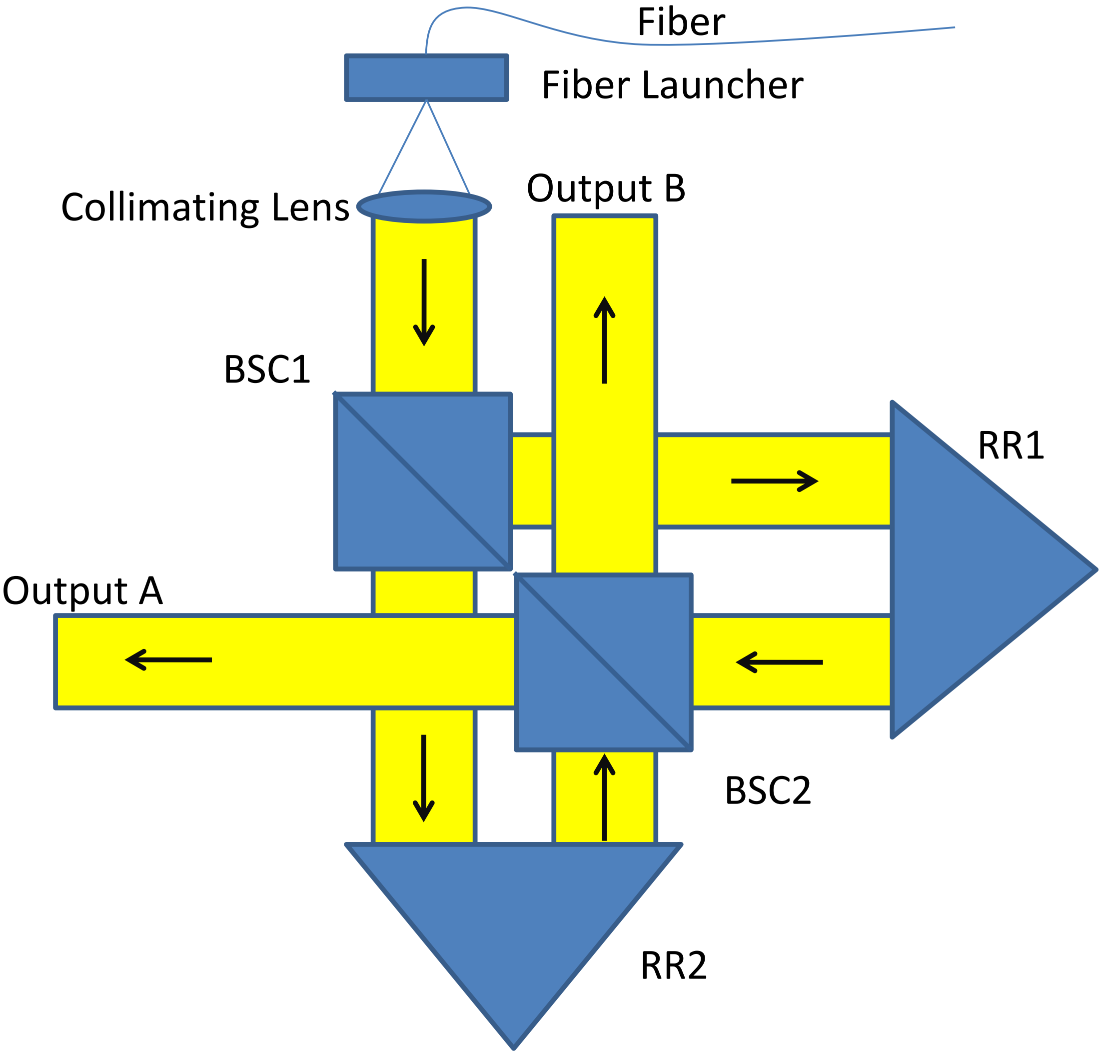


Figure 5.3: A simple optical layout to illustrate the techniques described in the text. BSC1 and BSC2 are beam splitter cubes while RR1 and RR2 are corner cube retroreflectors. Black arrows indicate the direction of light propagation.

shear, (3) the two beams at Output B have the same tilt, (4) the two beams at Output B have the same shear, (5) Outputs A and B have appropriate overall tilts, and (6) Outputs A and B have appropriate overall shears.

In order for (1) to hold, it is a requirement that BSC1 and BSC2 have the same tilt—if this is not the case, the beams reflecting from BSC1 and BSC2 would each have a different induced tilt after having started with the same tilt before entering BSC1. The use of a single large beam splitter cube can mitigate this effect, although this can increase the amount of dispersive and absorptive glass in the system and does not allow for corrections of any imperfections in the retroreflectors or splitting surface. Motorizing the tip and tilt of one cube provides the necessary alignment freedom to maintain this axis. A tilt monitoring system (similar to the one used to monitor pinhole alignment in the OCT system) can provide the necessary feedback for this axis.

For (2) to hold, we need our beams to be coincident at BSC2 (if the two beams are coincident and have the same tilt, they will stay coincident as they travel further). Motorizing either RR1 or RR2 can correct for any relative offset in the two beams and a shear monitoring system similar to that in the OCT system can measure this offset.

As luck would have it, (3) holds automatically if (1) holds. The beam that passes through both beam splitters (or one beam splitter twice) sees no tilt and the beam reflecting twice will cancel out any tilt on the second reflection. Similar is true between (2) and (4)—if the beams are coincident with the same tilt at the beam splitter, they will follow the same path leaving in both directions. This allows the same hardware to ensure that both arms overlap in both outputs.

While we now can ensure that both arms of the interferometer will be overlapping at both outputs, the overall tilt (5) and shear (6) of these outputs may not be appropriate. Overall tilt can be easily added using the fiber position or collimating lens position, but these will adjust the two outputs simultaneously in opposite directions (because of the number of reflections seen by the two outputs). Also motorizing the tilt of the beam splitters allows for individual adjustment of the tilt of Output A—if we correct Output B using the fiber position and then adjust the overall beam splitter tilt to correct Output A, we can ensure both outputs have their own correct tilt. Adjusting both retroreflectors allows for a shear correction, again simultaneously adjusting both outputs in opposite directions. A shear of a beam splitter cube can allow separation of the control of the two outputs horizontally but not vertically. If additional optical surfaces are acceptable (or already present) and the outputs need to be adjusted individually, a motorized fold mirror (for tilt only) or dogleg (for tilt and shear) can be placed after beam recombination (for one or both arms). As above, similar tilt and shear sensors to those used in the OCT system can monitor these parameters for similar other systems.

As even this simplified example shows, the requirements and design for an alignment system will depend heavily on the requirements of the overall system. This process is similar to the design of any other control system—we extend this process to interferometric devices similar to our OCT system and suggest specific components and setups that can be useful for this.

# Chapter 6

# Conclusions

In chapter 3, we described the development and construction of a free-space OCT testbed system. This is a high-resolution OCT imaging system designed for imaging excised tissue samples at the structural and cellular level. The free-spaced based design gives us the flexibility to easily implement and test enhancements and minimizes the use of fiber (decreasing dispersion, allowing broader bandwidths, increasing throughput, and more—see section 1.1.1 for more detail). While modifications, enhancements, and additional testing will continue on the system, we have obtained quality images of different tissue samples. In addition, we present results that indicate that our system may achieve significantly greater penetration depths in tissue than standard OCT systems (section 3.3).

