CONVERTING A 2D SCANNING LIDAR TO A 3D SYSTEM FOR USE ON QUAD-ROTOR UAVS IN AUTONOMOUS NAVIGATION

THESIS

Matthew Aaron Cooper, 1st Lt, USAF

AFIT-ENG-MS-17-M-019

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENG-MS-17-M-019

CONVERTING A 2D SCANNING LIDAR TO A 3D SYSTEM FOR USE ON QUAD-ROTOR UAVS IN AUTONOMOUS NAVIGATION

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Matthew Aaron Cooper, B.S.E.E.

1st Lt, USAF

March 2017

AFIT-ENG-MS-17-M-019

CONVERTING A 2D SCANNING LIDAR TO A 3D SYSTEM FOR USE ON QUAD-ROTOR UAVS IN AUTONOMOUS NAVIGATION

THESIS

Matthew Aaron Cooper, B.S.E.E.
1st Lt, USAF

Committee Membership:

Dr. John F. Raquet
Chair

Dr. David R. Jacques
Member

Dr. Scott L. Nykl
Member

AFIT-ENG-MS-17-M-019

Abstract

A primary concern with adding sensors is the increase in energy consumption and weight, with a corresponding decrease in flight time, for each additional sensor. This research proposes equipping an existing two dimensional (2D) LiDAR sensor with a reflective super-structure to deflect the projected beam pattern to other areas of interest, thereby converting it into a three dimensional (3D) system, and reducing the need for additional sensors in autonomous Unmanned Ariel Vehicles (UAV) applications. This research proposes an externally mounted lightweight reflective structure which eliminates the need for additional moving parts, such as actuators for sensor re-orientation, that have a much higher failure rate than solid rigid parts.

This thesis describes the prototype design and testing with an emphasis on calibration algorithms. Once calibrated, the modified sensor allows for a smaller physical and computational footprint to perform navigation and mapping techniques with a single sensor in GPS-denied environments. By modifying the existing scanning LiDAR sensor, the data gathered by the sensor can be exploited beyond the initial 2D scanning capability. By limiting the new orientation of the sensor with respect to body frame of the UAV, the transformation between the 2D output can be pre-calculated before flight using the proposed calibration algorithm. Once calculated, the transformation can be implemented in real-time during flight by simple matrix multiplication in order to sense a 3D environment. The calibration algorithm was tested and shown to have an angular repeatability of ± 0.5deg in estimating both azimuth, and elevation of the deflected sensor beams caused by the reflective surfaces of the prototype.

Acknowledgements

Great appreciation goes out to my Advisor and to my Family, for without their guidance and support this research would have ended up as a smoldering fireball in a ditch alongside the road to nowhere.

Matthew Aaron Cooper

Table of Contents

Page

Page

List of Figures

List of Tables

Table Page

List of Abbreviations

Abbreviation		Page
2D	two dimensional	iv
3D	three dimensional	iv
UAV	Unmanned Ariel Vehicles	iv
GPS	global positioning system	1
LiDAR	Light Detection and Ranging	1
HUMVEE	High Mobility Multipurpose Vehicle	2
RGB-D	Red-Green-Blue-Depth	5
FFT	Fast Fourier Transform	8
COTS	commercial-off-the-shelf	10
LASER	Light Amplification by Stimulated Emission of Radiation	10
TOF	time of flight	15
SLM	Spatial Light Modulator	15
FTH	Fourier Transform Heterodyne	17
APD	avalanche photodiodes	17
PMF	probability mass function	21
LRT	likelihood ratio test	21
IR	infra-red	29
RANSAC	RANdom SAmple and Consensus	32
PCA	Principal Component Analysis	32
SVD	Singular Value Decomposition	32
ICP	Iterative Closest Point	32

Abbreviation		Page
LDP	Location Determination Problem	32
ADMM	Alternating Direction Method of Multipliers	36
SLAM	Simultaneous Localization and Mapping	36
CAD	computer aided drafting	37
ROS	Robot Operating System	41
VM	virtual machine	43
CSV	comma separated variable	43
MDF	medium density fiberboard	69
RLS	Recursive Least Squares	72
ERLS	Extended Recursive Least Squares	134
WRLS	Weighted Recursive Least Squares	134
EWRLS	Extended/Weighted Recursive Least Squares	134
DOF	degrees of freedom	138
RMSE	Root Means Squared Error	154

List of Symbols

Symbol		Page
j	square root of -1	8
f_x	spatial frequency component in x	8
f_y	spatial frequency component in y	8
λ	wavelength of light	8
$U(f_x, f_y)$	phase plane at light source	8
$U(x, y)$	phase plane at receiving location	8
θ_0	angle of incidence from normal	9
n	index of refraction	10
c	velocity of light in a vacuum	10
v	velocity of light in a medium	10
τ_a	atmoshperic transmission rate	13
τ_o	optical system transmission rate	13
D_R	recieving aperture diameter of LiDAR detector	13
ρ_t	target surface reflectivity	13
P_t	trasmitted power of LASER	13
θ_d	angular divergence of transmitted LASER beam	13
θ_R	angular divergence of transmitted LASER beam due to target	13
dA	area of the target as seen by LiDAR detector	13
R	distance between LASER source and target	13
P_r	reflected power	13
P_i	peak reflected power	13

Symbol		Page
θ_i	azimuth angle of incident beam	13
ϕ_i	elevation angle of incident beam	13
θ_r	azimuth angle of reflected beam	14
ϕ_r	elevation angle of reflected beam	14
h	denotes surface integral	14
$d\Omega$	incremental changes in steradians with respect to surface integral	14
$\rho(\theta_T, \phi_T)$	function describing the reflectivity pattern of a surface whose reflectivity is not perfectly specular or lambertian	14
θ_T	functional variable representing the inputs θ_i, and θ_r	14
ϕ_T	functional variable representing the inputs ϕ_i, and ϕ_r	14
t	time	15
H_1	hypothosis one in binary hypothesis test	21
H_0	hypothosis two in binary hypothesis test	21
$P(D)$	probability of detect	21
$P(D\|H_0)$	probability of detect given that a target is not present	21
$P(H_1\|D)$	probability of target is detected given that a detect occurred	21
$P(H_0\|D)$	probability that no target is detected given that a detect occurred	21
Λ	likelihood ratio test result	21
θ_z	rotiaion angle about the z-axis	24
P_s	arbitrary point in the sensor frame	24
T_S^R	transformation matrix between the sensor frame to a robot body frame	25

Symbol		Page
T_R^W	transformation matrix between the robot body frame to a world frame	25
U	m×n unitary matrix for SVD algorithm	33
S	m×n non-negative diagonal matrix for SVD algorithm	33
V	n×n unitary matrix for SVD algorithm	33
P_P	target board plane coordinate set comprised of P_{P1}, P_{P3}, and P_{P3}	64
θ_B	azimuth of the originating LiDAR beam with respect to the sensor origin	64
θ_a	azimuth angle of LiDAR beam as a result of the deflection due to a mirror	64
ϕ	elevation angle of LiDAR beam as a result of the deflection due to a mirror	64
d	distance to deflection point of mirror with respect to the sensor origin	64
ν	unknown environmental noise	64
$r(\tilde{x})$	estimated range of LiDAR sensor	64
P_{L1}	deflection point of LiDAR beam in 3D space with respect to the sensor origin	65
P_{L2}	end point of line with arbitrary length, starting at P_{L2} propagated with θ_b ,$\Delta\theta$, ϕ, and d	65
P_{target}	intercept point of deflected LiDAR beam with calibration target	65
$\vec{N_P}$	normal vector of target board	66
a	normal vector component of target board in x direction	66
b	normal vector component of target board in y direction	66
c	normal vector component of target board in z direction	66

Symbol Page

ρ_d arbitray length to help define the line described by P_{L1}, and P_{L2} 66

d_2 distance between P_{L1}, and P_{target} 66

d_3 distacnce between P_{target}, and P_{L2} 66

θ azimuth angle of LiDAR beam with respect to the sensor origin 67

V_L vector described by $P_{L1} - P_{L2}$ 67

x_x x-coordinate of beam intercept point with respect to LiDAR origin 67

y_y y-coordinate of beam intercept point with respect to LiDAR origin 67

z_z z-coordinate of beam intercept point with respect to LiDAR origin 67

r_{gray} general range error offset due to the color of target board 68

r_{offset} range error offset due to orientation of the target board 68

$r(\hat{x})$ error corrected estimated range of LiDAR sensor 69

P_c point in space of the target board centroid center represented by P_{cx}, P_{cy}, and P_{cz} 71

r_c distance to target board centroid with respect to LiDAR origin 72

$\delta\theta$ small perturbations in θ to numerically calculate RLS Jacobian 73

$\delta\phi$ small perturbations in ϕ to numerically calculate RLS Jacobian 73

δd small perturbations in d to numerically calculate RLS Jacobian 73

H Jacobian matrix consisting of the partial differentials of the range estimation function 73

$\hat{\theta}$ current iteration estimate of θ in RLS algorithm 74

Symbol		Page
$\hat{\phi}$	current iteration estimate of ϕ in RLS algorithm	74
$\hat{d}$	current iteration estimate of d in RLS algorithm	74
$\Delta\theta$	incremental correction in θ in the current iteration of the RLS algorithm	74
$\hat{\theta}_{new}$	new estimate of θ derived in the current iteration of the RLS algorithm	74
$\hat{\phi}_{new}$	new estimate of ϕ derived in the current iteration of the RLS algorithm	74
$\hat{d}_{new}$	new estimate of d derived in the current iteration of the RLS algorithm	74
W	weighting matrix in weighted RLS algorithm	75
σ	variance of sensor measurement due to the target	75
ϵ	extended Jacobian parameters in Extended RLS algorithm	75
$R_{\hat{\theta}\hat{\phi}}$	3-2 rotation matrix using $\hat{\theta}$, and $\hat{\phi}$	81
$t_0^1[n]$	the translation of the n^{th} beam from the origin to the 1^{st} frame	81
PC_0	initial 2D planar point cloud from the LiDAR sensor range measurements	82
PC_1	transformed 3D point cloud from the original 2D planar point cloud	82
$E_{r_{nm}}$	range test error as a function of distance to target	96
E_{r_m}	range test error as a function of distance to target, with a mirror in the beam path	96
Θ	LiDAR beam divergence angle	99
E_a	range function error due to the orientation of the target board	109
θ_N	angle between the normal vector of the target board and the LiDAR beam	109

Symbol		Page
E_w	LiDAR range error due to using a white target	121
E_b	LiDAR range error due to using a black target	121
I_{nm}	LiDAR sensor intensity return as a function of distance on a white target	132
I_m	LiDAR sensor intensity return as a function of distance on a white target, and a mirror in beam path	132

CONVERTING A 2D SCANNING LIDAR TO A 3D SYSTEM FOR USE ON QUAD-ROTOR UAVS IN AUTONOMOUS NAVIGATION

I. Introduction

1.1 Problem Definition

Indoor autonomous navigation is a challenging topic, because global positioning system (GPS) is generally unavailable indoors. Without the accuracy of GPS positioning data for localization of an autonomous vehicle, the vehicle must rely on alternative positioning sources to locate themselves, map the environment, and to perform path planning. A plethora of sensors are available for unmanned vehicles. Of these, a common choice is an electro-optic camera system which captures images within the field of view of the lens systems. Camera systems can provide an enormous amount of data. An iPhone 7, for example, can record 1080p resolution video at up to 60 frames per second which nears a 3 Gigabit per second data rate for imagery for autonomous vehicles[11]. Light Detection and Ranging (LiDAR) systems are another common sensor. A multiple plane scanning LiDAR sensor can collect up to 2.2 million data points per second for a system designed to capture a 3D environment, one example is the Velodyne[12] sensor used in the prototype autonomous car first developed by Google[13]. These types of sensors offer robust solutions in physically large packages, with potentially megabytes, or gigabytes, of data to be processed in real time to garner a navigation solution. What about small scale sensors that can achieve a similar result?