We also describe the development of an automatic alignment system for our OCT testbed. This enhancement is designed to compensate for the additional complexity involved in maintaining alignment of a free-space interferometer over the standard fiber based approach in settings outside tightly controlled laboratory environments (see section 1.1.2). This system is designed to compensate for the small misalignments that can be caused by shock events and temperature drifts, which can greatly affect the efficiency of our system. We describe the requirements, implementation, and temperature drift results, showing the effectiveness of this development in chapter 4.

## 6.1 Contributions

In this work, I present my significant contributions to the fields of interferometry and optical coherence tomography, including:

- A free-space OCT system designed for pathology that shows the potential for increased depth penetration through animal tissue and allows for additional enhancements to be easily implemented.

- An alignment system capable of maintaining the performance of this free-space OCT system outside of highly controlled laboratory environments through computer control.
- Methods for automating the alignment of said system, removing the need for specialist knowledge to correct alignment errors.
- Methods for extending the automatic alignment techniques to other interferometric technologies.

This work was presented at the SPIE Photonics West conference [7] and is the basis for US Provisional Patent Application no. 61/414,044 [8].

## 6.2 Future Work

Significant future work is already planned for both our free-space OCT design and for our automatic alignment system. A collaboration with the Thunder Bay Regional Research Institute (TBRRI) and the Thunder Bay Regional Health Sciences Centre (TBRHSC) will provide an excellent environment to develop clinical applications of our OCT system along with testing the reliability of our automatic alignment system in a real world environment. Work is also proceeding on alternative sample scanning geometries to further expand the useful regimes for our free-space interferometer design. In addition, several enhancements for our testbed system are planned, such as an enhanced spectrometer design, leveraging the flexibility of the free-space layout to easily upgrade the capabilities of our system.

In addition, we can extend our optical alignment enhancements to other systems. By examining the degrees of freedom in different interferometer configurations (such as done for our OCT system in section 4.2), we can identify the points where alignment drifts will cause significant errors and identify the necessary monitoring and control components to correct these errors. Chapter 5 expounds on this type of extension.

## 6.3 Final Words

As a whole, this work shows the development of an automatically-aligning free-space high-resolution optical coherence tomography system. This system allows us to bring enhancements typically only available in research laboratories to places, such as hospitals, where significant real-world benefits can be seen. In addition, we provide a process that can aid in developing similar enhancements for related devices.

# APPENDICES

# Appendix A

# Code

## A.1 Measurement Code

The code in this section is used to measure offsets for our alignment system. While the code has been simplified for clarity and brevity (for example, removing interface specific functions and hardware error monitoring code), the core algorithms should be apparent.

### A.1.1 Centroiding Code

Several of our other algorithms rely on locating the positions of spots and circles in the focal plane. We use the following centroiding algorithms to obtain these positions. The choice between the two centroiding algorithms given depends upon the target being imaged–for clean, sharply defined spots the weighted centroiding provides higher accuracy while the threshold centroiding performs better with large, diffuse, speckly returns from highly scattering samples.

#### Threshold Centroiding

This centroiding method computes the average position of all pixels above a specified threshold. This method is most useful when we expect a large spread of returning light without a clearly focused spot profile, such as when imaging a highly scattering sample.

```
/** Return the average position of the points above a given threshold in
 * an image.  If all points are below the threshold, return [-1, -1].
 *
```

```
 * @param img The image to centroid.
 * @param thresh The threshold.
 * @return The average position of pixels above the threshold.
 */
public Point2D thresholdCentroid(BufferedImage img, int thresh)
{
    //initialize centroiding variables
    double cenx = 0;
    double ceny = 0;
    int counted = 0;

    //determine the image size
    int width = img.getWidth();
    int height = img.getHeight();

    //extract the image data into an array
    int imgdata[] = img.getData().getSamples(0, 0, width, height, 0,
                                              (int[])null);

    //loop over the pixels in the image
    for(int i=0; i < height; i++)
    {
        for(int j=0; j < width; j++)
        {
            //if a pixel is above the threshold
            if(imgdata[i*width + j] >= thresh)
            {
                //increment the pixel count
                counted++;
                //add the pixel position to the averaging variables
                cenx += j;
                ceny += i;
            }
        }
    }
    //if we found any pixels above the threshold
    if(counted > 0)
    {
        //convert the sums to an average position
        cenx /= counted;
```

```
        ceny /= counted;
    }
    //otherwise, return the error condition
    else
    {
        //if no points were above the threshold, return [-1,-1]
        cenx = -1;
        ceny = -1;
    }
    if(debug)
    {
        System.out.println(cenx+" "+ceny);
    }
    return new Point2D.Double(cenx, ceny);
}
```

### Weighted Centroiding

This centroiding method weights the centroided pixels by their intensity. Focused spots should have more intensity near the center of the spot and this accommodates that in the position measurement. This method also allows an offset term to ignore background noise or correct for a negative bias.

```
/** Return the average position of the points in an image, weighted by
 * their intensity.  Allows an offset that values are shifted by to
 * affect weighting (values reduced below 0 become 0).  Return [-1,-1]
 * if all pixels are 0.
 *
 * @param img The image to centroid.
 * @param offset The shift to apply to pixel values.
 * @return The average position of pixels weighted by their intensity.
 */
public Point2D weightedCentroid(BufferedImage img, int offset)
{
    //initialize centroiding variables
    double cenx = 0;
    double ceny = 0;
    long weight = 0;

    //determine the image size
```

```
int width = img.getWidth();
int height = img.getHeight();

//extract the image data into an array
int imgdata[] = img.getData().getSamples(0, 0, width, height, 0,
                                         (int[])null);
//loop over the pixels in the image
for(int i=0; i < height; i++)
{
    for(int j=0; j < width; j++)
    {
        //remove the requested offset from the image
        int val = imgdata[i*width + j] - offset;
        //require pixel values to be positive (no negative photons)
        val = (val > 0)?val:0;

        //sum the total image counts used for centroiding
        weight += val;

        //sum the pixel positions appropriately weighted
        cenx += j*val;
        ceny += i*val;
    }
}
//if we found useful pixels
if(weight > 0)
{
    //convert the weighted sum to a weighted average
    cenx /= weight;
    ceny /= weight;
}
//otherwise, return the error condition
else
{
    //if the entire image has an intensity of 0, return [-1,-1]
    cenx = -1;
    ceny = -1;
}
if(debug)
{
```

```
        System.out.println(cenx+" "+ceny);
    }
    return new Point2D.Double(cenx, ceny);
}
```