There are many cases where autonomous navigation in an indoor type situation may arise. Lets imagine one of these possible scenarios for a moment. You're a security forces

member on duty in Iraq, and you've just been called up as part of an initial reaction force to investigate a potential source of mortar attacks that just occurred on base. The base is on red alert, and you still see base personnel hiding under whatever cover was available to them, as you speed off to location with the rest of your squad in an armored up military HUMVEE with 4-inch thick bullet proof windows. Cautiously arriving on scene to the potential origin of the attacks, someone sees a distant dust trail of a white SUV fleeing towards Baghdad. A rapid pursuit is initiated. Holding onto your M4 semi-automatic rifle in one hand, and the inside door edge in the other with white knuckles, you are mentally willing your vehicle closer to the enemy. The HUMVEE bounds through rough terrain, and unpaved roads, attempting to close in on the SUV whose driver is much more familiar with that route than yours. Now entering the city, and a new set of obstacles to avoid present themselves immediately as the your vehicle sharply swerves around local pedestrians in an attempt not to loose the SUV in the cluttered streets.

At a point halfway though the city your team catches a break as your vehicle comes upon the white SUV you've been pursuing for the past 20 minutes, and it's been slammed into the corner of a building in a failed attempt to take a sharp turn way too fast. The squad jumps out before the HUMVEE comes to a complete stop. The SUV is found crumpled and abandoned. A trail of blood is found near the vehicle, so at least one of the perpetrators are injured, but the trails dies quickly. The squad leader is about to devise a search plan for the immediate vicinity when one of the locals shout across the street to you, and point to a bullet riddled abandoned building nearby. Half the windows are barricaded shut while the other half are blown out. You know this is a bad situation, so what do you do? You pull out a handful of disposable mapping UAVs and throw them into the air, and set their destination for the abandoned building. They autonomously split up and fly in through the dilapidated front entrance and a broken side window. Covering all the exits, your team waits a few moments as a map of the building is slowly completed on your tablet. Once completed, the

squad leader can immediately identify likely hideouts and potential kill zones based on the recovered mapping data.

Current 3D mapping technology is both expensive and large, which seems to trend towards the adage that more is better. The 3D scanning LiDAR sensor by Velodyne is an effective and robust 3D mapping tool but the sensor alone weighs almost 30lbs. Smaller, and more compact 2D scanning LiDAR sensors can weight as little at 1% of this, at 0.30lbs but the number of measurements are significantly reduced from 2.2 million data points to just over one thousand data points. Current research has implemented the use of two of these smaller 2D LiDAR sensors to effectively map 3D environments by positioning them in differing orientations such as in [5], and [14].

This research presents the concept of taking only one small scale scanning LiDAR sensor, developing a reflective structure to be mounted onto the LiDAR sensor in such a fashion to deflect the scanning laser beam out of the originally intended single layer, 2D plane. From here, a sensor characterization and a calibration algorithm to map the resulting data output into 3D coordinate space is proposed and evaluated, which is the foundation to build the solution of scenario above.

1.2 Scope of Research

This research will be performed on data that is simulated via MATLAB® and on data that is produced by a VICON motion capture system, and a Hokuyo UST-20LX scanning laser rangefinder modified by a 3D printed structure with mirrored surfaces. While the general calibration algorithm is intended to be universal for application to any 2D scanning laser rangefinder modified with external modification, the characterization performed as part of the calibration procedure in this research is specific to the Hokuyo sensor, the 3D printed structure, and target combination presented therein.

Although a key part of this research is the documentation and examination of the pro-

posed calibration process, the effectiveness and design of a reflective superstructure is only described sufficiently to identify potential issues during the calibration procedure. An evaluation of the optimal design of a reflective structure is beyond the scope of this research.

1.3 Thesis Overview

The following section outlines the remaining chapters of this thesis. Chapter 2 presents a background in relevant optical theory, then follows through to a description of LiDAR systems to include ranging algorithms, and current calibration methods for LiDAR systems. Chapter 2 finishes off with an overview on vision aided navigation and mapping techniques. Chapter 3 proceeds through the research methodology, the descriptions, and goals of each experimental setup. Chapter 4 details the results of the series of experiments which lead to the final results of the proposed calibration algorithm for converting a 2D scanning laser rangefinder into a 3D scanning LiDAR sensor. Chapter 5 presents the final conclusions and the suggestions for future work.

II. Background

A primary component used in the investigation within this document is the LiDAR sensor. To understand how a LiDAR sensor works one must understand the basics of the two main schools of thought, geometric optics and wave optics, and the benefits and shortcoming of each approach when describing optical systems. Additionally, a small introduction to three dimensional object registration and world mapping with respect to robotics will be presented.

The first section of this chapter describes the two different methodologies to represent an optical system, through geometric optics and wave optics. It will be important to understand both wave and geometric optics, as LiDAR sensors are generally understood through the theory of geometric optics and the product of ray tracing whereas the sources of noise are more easily understood when the theory of geometric optics is expanded to include those relevant components of wave optics. This is followed by walk-through in the foundations of LIDAR systems that covers the basic principals of LiDAR operation, noise sources, and then ranging techniques. Once ranging techniques are covered, a discussion on current calibration techniques used to identify extrinsic parameters of the system is presented, followed by some current LiDAR beam steering technologies. As a comparison to LiDAR systems, a description of a Red-Green-Blue-Depth (RGB-D) Camera system is covered. Next, this chapter discusses point-cloud registration algorithms and more recent methods to account for noisy data in those resultant points clouds. Finally, this chapter covers techniques used to construct world maps from inputs gathered by sensors that collect range data such as the LIDAR and RGB-D sensors.

2.1 Introduction to Optical Propagation Theory

Wave-Optics.

Wave-Optics: Light has properties of both waves and particles [15]. One tends to use the property that is easiest to implement in theory based on the application being theorized. As a visual representation of wave-optics Figure [1] is presented which shows a simple diagram of a plane wave propagating through a spherical lens to a focal point to produce an image on the receiving plane which could be the photodetector array of a digital camera or the silver film from a more classical camera.

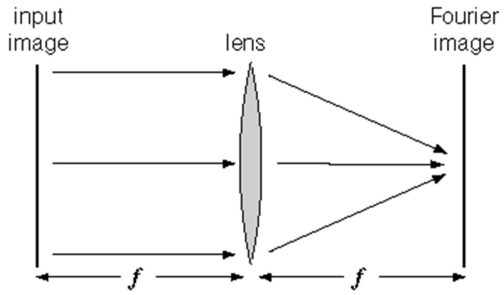


Figure 1. Plane Wave Propagating Through a Lens [1]

Figure 2. Wave Pattern of Two Waves in Water [2]

Wave interference as illustrated in Figure 2 is a phenomenon that occurs when two waves of the same type such as sound, gravity, or electromagnetic, meet while being propagated along the same medium. The interference of waves causes the medium to take on a shape that results from the net effect of the two individual waves upon the particles of the medium. Wave interference can be constructive or destructive in nature. Constructive interference occurs at any location along the medium where the two interfering waves have a displacement in the same direction. For example, if at a given instant in time and location along the medium, the crest of one wave meets the crest of a second wave, they will inter-

fere in such a manner as to produce a "super-crest." Similarly, the interference of a trough and a trough interfere constructively to produce a "super-trough". Destructive interference occurs at any location along the medium where the two interfering waves have a displacement in the opposite direction. For example, the interference of a crest with a trough of equal magnitudes that produces a moment of zero magnitude is an example of destructive interference. Destructive interference has the tendency to decrease the resulting amount of displacement of the medium [16].

Propagation delay is caused by the varying densities of the medium that the wave is traveling through. This research is primarily concerned with visible to infra-red light and as a subset of the electromagnetic spectrum, it will experience a propagation delay when traveling through a propagation path that is not uniform in material. An interesting example includes silicon crystals, which are currently being used to bend and guide x-rays and even gamma-rays in experiments that will evolve into gamma-ray telescopes [17][18][19]. Figure 1, shown previously, is a good example of why we are interested in propagation delay. The light from a point source at infinity will seem like a plane wave as it lands in the lens which is of a different material, nominally glass for visible light, than what the initial propagation went through, which can be air or a near vacuum if coming from space. When the light wave hits the glass lens it is not uniform because of the spherical nature of the lens. The symmetry of the lens causes a phase delay that is also symmetric as the light begins to propagate through the glass at a slower speed due to the density change. The differences in thickness along the lens is proportional to the length of time that the light propagation is slowed down as it travels through the lens. This delay, when coupled with the lens angles, causes the light to focus in to a focal point instead of traveling as a plane wave to infinity. The reader is directed to [1], [15], and [16] for a more comprehensive look into the effects of propagation through a non-perfect lens which also identifies lens aberrations such as tilt, and stigma, through the calculation of Zernike polynomials.

To propagate the beam through the air there are two routes we can take: an analytic method and an estimation method using a Fast Fourier Transform (FFT). The analytic approach uses the standard Rayleigh-Sommerfield Propagation also known as the Huygens-Fresnel Principle[15]:

$$\text{U(x,y)} = \frac{z}{j\lambda} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} U(f_x, f_y) \frac{e^{\frac{2\pi\sqrt{z^2+(x-f_x)^2+(y-f_y)^2}}{\lambda}}}{z^2+(x-f_x)^2+(y-f_y)^2} df_x df_y \tag{1}$$

where the double integral is with respect to the all spatial frequency combinations of f_x and f_y. λ is the wavelength of light being propagated, $U(f_x, f_y)$ is the phase of the light source with respect to the spatial frequencies f_x and f_y. The components of x, y, and z are coordinates where the light will land with the assumption that z is much larger than x and y. This results in the function $U(x, y)$, which describes the phase plane at a propagation distance, z, from the source where z = 0;

The basic principles in wave-optics that have evolved from the theory behind Rayleigh-Sommerfield Propagation into simple formulas and equations, within qualifying scenarios, have walked us down a path to a less computationally demanding way to simulate light, geometric optics.

Geometric-Optics.

Geometric Optics: also known as ray-optics, is the practice of using light as rays in describing the propagation through mediums, and the interactions thereof. Light does not actually propagate as a ray though the air, or any other medium, so the real justification for treating light as a ray is that, like other successful models in physics, rays are indispensable to our thinking. The ray is a model that works well in some cases and not at all in others, and light is necessarily thought about in terms of rays, scalar waves, electromagnetic waves, and with quantum physics [20]. Ray tracing in geometric optics can be summed up in three

components; the Law of Reflection, Snell's Law, and the Lens Makers Equation. The combination of these three elements form the elementary approach for most Ray Optics descriptions.

The Law of Reflections simply states that the angle of incidence equals the angle of reflection. In other words, the incoming light ray hitting a flat surface can be described by the angle from the vector normal of that surface, called the angle of incidence. The light is reflected to an angle equal to the angle of incidence but opposite in direction, which is directly across from the vector normal to the surface, as illustrated in Figure 3. Figure 3 uses θ_0 as the angle of incidence described from the vector normal of the surface.

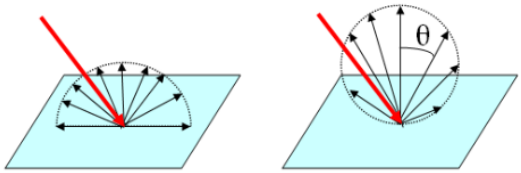


Figure 3. Reflection surface model of a flat surface. θ_0 is the reflection angle from the normal vector of surface, and the reflected intensity is proportional to the $sin(\theta_0)$[3]

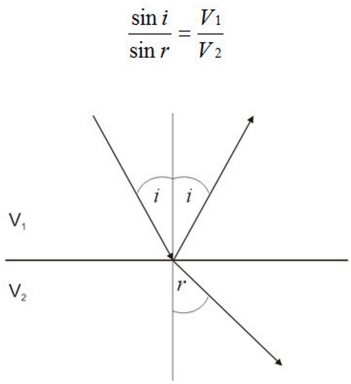


Figure 4. Example illustration Snell's Law between two transparent mediums such as glass and air[4]

The next component of Ray Optics is Snell's Law. If you can imagine the idea of the Law of Reflection, Snell's Law takes it one step further and applies it to a surface that is not mirrored but is now at least partially transparent at the wavelength of interest. Added to the description is the caveat of "the wavelength of interest" because different materials propagate electromagnetic waves differently. Lead, for example, will not allow any visible light to pass through but would act as a lens for gamma rays [19]. Figure [4] illustrates Snell's Law and shows the path taken by a ray of light where v_1 is the index of refraction for the medium the ray is traveling from, and v_2 is the index of refraction of the medium

the ray is traveling to. Equation 3 shows how to calculate the index of refraction using $v_1 = n_1$ and $v_2 = n_2$. As a reference the standard value for the index of refraction in air of $n_{air} = 1$[21].

$$n_1 \sin\theta_1 = n_2 \sin\theta_2 \tag{2}$$

$$\text{n} = \frac{c}{v} = \frac{velocity\ of\ light\ in\ a\ vacuum}{velocity\ of\ light\ in\ the\ medium} \tag{3}$$

where:

$$\text{v} = \frac{c_i}{\lambda} = \frac{speed\ of\ light\ in\ the\ medium}{wavelength\ of\ light} \tag{4}$$

2.2 LIDAR Systems and Calibration Techniques

Laser Scanning Radar or LIDAR sensors are common place in the the navigation and mapping world. There are many commercial-off-the-shelf (COTS) LiDAR products for many different applications ranging from hand-held 3D mapping sensors[22] to large 3D sensors for autonomous driving cars [13] to even more sophisticated space-based LIDAR sensors used to monitor the health of crops across nations[23].

LIDAR System Description.

LASER is an acronym for Light Amplification by Stimulated Emission of Radiation. A LASER is created when the electrons in atoms in special glasses, crystals, or gases absorb energy from an electrical current or another laser and become excited. The excited electrons move from a lower-energy orbit to a higher-energy orbit around the atoms nucleus. When they return to their normal or ground state, the electrons emit photons (particles of light). These photons are all at the same wavelength and are coherent, meaning the crests and troughs of the light waves are all in lockstep. In contrast, ordinary visible light comprises multiple wavelengths and is not coherent. Laser light is different from normal light in

other ways as well. First, its light contains only one wavelength (one specific color). The particular wavelength of light is determined by the amount of energy released when the excited electron drops to a lower orbit, which is dependent on the medium. Second, laser light is directional. Whereas a laser generates a very tight beam, a flashlight produces light that is diffuse. Because laser light is coherent, it stays focused for vast distances, even to the moon and back[24].

A LiDAR sensor's main driver is a pumped diode LASER cavity, in which the goal is to excite as many photons as possible[25]. The material, and the corresponding energy level governed by that material used to excite photons, dictates the wavelength the the LASER operates on. Atomic structures are inherently random in construction (unless we are talking about perfect crystalline structures, and we are not in this case). The next step involves the mirror used to pump the LASER diode by reflecting the photons back onto the LASER diode structure. The mirror placement cannot be perfectly perpendicular to the direction of the outgoing traveling wave and therefore induces another component of randomness into the system based on that imperfect alignment, small aberrations in the mirror and vibrations. Once excited the photons travel away from the sensor through a time-gate which blocks and unblocks the outbound light to produce a pulse or series of pulses. The timing of the range gate may be accurate to within milliseconds but the wavelength of near visible or infra-red light is much shorter than a millisecond. For example, green light at 550nm has a frequency of about 1×10^{14}Hz. This can introduce another source of randomness in the phase because the cutoff of the light. Due to the gate opening and closing, the gate can not be timed precisely with the phase of the beam. In addition, as the electronics vary in temperature (either hot or cold) the electron flow in the circuitry is varied slightly as well which will affect the time-gate timing as well. In summary, Figure 5 illustrates a simple LiDAR's laser propagator.

After the photons break free of the sensor they travel as a Gaussian beam[26], towards

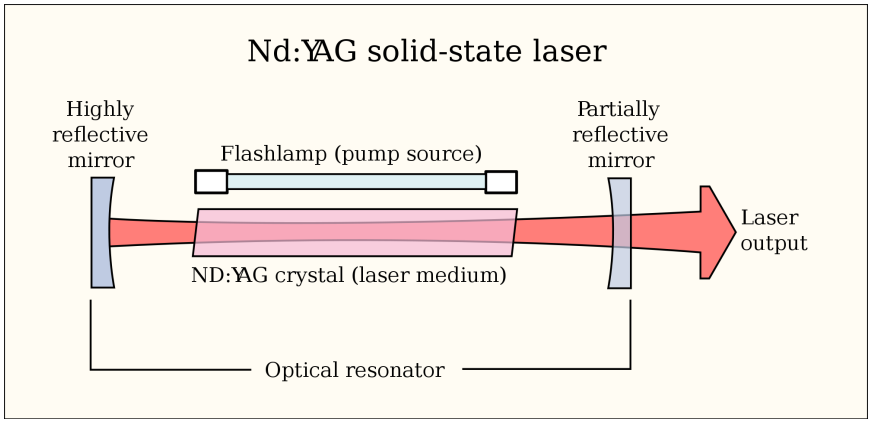


Figure 5. Simple Diagram of a Laser Source [2]

an obstacle through a propagation medium. Ideally the medium would be a vacuum and therefore would not have to account for any losses. However, typical propagation media are not in a vacuum but instead are made of a gas, liquid, or a solid, with standard room temperature air as probably most common. Since air is not motionless nor is it always composed of the exact same molecular heterogeneous mixture, we cannot know the exact make up of the air prior to measuring it. This causes another another type of randomness as the photons interact with the particle in the air, causing random phase changes and intensity changes based on the density, velocity, and size of the particles [16].

Once the propagation to an object is complete we look at the random nature of the object that is being ranged. Firstly, there is no guarantee that the object will exist, in the case of mapping indoor or outdoor environments, and there is no certainty of the actual distance to object from the sensor. This distance affects the expected range result of the sensor because the intensity of the received LiDAR signal is inversely proportional to the distance traveled. In addition, only a percentage of the beam being propagated lands on the object being ranged depending on the size of the object, distance to the object, and the beam divergence angle. Carry this forward and then taking into account the propagation back to the sensor's detector, and similarly only a portion of the reflected power of the

target reflects back to the sensor as described in Equation 5 as simplified by [27].

$$P_{detector} = \frac{\tau_o \tau_a^2 D_R^2 \rho_t (dA) P_t}{R^2 \theta_R \left(\theta_d R\right)^2} \tag{5}$$

where τ_a = atmospheric transmission rate, τ_o is the transmission rate of the optics, D_R is the receiving aperture diameter of the detector, ρ_t = target surface reflectivity, P_t = Laser transmitted power, θ_d = laser transmitted beam angular divergence, θ_R = the target surface angular dispersion. dA is the smaller of the angular area of the target, area of the field of view of the sensor or the area of the beam on the target, and R is the distance between the source and the target.