### A.1.2 Save Current Alignment Code

This section of code stores the current system alignment to allow the system to maintain the current alignment configuration. While best performed with good alignment, the system is designed to allow maintainence of any desired alignment configuration. To this end, additional code to obtain a good initial alignment state is included below. Note that error handling and interface specific code has been trimmed for brevity.

```
/** The store current alignment button was pressed.  Store the necessary
 * variables to maintain the current alignment.
 *
 */
private void storeCurrentAlignmentButtonActionPerformed()
{
    //store the configured threshold for tilt alignment
    int tiltCentroidThresh;
    try
    {
        //multiply by 256 to convert to 16 bit format of averaged images
        //instead of 8 bit original image format
        tiltCentroidThresh = Math.round(Float.parseFloat(
                                tiltThreshField.getText())*256);
    }//handle conversion exception here

    //store the configured threshold for shear alignment
    int shearCentroidThresh;
    try
    {
        //multiply by 256 to convert to 16 bit format of averaged images
        //instead of 8 bit original image format
        shearCentroidThresh = Math.round(Float.parseFloat(
                                shearThreshField.getText())*256);
    }//handle conversion exception here
```

```
//store tilt alignment
try
{
    if(tiltcam == null)
    {
        throw new AlignmentException("Null tilt camera.");
    }
    //obtain an averaged image
    tiltBase = tiltcam.getAverageImage(ntilt);
    //subtract the background data
    tiltBaseBack = subtractImages(tiltBase, tiltback);
    //compute the current spot center
    tiltCenter = centroid(tiltBaseBack, tiltCentroidThresh);
}//handle camera errors here

//store shear alignment
try
{
    if(shearcam == null)
    {
        throw new AlignmentException("Null shear camera.");
    }
    //obtain an image of the reference arm
    blockSampleArm();
    //obtain an averaged image
    referenceArmImage = shearcam.getAverageImage(nshear);
    //subtract the background data
    referenceArmImageBack = subtractImages(referenceArmImage,
                                           shearback);
    //compute the current pupil center
    referenceArmCenter = centroid(referenceArmImageBack,
                                  shearCentroidThresh);
    unblockSampleArm();

    //obtain an image of the sample arm
    blockReferenceArm();
    //obtain an averaged image
    sampleArmImage = shearcam.getAverageImage(nshear);
    //subtract the background data
    sampleArmImageBack = subtractImages(sampleArmImage, shearback);
```

```
        //compute the current pupil center
        sampleArmCenter = centroid(sampleArmImageBack,
                                   shearCentroidThresh);
        unblockReferenceArm();
    }//handle camera errors here

    //ensure that the reference and sample arms are not blocked.
    unblockReferenceArm();
    unblockSampleArm();
}
```

### A.1.3 Tilt Alignment Offset

This section of code computes the current offset from the desired alignment for the tilt monitoring system. Note that error handling and interface specific code has been trimmed for brevity.

```
/** The update tilt position button was pressed.  Determine the tilt
 * offset from the baseline position.
 *
 */
private void updateTiltPositionButtonActionPerformed()
{
    //store the threshold level desired
    int tiltCentroidThresh;
    try
    {
        //multiply by 256 to convert to 16 bit format of averaged images
        //instead of 8 bit original image format
        tiltCentroidThresh = Math.round(Float.parseFloat(
                                        tiltThreshField.getText())*256);
    }//handle conversion exception here

    try
    {
        if(tiltcam == null)
        {
            throw new IDSException("Null camera.");
        }
```

```
        //obtain a new averaged image and subtract the background
        BufferedImage img = subtractImages(tiltcam.getAverageImage(ntilt),
                                           tiltback);
        //compute the current centroid
        Point2D imgCen = centroid(img, tiltCentroidThresh);

        //throw an error if no centroid could be computed
        if(imgCen.getX() == -1 && imgCen.getY() == -1)
        {
            tiltOffset = null;
            throw new AlignmentException("Unable to computer tilt offset.");
        }

        //compute the tilt offset amount from the stored position
        //store the offset in the appropriate variable
        tiltOffset = new Point2D.Double(tiltCenter.getX()-imgCen.getX(),
                tiltCenter.getY()-imgCen.getY());
    }//handle errors here
}
```

### A.1.4 Shear Alignment Offset

This section of code computes the current offset from the desired alignment for the shear monitoring system. Note that error handling and interface specific code has been trimmed for brevity.

```
/** The update shear position button was pressed.  Determine the shear
 * offset from the baseline position.
 *
 */
private void updateShearPositionButtonActionPerformed()
{
    try
    {
        if(shearcam == null)
        {
            throw new IDSException("Null camera.");
        }
        //obtain a new averaged image and subtract the background
```

```
        BufferedImage img = subtractImages(shearcam.getAverageImage(
                                            nshear), shearback);
        //compute the shear offset
        int imgOff[] = fitOffset(img, referenceArmImageBack,
                                 sampleArmImageBack);
        //throw an error if no offset could be computed
        if(imgOff == null)
        {
            shearOffset = null;
            throw new AlignmentException("Unable to compute shear offset.");
        }
        //store the offset into the appropriate variable
        shearOffset = new Point2D.Double(-imgOff[0], -imgOff[1]);
    }//handle errors here
}
```

The following function is called to compute the actual offset above.

```
/** Compute the offset between the combined image and the reference
 * and sample images.
 *
 * @param img The combined image to compare against.
 * @param ref The reference arm image to shift.
 * @param sam The sample arm image to shift.
 * @return An array of 4 integers containing the x and y shift for the
 * reference and sample images.  If any of the original images are
 * null, returns null.
 */
public int[] fitOffset(BufferedImage img, BufferedImage ref,
                       BufferedImage sam)
{
    //if any of the images are null, return null
    if(img == null || ref == null || sam == null)
    {
        return null;
    }

    //obtain the width and height of the base image
    //assume all 3 are the same
    int width = img.getWidth();
```

```
int height = img.getHeight();

//fit the reference and sample images to the combined image
int refx = 0;
int refy = 0;
int samx = 0;
int samy = 0;
//total number of steps = 2*steps+1
//this is the number of steps above and below 0
//increasing this parameter increases the computation time
//increasing this parameter improves resistance to non-smooth
//data
int steps = 1;
//the total range (both + and -) over which to look, in pixels
int range = 64;
//initialize the error measurement to a large value
int minval = Integer.MAX_VALUE;
//get the image rasters
//these contain the image data in an easily usable format
int[] curras = img.getData().getPixels(0, 0, width, height,
                                       (int[])null);
int[] refras = ref.getData().getPixels(0, 0, width, height,
                                       (int[])null);
int[] samras = sam.getData().getPixels(0, 0, width, height,
                                       (int[])null);
//loop until we have one pixel steps
//compute the shift in pixels from the reference and sample images
//to the combined image
for(;range >= 1;range /= (2*steps))
{
    //compute the starting and ending shifts for each image
    //these determine the search range at each iteration
    int mini = refx - range;
    int maxi = refx + range;
    int minj = refy - range;
    int maxj = refy + range;
    int mink = samx - range;
    int maxk = samx + range;
    int minl = samy - range;
    int maxl = samy + range;
```

```
//loop through the various image shifts
for(int i=mini;i <= maxi;i += range/steps)
{
    for(int j=minj;j <= maxj;j += range/steps)
    {
        for(int k=mink;k <= maxk;k += range/steps)
        {
            for(int l=minl;l <= maxl;l += range/steps)
            {
                int tmp = 0;
                for(int n=0;n<height;n++)
                {
                    for(int m=0;m<width;m++)
                    {
                        //compute the shifted difference
                        //use m+i+width and similar to ensure
                        //a positive remainder
                        //wraparound happens in this setup
                        //since the images should be dark
                        //at most edges, this doesn't cause
                        //problems
                        tmp += Math.abs(curras[m + n*width] -
                                refras[((m+i+width) % width) +
                                ((n+j+height) % height)*width] -
                                samras[((m+k+width) % width) +
                                ((n+l+height) % height)*width]);
                    }
                }
                //if we've found a reduced residual
                if(tmp < minval)
                {
                    //update the new best shift parameters
                    minval = tmp;
                    refx = i;
                    refy = j;
                    samx = k;
                    samy = l;
                }
            }
```

```
                }
            }
        }
    }

    //store the best shift parameters
    int toReturn[] = new int[4];
    toReturn[0] = refx;
    toReturn[1] = refy;
    toReturn[2] = samx;
    toReturn[3] = samy;

    if(debug)
    {
        System.out.println(refx+" "+refy+" "+samx+" "+samy);
    }

    return toReturn;
}
```

## A.2 Correction Code

The code in this section is used to calibrate our system and convert measured offsets to physical corrections. While the code has been simplified for clarity and brevity (for example, removing interface specific functions and hardware error monitoring code), the core algorithms should be apparent.