These considerations combine to greatly effect the expected returns received by the sensor, which is entirely scene dependent. An additional consideration of the object is the surface texture of said object being sensed by the ranging sensor. Surface texture will affect both the phase of the light reflecting off of the object as well as the reflectance angles. For example, as the most extreme case a mirror does not reflect in all directions but is based on the angle of incidence. If a LASER source is pointed towards a perfect mirror, you will not see any spot on the reflecting surface to indicate that the LASER landed on the mirror. The lack of roughness of the surface, and color of the surface will have minimal effect on the received signal level of the detector as seen by Equation 6 and Figure 3 but will only depend only on the angle of incidence.

$$P_r(\theta_0) = P_i cos(\theta_s) \tag{6}$$

where θ_0 is the look angle to the sensor between the target surface normal vector, P_r is the reflected power directed towards the sensor at angle θ_r, and P_i is the peak reflected power when $\theta_0 = 0$. Once a rough surface is introduced, the reflected power is now also a function of both the azimuth and elevation of the incident angles, θ_i, and ϕ_i, and of both the azimuth

and elevation of the reflected angles, θ_r, and ϕ_r [28] as seen in Equation 7 to account for the surface features, the dependence on incident direction, and the location of the detector.

$$P_r(\theta_r, \phi_r) = \int_h \rho(\theta_T, \phi_T) P_i(\theta_i, \phi_i) d\Omega \tag{7}$$

where h represent the surface integral across the incremental changes in steradians, $d\Omega$, which is comprised of both the azimuth and elevation of the incident angles, θ_i, and ϕ_i. $\rho(\theta_T, \phi_T)$ represents a known function describing the reflectivity pattern of a surface whose reflectivity is not perfectly specular or lambertian but depends on both the angles of incidence, θ_i, and ϕ_i and angles of reflectance, θ_r, and ϕ_r. When expanded to show all the relevant variables, $\rho(\theta_T, \phi_T)$ can be described as $\rho(\theta_i, \phi_i, \theta_r, \phi_r)$, where θ_T represents the azimuth variables θ_i, and θ_r, while ϕ_T represents the elevation variables ϕ_i, and ϕ_r.

The next important consideration that is random in nature is the local illumination of the object. The illumination is based on the lighting condition of the local environment, and the sensor relies on the fact that the local area illumination is not expected to be larger in amplitude than the source at the source's wavelength. This illumination is highly dependent on the scene, time of day, and many other unknown variables which make it practically impossible to calculate and is needed to be characterized as a random process.

LiDAR System Types.

A common LiDAR sensor, based on a linear mode design, uses a series of pulses sent in predetermined directions defined by a system controller or through hardware design. The intensity of the return signal for each pulse is received and monitored by the LiDAR sensor and once the intensity meets a given threshold value (predetermined in hardware design) the time is calculated and a corresponding range is calculated using Equation 8.

The expected operational range of the sensor will determine how much time the sensor will wait to listen for a return signal. Using a rule of thumb such that the speed of light is 3x10^8 meters per second [29] one can use the distance equation shown in Equation 8 to calculate the anticipated time of flight (TOF) of the signal:

$$\mathrm{t} = \frac{2R}{c} \tag{8}$$

where R is the distance to the target, t is the time it takes for light to travel to the target, and c is the speed of light. The distance is doubled to account for traveling the same distance twice to get to the target and back. As an example, an indoor sensor such as one offered by Hokuyo [30] with a maximum range of 60m may only allow for 400ns to listen for a return pulse before sending out the next pulse in the series. There are many parameter considerations to keep in mind such as the how the wavelength of light is affected by a propagation medium, as discussed in the Section 2.2, noise considerations in the environment, and the expected distance to be ranged which is related to the divergence angle of the LiDAR beam. Some of the simpler systems discussed are designed through the use of ray geometries while other systems such as a Spatial Light Modulator use the properties of Wave theory as it pertains to optics in order to pull out the relevant ranging data from what the sensor detects.

There are essentially four types of LIDAR system designs relative to navigation operations; Linear Mode [31], Flash LiDAR systems [32], Spatial Light Modulator (SLM) designs [33], and a Geiger-Mode [34] designs.

In the category under linear-mode LIDAR systems there are a handful of different ways to take advantage of the linear operations by using one or a few laser sources to range the surrounding environment. The first benefit is that the processing and hardware requirements are minimal when compared to the other system designs. The most basic design in this category uses a single source and through the use of an optical system the beam

is swept across a given field of view (FOV) [33]. The FOV of the LiDAR sensor can be designed to give a larger or smaller FOV by manipulating the optical system. The downside is that the sensor can only handle one timing pulse at a time and this pulse rate (in Hertz) is inherently limited by the quality of the electronics and the on/off properties of the output gate controlling the pulse duration. Because of this limitation, the cost due to processing and power requirements of imaging a dense point cloud grows exponentially with increasingly dense data collects until the cost of the sensor collection is too large to be implemented on small platforms. Another disadvantage is that this type of LIDAR sensor has some moving components because of the optical system used for beam steering, increasing the probability of failure over a solid state design. Additional research include prototype designs proposed by Hokoyu [35] which improves upon the small scale LiDAR sensor design by increasing the vertical field of view of a single beam horizontally scanning sensor to 40 degrees. Most large FOV sensors are designed for larger applications such as Autonomous Road vehicles and self-driving cars [36] which make this sensor interesting in small scale applications where those larger sensors either just cant fit or are too heavy for the application.

The next LiDAR design considered is Flash LiDAR. Flash LiDAR is very similar in operation to a 2D digital camera [32]. A 2D plane of photodiodes measures an initial pulse in all directions in the FOV set off by the sensor which marks a zero-time state. Then timing is calculated similarly to a linear sweeping laser scanner or the traditional LIDAR sensor but across all the sensor diodes at the same time. The main benefit of using Flash LiDAR is that system can be smaller than a comparable stereo-vision system, and because the calculations involved in constructing a 3D environment are similar to traditional LiDAR. A potential disadvantage is that the technology is not mature enough to allow for widespread applications let alone an application requiring a small scale package.

The third major class of LiDAR systems is one that uses SLMs to illuminate the scene,

and then use Fourier Transform Heterodyne (FTH) techniques [37] to reconstruct a 3D point cloud. The LIDAR source illuminates the target via a continuous wave, and the LiDAR sensor component in the system receives the reflected LIDAR signal just like in the traditional LiDAR design. The difference is that the source illuminates the target scene continuously and is compared to a reference source being manipulated by the SLM. The effect of the SLM is to vary the phase of the light which causes fringe patterns on the detector array. The description of the resulting fringe patterns is used to calculate the relative ranges of each point to each other. A major design difference between this and the three other methods is that it is based off of wave optics principles, treating light as waves instead of rays. While this type of LiDAR system has no moving parts and only a single detector, it must make approximations of the scene geometry in order to apply the FTH algorithms[38].

The last type of system design is Geiger mode LIDAR. This could be consider the next evolution of Flash LIDAR. An example of a Geiger Mode system being designed by Harris Corp [34] has a system design that uses an array of 4,096 avalanche photodiodes (APD) and achieves a timing pulse width of less than 1ns which corresponds to a possible 50,000 pulses per second being simultaneously broadcast across all 4,096 APDs. This translates to an advertised measurement rate of 205 million measurements per second in the target area. The first thought when seeing these numbers is how much processing needs to be done to handle 205 million points of data per second. Given this consideration, the Geiger mode LiDAR sensor is not suitable for small scale applications, but is more suitable for large satellite or airborne systems designed for imaging large areas, post-processing the data after the mission.

LiDAR Data Noise Sources .

In addition to the random nature of environmental and system level factors stated previously there exist several intrinsic sources of noise within a LiDAR system that can affect the returned range measurements. These factors may induce range error, and some types of noise sources which are internal to the system may be compensated for whereas noise sources external to the system may not be. Some of these noise sources include photon counting noise, laser speckle noise, thermal noise, and detector background noise [27].

Photon counting noise, also considered as shot noise, describes the error caused by the fact that the arrival of photons happen at random times, which introduce uncertainty in the photon measurements by the photodetector during a finite integration interval, δt, used to take an individual measurement. The detector will have a consistent range gate time equal to δt, but the time of flight for each photon is not the same. The number of photons collected by the detector can be modeled as a Poisson random variable.

The next source of noise is speckle noise. This effect is caused by the constructive interference at the detector. Each photon can arrive at the photodetector with a random phase, and the phase of each photon will interact with the phase of all the other photons arriving at the same time either constructively of destructively [1]. Speckle occurs when the photon phases add up to much greater than an expected returned intensity value. Speckle occurs more frequently when the propagated laser beam is coherent, or when the beam is propagated in phase as a whole, and the phase is much more likely to have an effect of the detector.

Another important internal noise source is thermal noise of the system. This noise is cause by electron flow in the photodetector due to the temperature. The electron flow produced by the thermal noise cause what's known as dark current, which is the current flow through the photodetector when no light is present. In an ideal situation there would be no current flow when no light is present. The dark current can add a bias to the photodetector

measurements.

Background is also a major noise source. Background noise includes and light collected by the sensor that does not originate from the designed laser source. This can include ambient light from the environment such as sunlight, or other environmental sources that emit an electromagnetic signal at near the laser beam frequency. The photons collected by the photodetector due to background noise adds no information to the measurement, but may hinder the measurement precision when the background noise is large enough. This phenomenon is also Poisson in nature.

The next noise source, known as glint [39], is similar to speckle but does not depend on the phase, or the coherency of the propagating laser beam, but the end result is similar in that the range measurement is effected. This noise is due to the surface texture of the surface that the beam is reflecting off. The random nature of a rough surface causes a random subset of that surface to be perpendicular to the beam, and as seen in equation 6, when the of the angle between the normal vector of the surface and the beam are zero degrees the portion of the beam that landed on the surface will entirely reflect back towards the detector. For porous, or matte surfaces there are numerous amounts of surface normal vectors and as the roughness increase the reflection becomes more and more representative of a lambertian surface [40]. When a highly reflective surface, such as polished metal or glass, has only a slight roughness there exists a potential for a strong glint noise to return.

LIDAR Ranging Techniques.

Taking a closer look at linear mode, also known as direct detection, LiDAR systems we will look at some of the detection methods based on TOF. Estimating the t in Equation 8 is where multiple different approaches can be taken. Three basic designs will be considered here: threshold circuits, envelope detectors, and a matched filter receiver[41]. All of these techniques rely on photon counting detectors in the receiving optics and therefore suffer

from the noise sources discussed in previous sections.

The first method, and arguably the easiest to implement, are the threshold circuits. This time of flight estimator compares the receiving LiDAR signal to a given threshold value predetermined via internal circuity, and once the signal threshold is reached the corresponding time, t, between the transmitting pulse time and the time the threshold is breached, is used to estimate the range. The threshold detector described here is also considered a leading-edge detector,[27] due to the fact that the leading edge of the LiDAR return signal is what crosses the threshold first. There are inherent sensitivities to speckle, glint, background noise, and return pulse shape when relying on a threshold circuit for detection. These types of effects can distort the received LiDAR signal level or present spikes or peaks in the return which artificially trigger the the threshold circuit, causing a false range return. Figure 6 shows a simulated, and noiseless, LiDAR return in blue, when two targets in black are being ranged against. An example threshold is highlighted in red, which illustrates the potential error in t due to using a threshold detector to calculate time of flight. Figure 6 shows a simulated return signal as compared to a noisy return signal in red. These illustration assumes that the time resolution of the detector is small enough to show a wave form in the return. To overcome for this an envelope detector can be used.

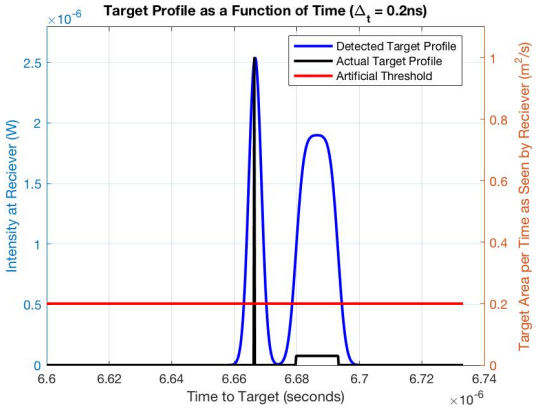


Figure 6. Range Return Example

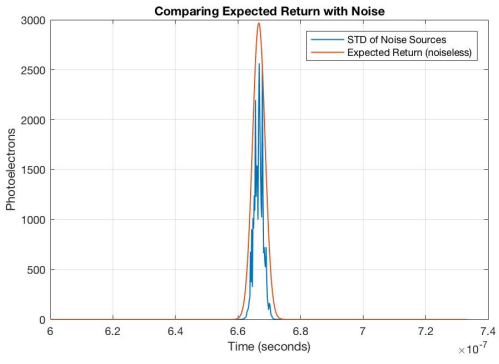


Figure 7. Comparison on Expected Return with a Noisy Return

An envelope detector looks at the receiving LiDAR signal interpolates the overall shape

of the receiving amplitude during the expected time of a received signal, as determined by the timing gate circuity in the LiDAR system. The received signal, which in its raw form is discretized into chunks based on the integration time of the receiving photodetector. The envelope detector smooths out the discretized form of the signal, and then find the maximum value of the return. The time of flight corresponding to the peak value is returned for a range calculation. This method assumes the intensity of the return signal reflected from the target is greater than any other noise source, and can give improved results over using a threshold detector. In situations where there are increased levels of noise, and the expected range is fairly known, a matched filter receiver may be recommended[42].

The matched filter starts with a simple binary hypothesis test using H_1 as hypothesis one, in that a target is present in the, and H_0 as hypothesis two, that a target is not present. From here a probability mass function (PMF) of the probability of detect, $P(D)$, and using data gathered by the photodetector returns when no target is present a PMF of the probability detect given that no target exists, $P(D|H_0)$, can be used to describe the probability of a target given that a detect occurred, $P(H_1|D)$ and the corresponding probability of no target given that a detect occurred, $P(H_0|D)$.

$$P(H_1|D) = \frac{P(D|H_1)P(H_1)}{P(D)} \qquad (9) \qquad\qquad P(H_0|D) = 1 - P(H_1|D) \qquad (10)$$

Using equations 9 and 10, and the assumption that $P(D|H_1) > P(D|H_0)$ indicating that H_1 is true we can derive a likelihood ratio test (LRT) in equation 11 for whether a target is present for a given amount of received photons on the photodetector. A simulation using the LRT can be seen in 8 and a run using the results of a Monte Carlo simulation for 1000 random trials in each case, H_0, and H_1.

$$\Lambda = \frac{ln(P(D|H_1))}{ln(P(D|H_0))} \qquad (11)$$

where Λ is the resulting ratio result and that a return greater than one indicates that no target

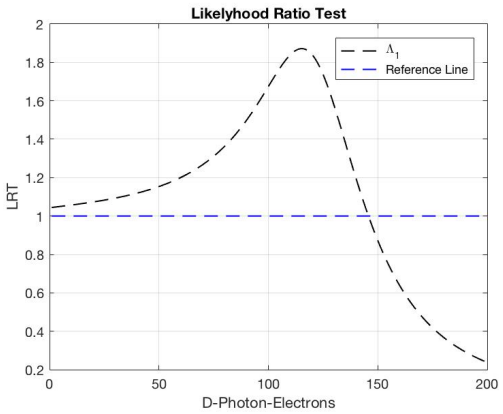


Figure 8. Likelihood Ratio Test Example

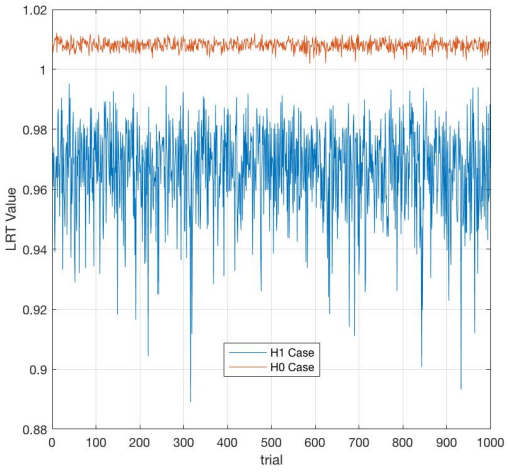


Figure 9. LRT Example Across 1000 Trials

is present.

Using the LRT at each time increment of the returned LiDAR signal allows for a waveform result similar to the raw data but is instead a plot of the likelihood of that a target is present given the amount of photons received at that time increment. A peak detector is then used to identify the maximum of the waveform to identify the estimated t.

LIDAR Calibration Techniques.

LiDAR sensor calibration is an important element of sensor operation and the importance of this seemingly simple task is doubly important as LiDAR sensors are typically deemed a critical sensor in robotics and autonomous UAV applications. Out-of-the-box, LiDAR sensors are designed to measure ranging information and provide this to the user, but the information being displayed to the user may not be the true range. The inherent noise of a LiDAR sensor can cause unexpected variations greater than 10cm[43] in the range measurements, which may be larger than acceptable for indoor navigation and mapping applications. It is therefore necessary to characterize the sensor before real-world use. Part of this characterization process will shed light on the expected error between measured ranges with the true range, object resolution at the target range as a function of angular res-

olution, and the variance and standard deviation of those measurements over time. Once this data is collected, and the internal but measurable error sources are identified, then the extrinsic calibration of the sensor can take place. If one had the opportunity to perform an additional characterization step, and had access to the internal electronics of the LiDAR sensor of choice, an intrinsic calibration would be done before any other characterization tests. This type of characterization would typically occur at a production facility for the commercially available sensors used in mobile robotics but an example using a continuous wave laser by Adams can be seen in [44].

Okubo published a characterization [43] of a similar 2D laser rangefinder to the one used in the research presented in this thesis. As a contrast, Muhammad performed a smaller characterization and calibration of a multi-beam LiDAR sensor [45]. The Hokuyo URG-04LX was characterized by Okubo in an effort to determine suitability for mobile robotics applications over larger and more expensive LiDAR sensors such as the LMS 200 produced by SICK Inc[46]. Okubo attempted to measure the transfer rate of the sensor output, as well as the effect of drift, surface properties, and incident angles to the sensor measurements. It was found that shiny targets such as aluminum and gold caused a variation in measured ranges between 5-9cm whereas matte colors and grays caused a lower variation between 2-2.5cm. Another important aspect of the paper identified an error known as mixed pixels in the range measurements. A mixed pixel is the result of the laser beam spot landing on the edge of the target. The measured range can then become a combination of the foreground and the background, and the resultant ranged returned is between those distances. The last key take away from Okubo's paper is that due to the nature of the returned measurements, it is necessary to use statistical models to map the environment using this raw LiDAR data. [47]

After the characterization is done, the extrinsic calibration can be performed. To better visualize and clarify the concepts of extrinsic LIDAR calibration it is necessary to intro-

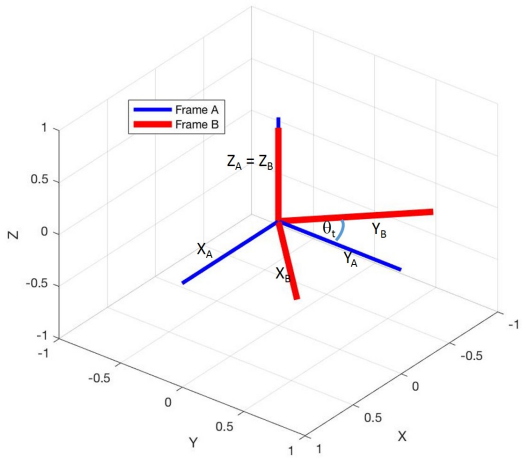


Figure 10. Example Frame Rotation by θ_t

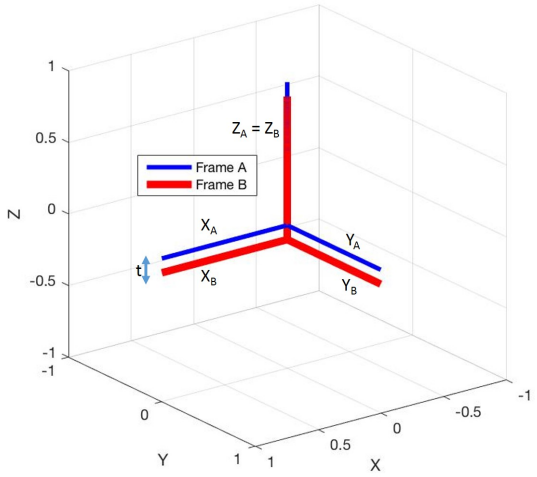


Figure 11. Example Frame Translation by t

duce the concepts of navigation frames. Navigation frames are used to identify location relationships between the origin of the coordinate systems used to describe the world, the data the sensors are seeing, and the vehicle(s) being used to navigate that world utilizing those sensors. An example transformation matrix to go from the world origin frame, Frame A, to a robot base frame, Frame B, via a rotation about the z-axis only is shown in equation 12 and Figure 10 while an example transformation matrix to go from the world origin frame to a robot base frame via a translation along the z-axis is shown in equation 13 and Figure 11. where θ_z is the rotation about the z-axis.

$$T_o = \begin{bmatrix} cos(\theta_z) & -sin(\theta_z) & 0 & 0 \\ sin(\theta_z) & cos(\theta_z) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12) \qquad T_o = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$

Each set of corresponding frames will have a transformation matrix associated with it which can be stacked with multiple components. For example, if we wanted to express the range information of a point in the sensor frame, P_s, to the world origin frame it would be computed by transforming P_s first to the robot frame via a transformation matrix between

the sensor frame to the robot, T_S^R. Next a transformation from the robot frame to the world frame would be be performed via T_R^W. To perform this procedure in a single step it's possible to do so using:

$$P_w = T_R^W T_S^R P_s \tag{14}$$

The extrinsic calibration is used to calibrate the extrinsic parameters that relates the external sensors, such as LiDAR, to each other, and with the navigation frame of the body attached to the sensors. This is an important step in order to use the sensor measurements for navigation and mapping. Extrinsic calibration has been covered extensively and is primarily done through several different general methods. One method takes multiple measurements of a designed target within a scene using a single sensor as done by Choi[5], and Chen[48], or within a natural environment as performed by [49]. Single scan 2D scanning LiDAR calibration techniques vary from author to author and can be boiled down to either relying on artificial or natural landmarks or on a combination of one planar surface or many planar surfaces such as the approach proposed by Choi which utilizes the idea of using two orthogonal planes in the scene to calibrate the LiDAR sensor as seen in Figure 12.