### A.2.1 Motor Calibration

The following code is used to calibrate the tilt correction motors to the tilt monitoring camera. Similar code is used for calibrating all the motors. As our required alignment corrections are calculated in pixel space, this calibration allows us to determine the required motor motions for different alignment offsets. Error handling and initialization code is omitted for brevity.

```
//obtain a 0 point image.
BufferedImage imgOri = subtractImages(tiltcam.getAverageImage(ntilt), tiltback);
```

```
//move in the positive X direction and take an image.
picocontroller.forward(motor1, stepsize);
BufferedImage f1Img = subtractImages(tiltcam.getAverageImage(ntilt), tiltback);

//move in the negative X direction and take an image.
picocontroller.reverse(motor1, stepsize);
BufferedImage r1Img = subtractImages(tiltcam.getAverageImage(ntilt), tiltback);

//move in the positive Y direction and take an image.
picocontroller.forward(motor2, stepsize);
BufferedImage f2Img = subtractImages(tiltcam.getAverageImage(ntilt), tiltback);

//move in the negative Y direction and take an image.
picocontroller.reverse(motor2, stepsize);
BufferedImage r2Img = subtractImages(tiltcam.getAverageImage(ntilt), tiltback);

//show the last image on screen
tiltPanel.changeImage(r2Img);

//compute the spot centers in each image
Point2D cenOri = centroid(imgOri, tiltCentroidThresh);
Point2D f1 = centroid(f1Img, tiltCentroidThresh);
Point2D r1 = centroid(r1Img, tiltCentroidThresh);
Point2D f2 = centroid(f2Img, tiltCentroidThresh);
Point2D r2 = centroid(r2Img, tiltCentroidThresh);

//convert the motions to appropriate parameters
//units are pixels per motor step
double m1Fx = (f1.getX() - cenOri.getX())/stepsize;
double m1Fy = (f1.getY() - cenOri.getY())/stepsize;
double m1Rx = -(r1.getX() - f1.getX())/stepsize;
double m1Ry = -(r1.getY() - f1.getY())/stepsize;
double m2Fx = (f2.getX() - r1.getX())/stepsize;
double m2Fy = (f2.getY() - r1.getY())/stepsize;
double m2Rx = -(r2.getX() - f2.getX())/stepsize;
double m2Ry = -(r2.getY() - f2.getY())/stepsize;

//update the appropriate fields
if(Math.abs(m1Fx) > Math.abs(m1Fy))
{
```

```
    //motor one moves more in the X direction than Y
    if(m1Fx > 0)
    {
        //motor one moves positive pixels for a forward move
        tilt1XField.setText(""+m1Fx);
        tilt1XNField.setText(""+m1Rx);
        tilt1YField.setText(""+m1Fy);
        tilt1YNField.setText(""+m1Ry);
    }else
    {
        //motor one moves negative pixels for a forward move
        tilt1XField.setText(""+-m1Rx);
        tilt1XNField.setText(""+-m1Fx);
        tilt1YField.setText(""+-m1Ry);
        tilt1YNField.setText(""+-m1Fy);
    }
}else
{
    //motor one moves more in the Y direction than X
    if(m1Fy > 0)
    {
        //motor one moves positive pixels for a forward move
        tilt1XField.setText(""+m1Fx);
        tilt1XNField.setText(""+m1Rx);
        tilt1YField.setText(""+m1Fy);
        tilt1YNField.setText(""+m1Ry);
    }else
    {
        //motor one moves negative pixels for a forward move
        tilt1XField.setText(""+-m1Rx);
        tilt1XNField.setText(""+-m1Fx);
        tilt1YField.setText(""+-m1Ry);
        tilt1YNField.setText(""+-m1Fy);
    }
}
if(Math.abs(m2Fx) > Math.abs(m2Fy))
{
    //motor two moves more in the X direction than Y
    if(m2Fx > 0)
    {
```

```
        //motor two moves positive pixels for a forward move
        tilt2XField.setText(""+m2Fx);
        tilt2XNField.setText(""+m2Rx);
        tilt2YField.setText(""+m2Fy);
        tilt2YNField.setText(""+m2Ry);
    }else
    {
        //motor two moves negative pixels for a forward move
        tilt2XField.setText(""+-m2Rx);
        tilt2XNField.setText(""+-m2Fx);
        tilt2YField.setText(""+-m2Ry);
        tilt2YNField.setText(""+-m2Fy);
    }
}else
{
    //motor two moves more in the Y direction than X
    if(m2Fy > 0)
    {
        //motor two moves positive pixels for a forward move
        tilt2XField.setText(""+m2Fx);
        tilt2XNField.setText(""+m2Rx);
        tilt2YField.setText(""+m2Fy);
        tilt2YNField.setText(""+m2Ry);
    }else
    {
        //motor two moves negative pixels for a forward move
        tilt2XField.setText(""+-m2Rx);
        tilt2XNField.setText(""+-m2Fx);
        tilt2YField.setText(""+-m2Ry);
        tilt2YNField.setText(""+-m2Fy);
    }
}
//correct for any residual offset in positioning
moveTilt(cenOri.getX() - r2.getX(), cenOri.getY() - r2.getY());
```

### A.2.2 Alignment Calibration Code

This is a sample of the algorithm used to calibrate the alignment system to improve alignment capabilities after significant system drifts. Similar code is used for other axes–

primarily using different motor axes and parameter variables–and omitted for brevity.

```
/** Maximize the throughput landing on the detector using the tilt
 * control.  No beams are unblocked or blocked--perform this first if
 * you wish to align using only a specific arm.
 *
 * @param moveTilt The initial step size for maximum searching of the
 * tilt axis.
 * @param tiltThresh Stop iterating when the requested movement size
 * is smaller than this.
 * @param reductionFactor The movement size is divided by this factor
 * every iteration.  Must be greater than one.
 */
public void tweakTiltAlignment(double moveTilt, double tiltThresh,
        double reductionFactor) throws AlignmentException
{
    if(reductionFactor <= 1)
    {
        throw new AlignmentException("Reduction factor must be " +
                                     "greater than 1: "+reductionFactor);
    }

    //get the initial flux value
    double lastFlux = getFlux();
    double curFlux;
    boolean direction = false;
    boolean swapped = false;

    //loop until we're moving less than our threshold
    while(Math.abs(moveTilt) >= tiltThresh)
    {
        //move a little in one direction
        if(direction)
        {
            moveTilt(moveTilt, 0);
        }else
        {
            moveTilt(0, moveTilt);
        }
```

```
        curFlux = getFlux();

        //if we've started going down in intensity
        if(curFlux < lastFlux)
        {
            //if we haven't already swapped directions for this axis
            if(!swapped)
            {
                //swap directions
                moveTilt = -moveTilt;
                swapped = true;
            }else
            {
                //reset the swapped variable
                swapped = false;
                //if we've swapped axes at this move size,
                //reduce the move size
                if(direction)
                {
                    moveTilt = moveTilt / reductionFactor;
                }
                //swap the axes
                direction = !direction;
            }
        }
        lastFlux = curFlux;
    }
}
```