The sensor takes a scan in one frame, is moved to a given position and another scan is taken again which should give a different set of range values. Using only the scan data and the fact that the target planes are orthogonal provides a linear constraint to the unknown vector due to coplanarity. The extrinsic parameters are then extracted from the unknown vector via the essential matrix[50] method in [5], and a least-squares technique is applied to minimize the error when the two planes are not exactly orthogonal to each other.

Another calibration method employs multiple 2D LiDAR sensors within a known environment such as the those done by Brookshire[14], and Quenzel[51]. A specific type of multi-sensor calibration utilizing the combination of LiDAR with a camera, which is a different method due to the nature of the visual imagery of the camera, is presented by Debattisti[52], Pandey[53], Huang[54], And Guo[55]. The end goal of performing the re-

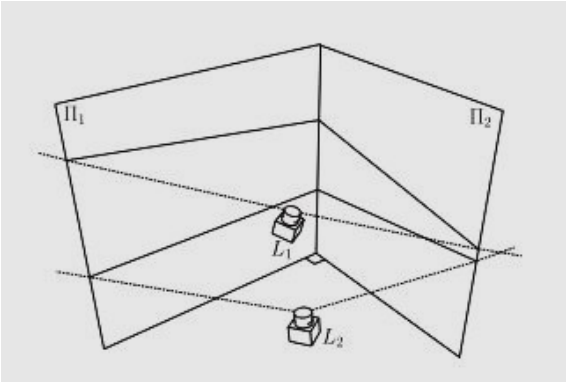


Figure 12. Setup of Proposed Extrinsic Calibration Method by Choi[5]

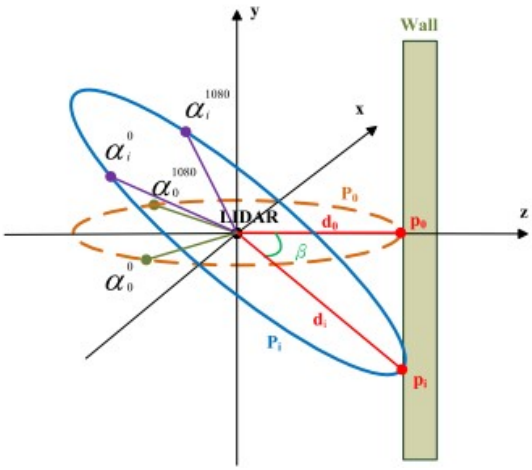


Figure 13. Ideal Multi-Layer Scanning Planes by Lin[6]

quired calibration is to sense, and reliably use those sensed measurements to create a 3D world. Creating a three dimensional model in a naturally noisy environment is complicated and an important issue to tackle. But in the interest of attempting to do more with less, a calibration algorithm limited to exploiting a single sensor is more beneficial.

Muhammad in [45], proposes a method of using twelve scans from a single sensor to reduce the measurement error, the algorithm proposed in [56] uses four calibration boards surrounding the LiDAR sensor and one underneath to calibrate the intrinsic and extrinsic parameters. For multi-layer LiDAR the approach can be conducted as published in the Velodyne GDL-64E S2 manual, while Chien-Chou Lin presents a calibration method for extending single layer 2D LiDAR to multi-layer LiDAR in [6]. Lin attempts to solve the calibration problem when using a single 2D scanning LiDAR sensor operating at different planes as a pan-tilt mechanism is maneuvered in a controlled manner, which can be applied to a scenario utilizing multiple scans of the environment. The proposed approach used a vertical wall as a base in the scene where measurements are taken and an estimate of the radius of the circular trajectory of the sensor is generated. Measured angles in roll, pitch, and yaw are used to calculate measured distances to the actual target point on the vertical wall from a given set of various roll-pitch-yaw and x-y-z coordinates. This allows the

same point on the wall, with respect to the x-axis, to be ranged by the sensor from multiple pan-tilt orientations as can be seen in Figure 13. The estimated ranges are compared to the measured values by the LiDAR sensor to obtain a shift error description between the measured and calculated values. Once known, the shift error can be added via a transformation matrix to convert the measured data points into the world origin frame.

Small Scale LIDAR Systems.

There are a variety of small 2D scanning LIDAR sensor options at the moment. Many of the major manufacturers such as Velodyne, Hokuyo [30], and SICK[46] have developed their own commercial-quality sensors, while some start-ups and do-it-yourself (DIY) enthusiasts have proposed even lower cost solutions down to $100 for a homemade prototype.

Velodyne currently offers the Puck, and Puck Lite which weigh 830 grams, and 560 grams respectively. Both of these sensors offer a 360° horizontal FOV, 30° vertical FOV, and an angular resolution of 0.1°at up to 20Hz refresh rate. Hokuyo currently showcases more than ten 2D scanning LIDAR sensors that range in weight from a simple 160g sensor to an environmentally robust sensor weighing 1.2kg.

Other solutions such as the RPLIDAR 360 Laser Scanner[57] offers a more economical solution for less than $\$400$ which offers a plug-n-play USB capability and similar capabilities to the sensors described above. In addition to the RPLIDAR there is even a scanning LIDAR solution up on Kickstarter, a website to crowd fund business ideas, to build a scanning LIDAR sensor for $\$255$ [58].

Moving from small scale to larger systems show current technological developments in the field that are not yet implemented in small scale systems which could be utilized in UAVs. One of the key differences between large scale systems (often for use on airborne or space-borne platforms) and small scale systems is the mechanism with which to steer the LiDAR beam. Small 2D scanning LiDAR systems tend to use a mirror pivoted on a single

axis relative to the sensor beam and rotated continuously about another axis to deflect the beam to capture a 2D planar view of the world. For an application that is designed to capture ranges on the X-Y plane, the LiDAR beam is pointed directly vertical in the Z-direction (up) with a mirror orientated at 45° with respect to the beam placed in the beam path. The mirror is continuously rotated about the z-axis while the beam is perpetually deflected at 45° to be able to capture the X-Y plane. Larger systems tend to have more space with which to design the system, and therefore have more flexibility to create more advanced steering technologies. Simple designs include the use of a steering mirror that is able to rotate along 2-axis to point the single LiDAR beam to a desired location[59] within the field of view. An advancement of this technology includes the use of a set of prisms in the optical beam path which split the original LiDAR beam into multiple beams[60], and cause the beam to "fan" out in the direction of the first steering mirror. The prisms allow for a wider area to be covered in the same amount of time as a similar system without the prisms.

In 2011 NASA sponsored a collaborative effort to look into the effectiveness and demonstration of another system, The Electronically Steerable Flash LiDAR system, where they are able to combine the concept of flash LiDAR, as discussed previously, with a non-mechanical steering solution. The steering is achieved via an acousto-optic beam deflector that splits the laser into multiple beams that can be independently accessed and pointed without the need of mechanical scanners or boresight mechanisms[61]. Combining these two technologies successfully reduced the number of required moving parts needed in a spaced-based LiDAR system.

A more recent research area of beam steering involves the concept of phased array antennas and applying that theory to optical systems such as LiDAR. Figure 14 shows the concept of phased array antennas as illustrated by Yaacobi [7].

Figure 14a illustrates a linear array of antennas, which are all facing an identical direc-

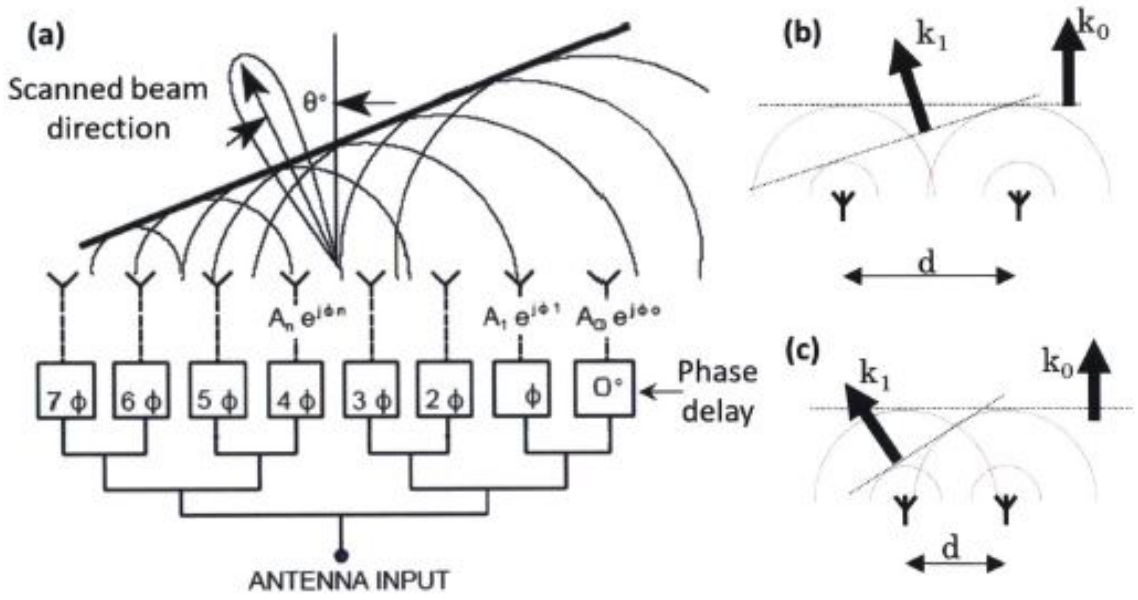


Figure 14. Optical Phased Array by [7]

tion. Each antenna propagates a coherent signal at a desired wavelength, and by inducing a phase lag in a prescribed way between the antennas allows the beam patterns to interact both constructively and destructively. The phases of each antenna react constructively to form a larger beam pattern at the desired location whereas the phases react destructively to decrease the signal level at all other locations. Both the time of phase delay and the physical distance between antennas effect the resulting beam shape. Figures 14b and 14c diagram how the distance between antennas can effect the direction of the culminating signal between the two antennas.

Overview of RGB-D Camera System Operation.

A relatively new development in cameras are the RGB-D cameras. The RGB-D camera has two components: a three-color video camera that captures in red, green, and blue and also an infra-red (IR) projector and sensor that is able to capture depth. The IR projector projects a 2D array of dots with an expected angular separation with respect to the sensor. The sensor compares a reference pattern, based on the expected angular separation, to the

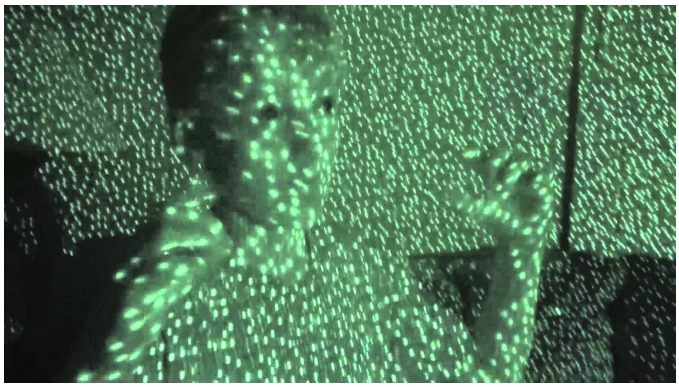


Figure 15. Illustration of Kinect's IR Dots by [8]

received IR image of the projected IR pattern. Any type of surface that differs from the reference pattern, such as the one in Figure 15, will result in pattern displacement[62]. The shifts in the pattern displacement are the basis of the depth calculations. This combination of sensors allows the sensor to give both color and depth in a more economical package than using both a LIDAR sensor and a vision camera. One such system is the Microsoft Kinect[63] used in the X-Box gaming station, it has a depth resolution of 2.5cm at a range of 4m [64].

The advantage of using an RGB-D camera is the amount of data that is provided in real time. Each "data cube" will consist of a range and color at each pixel for an entire array of pixels that covers the camera's FOV. At close ranges, less than 5m, this data can be very accurate but becomes less accurate as the range increases. Traditional video is able to provide "structure from motion", feature detection, and object recognition, and coupling this information with depth makes the system more capable to map out an ever changing environment.

2.3 Overview of Vision Aided Navigation

Image aided navigation is the general area concerned with utilizing electro-optical information for navigation purposes. This can include cameras in any spectrum such as the

infra-red, near infra-red, ultraviolet, or in the visual spectrum. Each spectrum has a different response to the world and comes with advantages and disadvantages in utilizing them. Visible light wavelengths between 400-700μm are not able to image through clouds, fog, or other forms of precipitation, whereas the lower wavelengths of light such as infra-red do not suffer nearly as much. Another advantage is that there is more commercial technology developed in the visual range than in any other spectrum, and therefore it's much easier to find information and current research in these areas than in other areas of the electromagnetic spectrum.

There are many different ways to use images, of all spectra, to be able to navigate through the world without the use of GPS and other more robust navigational techniques. One technique to utilize images is through the use of landmarks using machine vision concepts that allow you to pick out localized landmarks to create an internal map of the world[65]. By locating yourself with respect to the landmarks, you are able to decide where you are and therefore calculate what direction you need to go. Another technique includes a concept based off of how insects see the world. By using a handful of cheap and lightweight cameras to capture the world, although not in high-definition as we are used to, but enough of it can be seen to evaluate the proximity to walls, the floor, or another large object by being able to detect the presence and location of shadow contours associated with illumination changes in natural environments[66].

2.4 Mapping Techniques in a 3D Environment

3D Point Cloud Registration Algorithms.

Three dimensional environments as seen by a LiDAR sensor retrieve a data-set that can be reformatted into a three dimensional point cloud, or an array of measurements that describe a series of Cartesian or polar coordinates in a 3D space. This resulting set of data can be seen as a 3D figure, and using that we can treat the measured environment as an object

and therefore object tracking based algorithms can be used. Some registration algorithms that fit in this scenario are RANdom SAmple and Consensus (RANSAC), Principal Component Analysis (PCA), Singular Value Decomposition (SVD), and Iterative Closest Point (ICP) family of algorithms. ICP has many published variants to include Point-to-point, point-to-plane, non-linear, and sparse data variants[9].

The first algorithm, RANSAC, was originally proposed by Fischler and Bolles[67] in 1981 from within the computer vision community, and is designed as a general parameter estimation approach designed to deal with a large amounts of outliers. This approach assumes that the raw experimental data collected can be segregated into two categories, outliers and inliers[68]. Inliers represent the data points within the original data set that have some distribution which can be modeled by a set of model parameters, whereas the outliers are data points that do not, and can be excluded. The values of the outliers as compared to the whole data set, which cause them to become outliers, are assumed to be from extreme values of noise or from erroneous measurements[67].

Originally used by Fischler and Bolles to solve the Location Determination Problem (LDP), which is to determine the point in space from which an image is taken if given an image depicting a set of landmarks with know locations[67]. Although originally used with imagery data, this algorithm can be utilized to find the translation and rotation between two point clouds as in [69], and [70]. The basic algorithm as applied to point clouds, modified from [71], is as follows:

- Reduce the two point clouds into a set of features
- Start with a large set of potential feature correspondences between two point clouds
- Randomly select a subset of feature from the set of potential feature correspondences.
- Calculate the translation and rotation to match the potential feature correspondences.
- Apply the calculated translation and rotation to the entire point cloud.

- Determine how many remaining features fit the solution
- Repeat the previous step to exhaustion, keeping the solution that had the most matching features to the estimated parameters. This would be the consensus solution.

The broadest technique is PCA, which can be used in both classification and compression techniques to project data on a new orthonormal basis in the direction of largest variance [9]. For use as a registration algorithm the technique is as follows. The original 3D point cloud and the new 3D point cloud are calculated and centered at the same center of mass, or centroid. The covariance of each point cloud is computed, the eigenvalues are calculated for each matrix, and the two point clouds are aligned in the direction of largest eigenvalue. This can be a single step algorithm.

SVD is similar to PCA. The process of using the SVD approach also co-locates the centroids of the 3D point clouds as in PCA and calculates the covariance of each point cloud matrix. Once computed the points clouds are then simply aligned with the largest variance. This works well in data-sets with few outliers but is not as robust as the PCA algorithm described previously [9]. Equations 15, 16, and 17 show the progression of calculating the matrices to get the singular value decomposition, the corresponding rotation matrix, and the point cloud to point cloud translation respectively. where T_0 is the cross correlation matrix

$$T_0 = USV^T \qquad (15) \qquad\qquad R_1^2 = UV^T \qquad (16)$$

between the two point clouds, PC_1, and PC_2. The USV^T is the eigenvalue decomposition which results in two unitary matrices, U, and V, and a diagonal matrix, S, of the eigenvalues. From the decomposition, the rotation matrix from the first point cloud to the second point cloud is obtained by calculating R_1^2 in Equation 16 while the translation is calculated by:

$$\mathrm{t} = PC_2 - R_1^2 * PC_1 \qquad (17)$$

Although RANSAC, SVD, and PCA have their places as registration algorithms within

a world of perfect data-sets they all lack in the ability to reject more than a small number of outliers. Another method is introduced that iteratively ignores more and more outliers in order to improve upon the previous estimate of the rotation and translation matrices between 3D point clouds: the ICP Algorithm. Basic ICP starts with using the SVD algorithm to calculate the first guess of the affine transformation matrix, which is a combination of the translation and rotation matrices. Next, major outliers are identified and eliminated and then the affine transformation is calculated again. This process repeats a given number of times or until a set error tolerance is reached[72].

ICP Point-to-Point: This algorithm is the main spear of the ICP algorithm set and uses the individual points in the 3D point cloud to obtain point correspondences be searching for the nearest neighbors target point. In the end this algorithm will iterate until the solution with the closest point-to-point matches are found. This algorithm is still fairly sensitive to outliers [9].

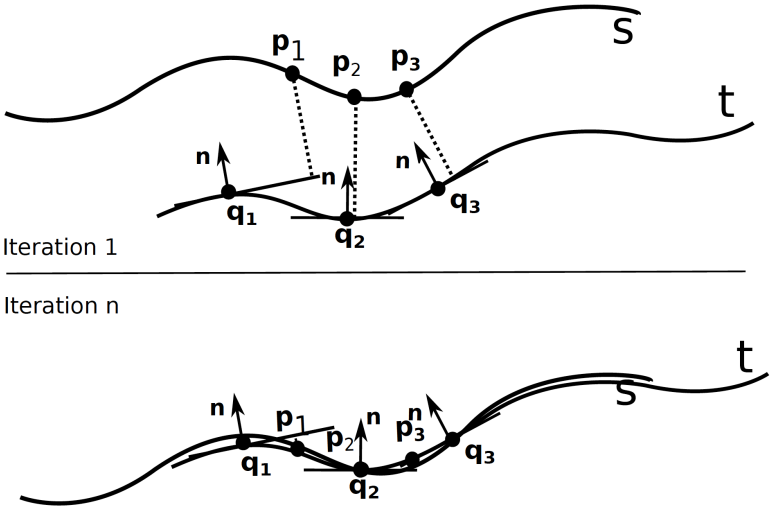


Figure 16. Example illustration of ICP point-to-plane concept using two similar lines. As the number of iterations increase the error between the two lines decrease[9].

ICP Point-to-Plane: Taking point-to-point one step further in outlier rejection is the point-to-plane algorithm[73]. The points in the first 3D point cloud are no longer being

directly matched to a specific point but to a neighborhood of points that now are assumed to collectively share a coplanar surface and a corresponding normal vector is calculated. Now the scalar projection of the error between the point to the neighborhood surface is what will be iteratively minimized. An illustration of this can be seen in figure 16.

ICP non-linear: both the point-to-point and point-to-plane algorithms assume that the minimization process can be handled like a linear least square optimization problem that can be solved numerically by SVD. In Optimization this is called L2-Optimization, and is still sensitive to outliers because the residuals are squared. An alternative to this is to use L1-Optimization which minimizes the sum of the absolute values of the residuals but is not differentiable at zero, and therefore a mix between L2-optimization and L1-optimization is recommended. This approach typically uses the Huber loss function[74] and allows the function to behave like a L2-Optimization problem for small values and like a L1-Optimization problem for large values. This combination enables the minimization algorithm to slowly converge in small error values but also rapidly converge when the error is large.