### A.2.3 Pixel Offset to Motor Command Conversion

Our alignment monitoring code measures offsets from our desired alignment in pixel space. Our motor calibration provides us with conversion parameters from pixel space to motor movements. The code below shows the conversion process for our tilt system. Directly sending the converted commands to the motor (with or without a damping factor) is sufficient to maintain system alignment. Similar code, simply changing the appropriate variables, provides the same functionality for other axes. Interface specific code (such as confirming moves with the user) is omitted for brevity.

```
/** Move the motor controlling the tilt.
```

```
 *
 * @param x The number of pixels to shift horizontally by.
 * @param y The number of pixels to shift vertically by.
 * @throws AlignmentException If there is an error controlling the
 * motors.
 * @return Returns the number of motor counts moved by motor 1 and
 * motor 2.
 */
public Point2D moveTilt(double x, double y) throws AlignmentException
{
    //variables to store motor calibration data
    double x1;
    double y1;
    double x2;
    double y2;
    //load the motor calibration data from the interface
    try
    {
        if(x >= 0)
        {
            //use the positive movement X fields
            x1 = Double.parseDouble(tilt1XField.getText());
            x2 = Double.parseDouble(tilt2XField.getText());
        }else
        {
            //use the negative movement X fields
            x1 = Double.parseDouble(tilt1XNField.getText());
            x2 = Double.parseDouble(tilt2XNField.getText());
        }

        if(y >= 0)
        {
            //use the positive movement Y fields
            y1 = Double.parseDouble(tilt1YField.getText());
            y2 = Double.parseDouble(tilt2YField.getText());
        }else
        {
            //use the negative movement Y fields
            y1 = Double.parseDouble(tilt1YNField.getText());
            y2 = Double.parseDouble(tilt2YNField.getText());
```

```
    }
}catch(NumberFormatException ex)
{
    //error parsing the calibration data
    throw new AlignmentException(ex);
}

//these are the motors we wish to control for the tilt control
String motor1 = tiltMotor1IDField.getText();
String motor2 = tiltMotor2IDField.getText();

//calculate the amount to move each motor to get to the desired
//position
//n * x1 + m * x2 = x
//n * y1 + m * y2 = y
//two equations with two unknowns
//we assume that the motors are mostly orthogonal and mostly aligned
//with the image axes and so the following solutions should be stable:
//n = (x - y*x2/y2)/(x1 - y1*x2/y2) -- bad when y2->0
//n = (y - x*y2/x2)/(y1 - x1*y2/x2) -- bad when x2->0
//m = (x - n*x1)/x2 -- bad when x2->0
//m = (y - n*y1)/y2 -- bad when y2->0
//final movement will need to be rounded to an integer number of
//pico steps.  Our motors can only move discrete counts.
double n;
double m;
//determine which set of equations to use
if(Math.abs(x2) > Math.abs(y2))
{
    //x2 is less likely to be close to 0
    n = (y - x*y2/x2)/(y1 - x1*y2/x2);
    m = (x - n*x1)/x2;
}else
{
    //y2 is less likely to be close to 0
    n = (x - y*x2/y2)/(x1 - y1*x2/y2);
    m = (y - n*y1)/y2;
}

//round the amount to use each motor
```

```
    int motor1move = (int)Math.round(n);
    int motor2move = (int)Math.round(m);

    try
    {
        //forward with a negative value is equivalent to reverse
        //no need to check direction before sending the command.
        picocontroller.forward(motor1, motor1move);
        picocontroller.forward(motor2, motor2move);
    }catch(Exception ex)
    {
        throw new AlignmentException(ex);
    }

    //return the amount each motor was moved
    return new Point2D.Float(motor1move, motor2move);
}
```

## A.3 Reduction Code

Here we include samples of our code used to produce images from OCT data. This code is designed for interactive analysis to help us debug our system operations–it is not meant to be a fully featured reduction package.

### A.3.1 Initial Processing Code

This code takes our data (already loaded in) and extracts the sinusoidal interferograms that contain the spatial information for our images.

```
;identify the pixels to use
suse = 0
euse = 2047

;optionally isolate a subset of the date by uncommenting this
;adjust the selection variables as necessary
;array format =
;dimension 1 selects individual b-scans loaded
;dimension 2 selects spectrometer pixels
```

```
;dimension 3 selects a-scan repetitions
;dimension 4 selects different a-scan positions
;data = data[*,*,0,*]

;obtain the number of dimensions of each axis of data
sd = size(data)
;ignore specific data if needed by artificially fixing a dimension size
;sd[3] = 2

;create an array to store the processed data
data_avg = dblarr(sd[4], sd[2])

;loop through the different a-scan positions in the data set
for i=0,sd[4]-1 do begin
  ;select the appropriate averaging method
  if(sd[3] gt 1) then begin
    ;if we've chosen to average multiple b-scan repetitions
    data_avg[i,*] = reform(total(total(data[*,*,*,i],3),1))
  endif else begin
    ;only average a-scan repetitions
    data_avg[i,*] = reform(total(data[*,*,0,i],1))
  endelse
endfor
;normalize the averaged data to a single scan
data_avg = data_avg / double(sd[1]*sd[3])

;compute the average sample spectrum
samp = dblarr(n_elements(reference))
for i=0,n_elements(reference)-1 do begin
  samp[i] = mean(data_avg[*,i] - reference[i])
endfor

;normalize the data spectrum by some combination of the sample and
;reference spectra
;choose the method by uncommenting the appropriate lines
data_norm = data_avg
for i=0,sd[4]-1 do begin
  ;only subtract off the reference
  ;data_norm[i,*] = (data_avg[i,*] - reference)
```

```
  ;subtract off the reference, but first normalize the reference
  ;amplitude by the mean signal amplitude
  ;data_norm[i,*] = (data_avg[i,*] - mean(data_avg[i,*]/reference)*reference)

  ;compute the relative intensity offset between the mean sample
  ;spectrum and the reference subtracted data
  ;sampamp = mean(data_norm[i,*]/samp)

  ;subtract a rescaled copy of the average sample spectrum from the data
  ;data_norm[i,*] = (data_norm[i,*] - sampamp*samp)

  ;normalize the interferogram by the square root of the sample power
  ;times the reference power
  ;data_norm[i,*] = data_norm[i,*]/sqrt(sampamp*samp*reference)

  ;if we end up dividing by 0, stop the code and ask for user input
  ;if((where(sampamp*samp*reference le 0))[0] ne -1) then stop

  ;process with sampleonly data
  ;this is sample data actually measured for different points in the
  ;sample
  ;subtract off the reference and sample value at each sample point
  ;data_norm[i,*] = (data_avg[i,*] - sample[i,*] - reference)

  ;fit a low order polynomial to the data to correct residual low
  ;frequency noise
  zz = poly_fit(lindgen(n_elements(data_norm[i,*])),data_norm[i,*],3,yfit=yfit)
  ;data_norm[i,*] = data_norm[i,*] - yfit

  ;ignore anything uncommented above and remove reference signal
  data_norm[i,*] = (data_avg[i,*] - reference)
  ;sample only removal
  ;data_norm[i,*] = (data_norm[i,*] - sample[i,*])
  ;normalize by sample*reference power
  ;data_norm[i,*] = data_norm[i,*]/sqrt(reference*sample[i,*])
endfor

;ensure the data has 0 mean so no residual FT power exists
data_use = dblarr(sd[4], euse-suse+1)
for i=0,sd[4]-1 do begin
```

```
  data_use[i,*] = data_norm[i,suse:euse]
  data_use[i,*] = data_use[i,*] - mean(data_use[i,*])
endfor

;indicate that no wavelength->wavenumber resampling has been done
resampled = 0

end
```

### A.3.2 Image Generation Code

This code takes the sinusoidal interferograms extracted from our data, performs resampling and basic dispersion correction, and then generates images.

```
;approximate the wavelength of each spectrometer pixel
pixlow = 0
wnlow = 1d/0.790d
pixhigh = 2047
wnhigh = 1d/0.890d
dwn = wnlow-wnhigh
dpix = pixhigh-pixlow
dwnpix = dwn/dpix
nuse = n_elements(data_use[0,*])
dum = dindgen(nuse)/(2*dwnpix*nuse)

;resample to wavenumber space
if(1) then begin
  if(resampled eq 0) then begin
    data_use_ori = data_use
    resampled = 1
    wavenumbers = dindgen(pixhigh-pixlow+1)/(pixhigh-pixlow)* $
                  (wnhigh-wnlow)+wnlow
    lambdas_wn = 1/wavenumbers
    lambdas = dindgen(pixhigh-pixlow+1)/(pixhigh-pixlow)* $
              (1d/wnhigh - 1d/wnlow)+1d/wnlow
    for i=0,n_elements(data_use[*,0])-1 do begin
      data_use[i,*] = interpol(data_use[i,*], lambdas, lambdas_wn, /spline)
    endfor
  endif
```

```
endif

;initialize some storage variables
img = data_use*0
data_disp = data_use*0
img_disp = img*3

;optional dispersion parameters
;dist0 = -3d6
dist0 = 0d6
;change the dispersion linearly for each a-scan position
diststep = 0;2.25d6/49

;loop over all the a-scan positions
for i=0,n_elements(img[*,0])-1 do begin
  ;compute the FFT of the Hilbert transform of the data
  img[i,*] = (abs(fft(complex(data_use[i,*], hilbert(data_use[i,*], -1)))))

  ;apply some dispersion correction to the data before
  ;computing the FFT of the Hilbert transform
  data_disp[i,*] = dispersion_correction(data_use,i,dist0+diststep*fend,$
                                         wavenumbers)
  img_disp[i,*] = (abs(fft(complex(data_disp[i,*], $
                                   hilbert(data_disp[i,*], -1)))))
end

;compute the number of useful image pixels
nels = (euse-suse)/2

;convert the image to a log scale
img_sub = (alog10(img[*,0:nels] + min(img[*,0:nels]) + 1))
;create an 8 bit version of the image
img_res = round((img_sub+min(img_sub))/(max(img_sub)-min(img_sub))*(2^8))

;convert the dispersion corrected image to a log scale
img_disp_sub = alog10(img_disp[*,0:nels] + min(img_disp[*,0:nels]) + 1)
;create an 8 bit version of the dispersion corrected image
img_disp_res = round((img_disp_sub+min(img_disp_sub))/ $
                     (max(img_disp_sub)-min(img_disp_sub))*(2^8))
```

```
;different display methods
;write the 8 bit non-dispersion corrected image to a tiff file
;write_tiff, samplename+'_'+strtrim(fstart,2)+'-'+$
             strtrim(fend,2)+'.tiff', img_res
;write the 8 bit dispersion corrected image to a tiff file
;write_tiff, samplename+'_disp_'+strtrim(fstart,2)+'-'+$
             strtrim(fend,2)+'.tiff', img_disp_res
;display the 8 bit non-dispersion corrected image in a window
;flip the image vertically
;iimage, reverse(img_res,2)
;display the 8 bit dispersion corrected image in a window
;flip the image vertically
;iimage, reverse(img_disp_res,2)
;display the 8 bit dispersion corrected image in a window
;iimage, img_disp_res
;display the non-dispersion corrected image in a window
;flip the image vertically
;iimage, reverse(img_sub,2)
;display the dispersion corrected image in a window
;flip the image vertically
;iimage, reverse(img_disp_sub,2)
;display the dispersion corrected image in a window
;flip the image vertically and horizontally
;iimage, reverse(reverse(img_disp_sub,2),1)
;display the dispersion corrected image in a window
;iimage, img_disp_sub
;display the dispersion corrected image in a window
;flip the image horizontally
iimage, reverse(img_disp_sub,1)

end
```

### A.3.3 Dispersion Compensation Code

Here we provide the code for our basic dispersion correction algorithm. We use a simplified form of the algorithm presented by Wojtkowski et al. [45].

```
;uses a simplified form of the algorithm from
;Wojtkowski et al. May 2004 (Optics Express Vol. 12 No. 11)
```

```
;and use Sellmeier's equation for refractive index
;from http://en.wikipedia.org/wiki/Sellmeier_equation
;assume BK7 glass
function dispersion_correction, data_use, id, dist, freq

;Sellmeier parameters
B1 = 1.03961212D
B2 = 0.231792344D
B3 = 1.01046945D
C1 = 6.00069867d-3 ;um^2
C2 = 2.00179144d-2 ;um^2
C3 = 1.03560653d2  ;um^2
;Sellmeier's Equation:
;Beta(lambda) = eta(lambda)^2
;             = 1 + B1*lambda^2/(lambda^2 - C1)
;                 + B2*lambda^2/(lambda^2 - C2)
;                 + B3*lambda^2/(lambda^2 - C3)

;obtain the hilbert transform of the data
hil_data = complex(data_use[id,*], hilbert(data_use[id,*], -1))

;take the magnitude and phase
mag = abs(hil_data)
phase = atan(hil_data, /phase)

;compute the Sellmeier equation
beta = (1d + B1*(1/freq)^2/((1/freq)^2 - C1) $
           + B2*(1/freq)^2/((1/freq)^2 - C2) $
           + B3*(1/freq)^2/((1/freq)^2 - C3))
eta = sqrt(beta)

;first derivative
dbeta = (beta-shift(beta,1))/(freq-shift(freq,1))
;remove edge effect
dbeta[0] = dbeta[1]

;second derivative
d2beta = (dbeta-shift(dbeta,1))/(freq-shift(freq,1))
;remove edge effects
d2beta[1] = d2beta[2]
```

```
d2beta[0] = d2beta[1]

;third derivative
d3beta = (d2beta-shift(d2beta,1))/(freq-shift(freq,1))
;remove edge effects
d3beta[2] = d3beta[3]
d3beta[1] = d3beta[2]
d3beta[0] = d3beta[1]

;fourth derivative
d4beta = (d3beta-shift(d3beta,1))/(freq-shift(freq,1))
;remove edge effects
d4beta[3] = d4beta[4]
d4beta[2] = d4beta[3]
d4beta[1] = d4beta[2]
d4beta[0] = d4beta[1]

;fifth derivative
d5beta = (d4beta-shift(d4beta,1))/(freq-shift(freq,1))
;remove edge effects
d5beta[4] = d5beta[5]
d5beta[3] = d5beta[4]
d5beta[2] = d5beta[3]
d5beta[1] = d5beta[2]
d5beta[0] = d5beta[1]

;sixth derivative
d6beta = (d5beta-shift(d5beta,1))/(freq-shift(freq,1))
;remove edge effects
d6beta[5] = d6beta[6]
d6beta[4] = d6beta[5]
d6beta[3] = d6beta[4]
d6beta[2] = d6beta[3]
d6beta[1] = d6beta[2]
d6beta[0] = d6beta[1]

;choose which derivatives to use for dispersion correction
a1 = 0d;dbeta[nlambda/2]
a2 = 0.5d * d2beta[nlambda/2]
a3 = 1d/6d * d3beta[nlambda/2]
```

```
a4 = 0;1d/24d * d4beta[nlambda/2]
a5 = 0;1d/120d * d5beta[nlambda/2]
a6 = 0;1d/720d * d6beta[nlambda/2]

;compute the corrected phase at the desired position
cor_phase = phase - dist*(a1*(freq-freq[nlambda/2]) $
                          + a2*(freq-freq[nlambda/2])^2 $
                          + a3*(freq-freq[nlambda/2])^3 $
                          + a4*(freq-freq[nlambda/2])^4 $
                          + a5*(freq-freq[nlambda/2])^5 $
                          + a6*(freq-freq[nlambda/2])^6)
;compute the corrected data using the corrected phase
cor_data = complex(mag*cos(cor_phase), mag*sin(cor_phase))

return, cor_data

end
```

# References

[1] F. Acernese, P. Amico, M. Alshourbagy, S. Aoudia, S. Avino, D. Babusci, G. Ballardin, F. Barone, L. Barsotti, M. Barsuglia, F. Beauville, M.A. Bizouard, C. Boccara, F. Bondu, L. Bosi, C. Bradaschia, S. Braccini, A. Brillet, V. Brisson, L. Brocco, D. Buskulic, E. Calloni, E. Campagna, F. Cavalier, R. Cavalieri, G. Cella, E. Chassande-Mottin, C. Corda, A.-C. Clapson, F. Cleva, J.-P. Coulon, E. Cuoco, V. Dattilo, M. Davier, R. De Rosa, L. Di Fiore, A. Di Virgilio, B. Dujardin, A. Eleuteri, D. Enard, I. Ferrante, F. Fidecaro, I. Fiori, R. Flaminio, J.-D. Fournier, S. Frasca, F. Frasconi, L. Gammaitoni, A. Gennai, A. Giazotto, G. Giordano, L. Giordano, R. Gouaty, D. Grosjean, G. Guidi, S. Hebri, H. Heitmann, P. Hello, L. Holloway, S. Kreckelbergh, S. Karkar, P. La Penna, N. Letendre, V. Loriette, M. Loupias, G. Losurdo, J.-M. Mackowski, E. Majorana, C. N. Man, M. Mantovani, F. Marchesoni, F. Marion, J. Marque, F. Martelli, A. Masserot, M. Mazzoni, L. Milano, C. Moins, J. Moreau, N. Morgado, B. Mours, A. Pai, C. Palomba, F. Paoletti, S. Pardi, A. Pasqualetti, R. Passaquieti, D. Passuello, F. Perniola, F. Piergiovanni, L. Pinard, R. Poggiani, M. Punturo, P. Puppo, K. Qipiani, P. Rapagnani, V. Reita, A. Remillieux, F. Ricci, I. Ricciardi, P. Ruggi, G. Russo, S. Solimeno, A. Spallicci, R. Stanga, R. Taddei, M. Tonelli, M. Tonelli, A. Toncelli, E. Tournefier, F. Travasso, G. Vajente, D. Verkindt, F. Vetrano, A. Vicer, J.-Y. Vinet, H. Vocca, M. Yvert, and Z. Zhang. The automatic alignment system of the Virgo interferometer. *10th ICALEPCS International Conference on Accelerator & Large Experimental Physics Control Systems*, Oct 2005. 4

[2] Avni Ceyhun Akcay. *System Design and Optimization of Optical Coherence Tomography.* PhD thesis, University of Central Florida, Orlando, Florida, 2005.

[3] F. Barone, L. Di Fiore, L. Milano, G. Russo, and S. Solimeno. Automatic alignment of a michelson interferometer. *Nuclear Science, IEEE Transactions on*, 39(2):232 –237, April 1992.

[4] K. Bizheva, B. Považay, B. Hermann, H. Sattmann, W. Drexler, M. Mei, R. Holzwarth, T. Hoelzenbein, V. Wacheck, and H. Pehamberger. Compact, broad-bandwidth fiber

laser for sub-2-$\mu$m axial resolution optical coherence tomography in the 1300-nm wavelength region. *Opt. Lett.*, 28(9):707–709, 2003. 1, 2

[5] Kostadinka Bizheva, Angelika Unterhuber, Boris Hermann, Boris Považay, Harald Sattmann, A. F. Fercher, Wolfgang Drexler, Matthias Preusser, Herbert Budka, Andreas Stingl, and Tuan Le. Imaging ex vivo healthy and pathological human brain tissue with ultra-high-resolution optical coherence tomography. *Journal of Biomedical Optics*, 10(1):011006, 2005. 1

[6] Max Born and Emil Wolf. *Principles of Optics*. Cambridge University Press, Cambridge, United Kingdom, seventh expanded edition, 1999. 15

[7] Andrew T. Cenko, Bradford B. Behr, Peter B. Christensen, Arsen R. Hajian, Jan Hendrikse, Jeff T. Meade, Frederic D. Sweeney, and Paul van der Vecht. Automatic alignment of a high-performance interferometric medical imaging device. In *Proceedings of the SPIE*, volume 7890. SPIE, 2011. 75

[8] Andrew T. Cenko, Arsen R. Hajian, and Jeff T. Meade. Methods and apparatus for alignment of interferometer. US provisional patent application no. 61/414,044, 2010. 75

[9] Barry Cense, Nader Nassif, Teresa Chen, Mark Pierce, Seok-Hyun Yun, B. Park, Brett Bouma, Guillermo Tearney, and Johannes de Boer. Ultrahigh-resolution high-speed retinal imaging using spectral-domain optical coherence tomography. *Opt. Express*, 12(11):2435–2447, 2004. 1, 14

[10] Wolfgang Drexler. Ultrahigh-resolution optical coherence tomography. *Journal of Biomedical Optics*, 9(1):47–74, 2004. 1, 2, 14

[11] Wolfgang Drexler and James G. Fujimoto, editors. *Optical Coherence Tomography - Technology and Applications*. Springer, New York, New York, United States, 2008. xi, 11, 12, 13, 15, 16, 17

[12] Arnaud Dubois, Kate Grieve, Gael Moneron, Romain Lecaque, Laurent Vabre, and Claude Boccara. Ultrahigh-resolution full-field optical coherence tomography. *Appl. Opt.*, 43(14):2874–2883, 2004.

[13] A F Fercher, W Drexler, C K Hitzenberger, and T Lasser. Optical coherence tomography - principles and applications. *Reports on Progress in Physics*, 66(2):239, 2003. 1, 2, 14

[14] Adolf F. Fercher. Optical coherence tomography. *Journal of Biomedical Optics*, 1(2):157–173, 1996.

[15] New Focus. Picomotor actuator product page, 2010. [Online; accessed 16-December-2010]. 4

[16] Kenneth A. Goldberg, Patrick Naulleau, and Jeffrey Bokor. Fourier transform interferometer alignment method. *Appl. Opt.*, 41(22):4477–4483, 2002.

[17] John E. Greivenkamp. *Field Guide to Geometrical Optics.* SPIE - The International Society for Optical Engineering, Bellingham, Washington, United States, 2004.

[18] Ireneusz Grulkowski, Michalina Gora, Maciej Szkulmowski, Iwona Gorczynska, Daniel Szlag, Susana Marcos, Andrzej Kowalczyk, and Maciej Wojtkowski. Anterior segment imaging with spectral oct system using a high-speed cmos camera. *Opt. Express*, 17(6):4842–4858, 2009.

[19] Arsen R. Hajian, Bradford B. Behr, Andrew T. Cenko, Robert P. Olling, David Mozurkewich, J. Thomas Armstrong, Brian Pohl, Sevan Petrossian, Kevin H. Knuth, Robert B. Hindsley, Marc Murison, Michael Efroimsky, Ronald Dantowitz, Marek Kozubal, Douglas G. Currie, Tyler E. Nordgren, Christopher Tycner, and Robert S. McMillan. Initial results from the usno dispersed fourier transform spectrograph. *The Astrophysical Journal*, 661(1):616, 2007. 11

[20] Michael R. Hee, David Huang, Eric A. Swanson, and James G. Fujimoto. Polarization-sensitive low-coherence reflectometer for birefringence characterization and ranging. *J. Opt. Soc. Am. B*, 9(6):903–908, 1992.

[21] Y. Hefetz and N. Mavalvala. Sensitivity of the LIGO Interferometer to Mirror Misalignment and Method for Automatic Alignment. In R. T. Jantzen, G. Mac Keiser, & R. Ruffini, editor, *Proceedings of the Seventh Marcel Grossman Meeting on recent developments in theoretical and experimental general relativity, gravitation, and relativistic field theories*, pages 1349–+, 1996. 4

[22] IDS Imaging. *USB UI-1225LE Product Page.* 2010. [Online; accessed 16-December-2010].

[23] D. Kalamatianos, J.M. Edmunds, P.E. Wellstead, R.J. Houston, P. Liatsis, S.M. Christie, R.J. Dewhurst, and M.S. Thorniley. Dynamic alignment system for an ft-nir michelson interferometer. In *Proceedings of the IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 120–124, 2004.

[24] Sang-Won Lee, Hyun-Woo Jeong, Yeh-Chan Ahn, Woonggyu Jung, Zhongping Chen, and Beop-Min Kim. Axial resolution and depth range of high-resolution spectral domain optical coherence tomography at 1.3 mu m. volume 7168, page 71682L. SPIE, 2009.

[25] Sang-Won Lee, Hyun-Woo Jeong, and Beop-Min Kim. High-speed spectral domain polarization- sensitive optical coherence tomography using a single camera and an optical switch at 1.3 $\mu$m. *Journal of Biomedical Optics*, 15(1):010501, 2010. 1

[26] R. Leitgeb, C. Hitzenberger, and Adolf Fercher. Performance of fourier domain vs. time domain optical coherence tomography. *Opt. Express*, 11(8):889–894, 2003. 11, 13

[27] A J Lewis and D J Pugh. Interferometer light source and alignment aid using single-mode optical fibres. *Measurement Science and Technology*, 3(9):929, 1992.

[28] G. Loeb and J. K. Barton. Imaging botanical subjects with optical coherence tomography: A feasibility study. *Transactions of the ASAE*, 46(6):1751–1757, 2003.

[29] Z J Lu, R A Koehler, W A Gault, and F C Liang. A dynamic alignment system for scanning michelson interferometers. *Journal of Physics E: Scientific Instruments*, 21(1):71, 1988.

[30] Leonard Mandel and Emil Wolf. *Optical Coherence and Quantum Optics*. Cambridge University Press, Cambridge, United Kingdom, 1995. 5, 6, 7, 9, 10

[31] M Mantovani and A Freise. Evaluating mirror alignment systems using the optical sensing matrix. *Journal of Physics: Conference Series*, 122(1):012026, 2008.

[32] Euan Morrison, Brian J. Meers, David I. Robertson, and Henry Ward. Automatic alignment of optical interferometers. *Appl. Opt.*, 33(22):5041–5049, Aug 1994.

[33] Freddy T. Nguyen, Adam M. Zysk, Eric J. Chaney, Jan G. Kotynek, Uretz J. Oliphant, Frank J. Bellafiore, Kendrith M. Rowland, Patricia A. Johnson, and Stephen A. Boppart. Intraoperative Evaluation of Breast Tumor Margins with Optical Coherence Tomography. *Cancer Research*, 69(22):8790–8796, 2009. 23

[34] F. C. Nix and D. MacNair. The thermal expansion of pure metals: Copper, gold, aluminum, nickel, and iron. *Phys. Rev.*, 60(8):597–605, Oct 1941. 30

[35] Alois K. Popp, Megan T. Valentine, Peter D. Kaplan, and David A. Weitz. Microscopic origin of light scattering in tissue. *Appl. Opt.*, 42(16):2871–2880, 2003.

[36] Benjamin Potsaid, Iwona Gorczynska, Vivek J. Srinivasan, Yueli Chen, James Jiang, Alex Cable, and James G. Fujimoto. Ultrahigh speed spectral / fourier domain oct ophthalmic imaging at70,000 to 312,500 axial scans per second. *Opt. Express*, 16(19):15149–15169, 2008.

[37] B. Povazay, K. Bizheva, A. Unterhuber, B. Hermann, H. Sattmann, A. F. Fercher, W. Drexler, A. Apolonski, W. J. Wadsworth, J. C. Knight, P. St. J. Russell, M. Vetterlein, and E. Scherzer. Submicrometer axial resolution optical coherence tomography. *Opt. Lett.*, 27(20):1800–1802, 2002. 1, 2

[38] Joseph M. Schmitt. Optical coherence tomography (oct): A review. *IEEE Journal of Selected Topics in Quantum Electronics*, 5(4):1205–1215, 1999.

[39] René Schödel and Gerhard Bönsch. Highest-accuracy interferometer alignment by retroreflection scanning. *Appl. Opt.*, 43(31):5738–5743, 2004.

[40] Thorlabs. *Thorlabs Catalog, Vol. 20.* Thorlabs, Newton, NJ, USA, 2009. xvi, 2, 4, 30, 38, 40, 41

[41] Martin van Buren and Nabeel A. Riza. Foundations for low-loss fiber gradient-index lens pair coupling with the self-imaging mechanism. *Appl. Opt.*, 42(3):550–565, 2003. 2

[42] Lihong V. Wang and Hsin i Wu. *Biomedical Optics - Principles and Imaging.* John Wiley & Sons Inc., Hoboken, New Jersey, United States, 2007.

[43] Wikipedia. Optical window in biological tissue — wikipedia, the free encyclopedia, 2010. [Online; accessed 16-December-2010].

[44] Wikipedia. Sellmeier equation — wikipedia, the free encyclopedia, 2010. [Online; accessed 16-December-2010].

[45] Maciej Wojtkowski, Vivek Srinivasan, Tony Ko, James Fujimoto, Andrzej Kowalczyk, and Jay Duker. Ultrahigh-resolution, high-speed, fourier domain optical coherence tomography and methods for dispersion compensation. *Opt. Express*, 12(11):2404–2422, 2004. 102

[46] Ping Xue and James G. Fujimoto. Ultrahigh resolution optical coherence tomography with femtosecond ti:sapphire laser and photonic crystal fiber. *Chinese Science Bulletin*, 53(13):1963–1966, 2008. 1, 2

[47] Gang Yao and Lihong V Wang. Monte carlo simulation of an optical coherence tomography signal in homogeneous turbid media. *Physics in Medicine and Biology*, 44(9):2307, 1999. 30

[48] Shifu Yuan and Nabeel A. Riza. General formula for coupling-loss characterization of single-mode fiber collimators by use of gradient-index rod lenses. *Appl. Opt.*, 38(15):3214–3222, 1999. 2

[49] Adam M. Zysk, Freddy T. Nguyen, Amy L. Oldenburg, Daniel L. Marks, and Stephen A. Boppart. Optical coherence tomography: a review of clinical development from bench to bedside. *Journal of Biomedical Optics*, 12(5):051403, 2007. 1, 2, 14, 17, 18