ICP Sparse: The sparse ICP algorithm is similar to the previously mentioned ICP algorithms but with a slight change to account for datasets with noisy measurements[75]. Similar to using non-linear ICP, the sparse ICP algorithm uses the Euclidean norms of the distance, but instead of using the L2 norms it is implemented by minimizing the LP norm of the residual errors. The LP norm is represented by:

$$||x||_p = (|x_1|^p + |x_2|^p + \ldots + |x_{n-1}|^p + |x_n|^p)^{1/p} \tag{18}$$

where x is a vector of length n, and p defines the norm level [76]. Common cases of the LP norm are the L2 norm where p would equal 2, which commonly referred to as the Euclidean norm. One version of sparse ICP introduced by Mavridis[77] imposes a lower penalty to large outliers. The goal of doing this is to maximize the number of zero-points between the

two objects and therefore maximizing the contact area between the two surfaces. Mavridis achieves this by using a Simulated Annealing search that is able to perform large random jumps in the parameter space and thus can approach faster to the solution, but with low accuracy. Simulated annealing is a probabilistic approach to finding a global minimum or maximum of a given function[78]. Once the error is below a given intermediate tolerance he switches to an Alternating Direction Method of Multipliers (ADMM)-based optimizer of the Sparse ICP to guarantee convergence.

World Mapping.

World Mapping is a challenging task and is continuously increasing in demand as personal and commercial navigation techniques require more and more detailed map data. On a local level, world mapping is necessary for the navigation in any environment not just to travel across the country. The release of Microsoft's Kinect sensor [63], which can provide a dense 3D map of the environment using the ranging techniques discussed prior in the LiDAR Systems Overview, has significantly increased interest in localization mapping. The term of mapping here refers to the creation of a 3D or 2D representation of the environment of interest whereas localization refers to being able to locate oneself in that environment. Mapping is necessary in autonomous navigation and is critical to trajectory planning to be able to navigate through the local environment. Within the field of robotics, mapping is traditionally performed in a Simultaneous Localization and Mapping (SLAM) algorithm which is ideally calculated in near real-time. SLAM will not be implemented in this thesis so for more information on SLAM the reader should investigate[79][80]. Discussed below are two generic ways to map the local environment: through the use of 3D models, and through the use of occupancy grids.

3D Models: This approach can lead to a highly accurate model of the three dimensional environment by using the point cloud information gathered from the environment

and identifying features in that environment by surface fitting of geometric primitives such as spheres and planes. Cheuk Yiu Ip and Satyandra K. Gupta[81] proposed a method of this type that classifies objects pulled out of the gathered point cloud data to reconstruct the environment via a hybrid of surface patch models derived from the point cloud, and surface mesh models from computer aided drafting (CAD) drawings. The technique described in [81] pulls in the point-cloud data. It is then segmented into surface meshes according to curvatures. A library of CAD models is also converted into surface meshes and this library is used to compare the segmented shapes in the collected point cloud. Once all the surface mesh segments are identified the next step is to compare each identified segment with a possible match in the CAD library through the use of the PCA registration algorithm. This will develop a translation and rotation between each identified mesh and a candidate CAD model. Only the largest N number of surface patches are used in order to reduce the number of mismatches. The resulting world model is a surface mesh based hybrid of CAD shapes and collected point clouds to make a more complete environment.

Occupancy Grids: Another way to derive a world map from point cloud information is through the use of occupancy grids. This approach segments the world into small segments, also known as voxels, which is comprised of a data cube that states what is in that specific location. In the very basic sense, the voxel states if an object has been detected in that location via sensor or if it is empty. Figure [17] shows an example of a 2D occupancy grid depicting a simple room. In this example the voxels where the walls are indicated simply state they are occupied with something whereas the voxels in the rest of the map state they are empty and therefore it is clear to navigate.

One variant of the 3D occupancy grid approach, as proposed by J. Jessup et. al.[82], uses Octree based occupancy grids. An octree is a hierarchical data structure for spacial subdivision which provides a collection of parent nodes that discretizes the 3D space into voxels as in the traditional occupancy grid but these voxels are recursively subdivided into

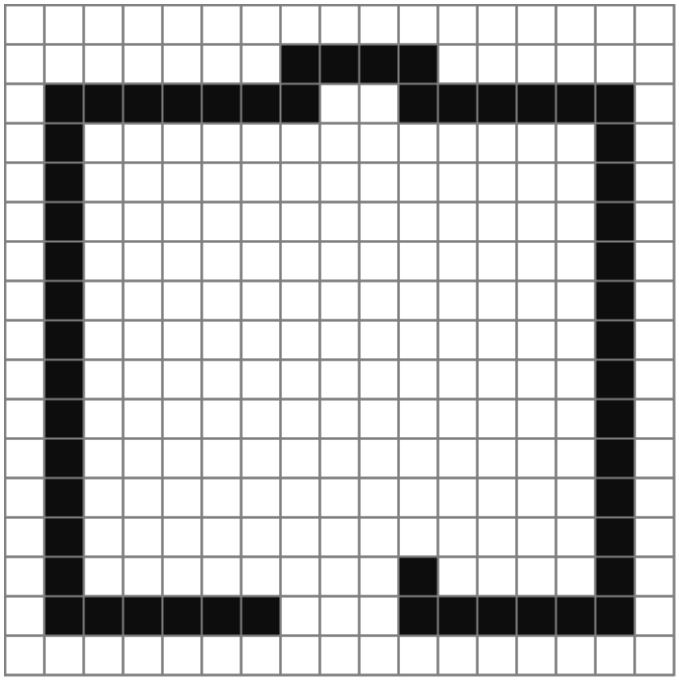


Figure 17. Generic Example of an Occupancy Grid of a Room

child nodes until a desired resolution is achieved[83]. Using the octree phenomenology potentially saves on processing power and memory requirements as the 3D space grows exponentially large as the detail in the world map increases. The efficiency in octree mapping is in the ability to prune and consolidate nodes. Each node has a probability of occupancy based on the gathered sensor data. After a given set of measurements are taken the probability will reach either an upper bound or a lower bound. When the probability of occupancy for a voxel crosses one of these thresholds then it is declared as either occupied or empty. If a group of voxels all contain the same value then the parent-voxel can take on that value representing the collective size of the child-voxels. If a new sensor measurement is taken that represents the opposite value from what the parent-voxel indicates then the parent voxel will be re-subdivided back out to the set of child-voxels.

III. Methodology

This chapter presents the experimental tests used to achieve the primary research goal, which was to calibrate a modified 2D scanning LiDAR sensor to be used as a 3D sensor. The end results present the extrinsic parameters of the modified LiDAR sensor used to map the output of the modified sensor from the original 2D design into a 3D space. A walk-through is conducted in chapter 3 starting with an operational foundation set by characterizing the unmodified sensor. Part of this process identifies an appropriate target type to use during the remainder of the experiments in order to simplify the experimental procedure in those that follow. This sensor characterization enables a stronger understanding of the sensor performance, sets expectations, and helps identify possible limitations for later experiments.

After the sensor was characterized, an additional characterization was conducted to understand the effects to the sensor performance once a mirror is placed into the path of the sensor beam. This was an unexpectedly important component of the experimental process. It was critical to understand how the sensor beam was modified by the inclusion of a mirror in the sensor beam path, and how that effected the sensor range measurements. Even more importantly, additional insight was achieved though the testing process that identified test scenario flaws in the initial experiments, which required further investigation in order to understand how to achieve a successful data collect for each experiment. The test experiments presented in this chapter reflect the final version of each test, which does not reflect the iterative steps it took to get there. The effects of a mirror that did not directly affect the sensor range measurements were taken into consideration as part of a heuristic approach to the design of the modification to the sensor presented in chapter 4.

A range estimation function is presented next, followed by a set of experiments used to validate the ability of the range estimation function to be able to accurately estimate the sensor range measurements. Part of this process is the development of an error model that

describes the error between the range estimation function and the sensor range measurements. The error model is built based on the results of the characterization of the sensor, and is intended to minimize the difference between the sensor range measurement and the range estimation function. Once finalized, the range estimation function, with a working error model, is used to create an algorithm to estimate the extrinsic parameters of the sensor beam being deflected by a mirror.

In the next section, a recursive least squares approximation is used to estimate the extrinsic parameters of the sensor beam once deflected by a mirror. The experiments step through a process to verify the convergence to a realistic solution, the repeatability of those results, and then to verify the accuracy of the results. The final sections presents the procedure to implement the results of the previous steps, to estimate the extrinsic parameters of the 3D system, to transform the output data of the 2D scanning LiDAR sensor into the actual 3D set of points in the true environment. Lastly, a high level step-by-step calibration procedure is presented as a summary to chapter 3.

3.1 Characterization of the Sensor

Characterization of the unmodified sensor is comprised of seven basic tests. After describing the basic test scenario and equipment used, the section presents the first experiment, the range test. The range test is used to statistically characterize the range measurements. The data collected by this experiment is a foundational component that leads to other experiments. The test experiment is designed to characterize the beam divergence pattern of the LiDAR sensor beam. It is important to understand the shape of the sensor beam as it propagates out from the sensor. This give insight into how the sensor "sees" the world.

The next experiment, the gray level test, is intended to characterize how the grayscale darkness of a target effects the sensor range returns. The results of this experiment decide

the desired target color for the remainder of the experiments. Once finished, an angle test is conducted to characterize how the sensor range return is effected by the orientation of the target if the target is at a constant distance.

The last two experiments in this section identify characteristics of the sensor which are useful during the design phase of the follow-on experiments presented later in the chapter, but are not used as part of the range estimation function derived in Section 3.3. The first experiment is conducted to characterize how small a target can be before the sensor can identify it as a target, and not as noise. The last experiment is designed to illustrate the intensity return as a function of range for a target with a constant orientation.

Basic Test Setup.

Before any characterization testing could be accomplished the equipment needed to be connected and tested for proper operation and to verify data can be accessed. The base test set up, as seen in Figures 18 and 19, involved the interoperability between the sensor, a data collecting device, a known target, and an additional sensor with a known output to verify the accuracy of the sensor under test.

One feature of the Hokuyo sensor is that it publishes its output and operating parameters via the use of a Robot Operating System (ROS) publisher. ROS is an open source library of Linux packages written in mostly C++ or python [84]. As a user it is possible to run through the openly published tutorials [85] to develop a completely virtual world within ROS, housed on a Linux machine, with virtual robots such as the turtlebot [86], to freely experiment with experimental algorithms and code. With the use of nodes, publishers, and subscribers, ROS allows users to establish communications lines between many different elements of a robot, sensors, and external code, to create simulations representing a real environment.

The data collecting device was a Macbook Pro laptop [87] which is running an Ubuntu

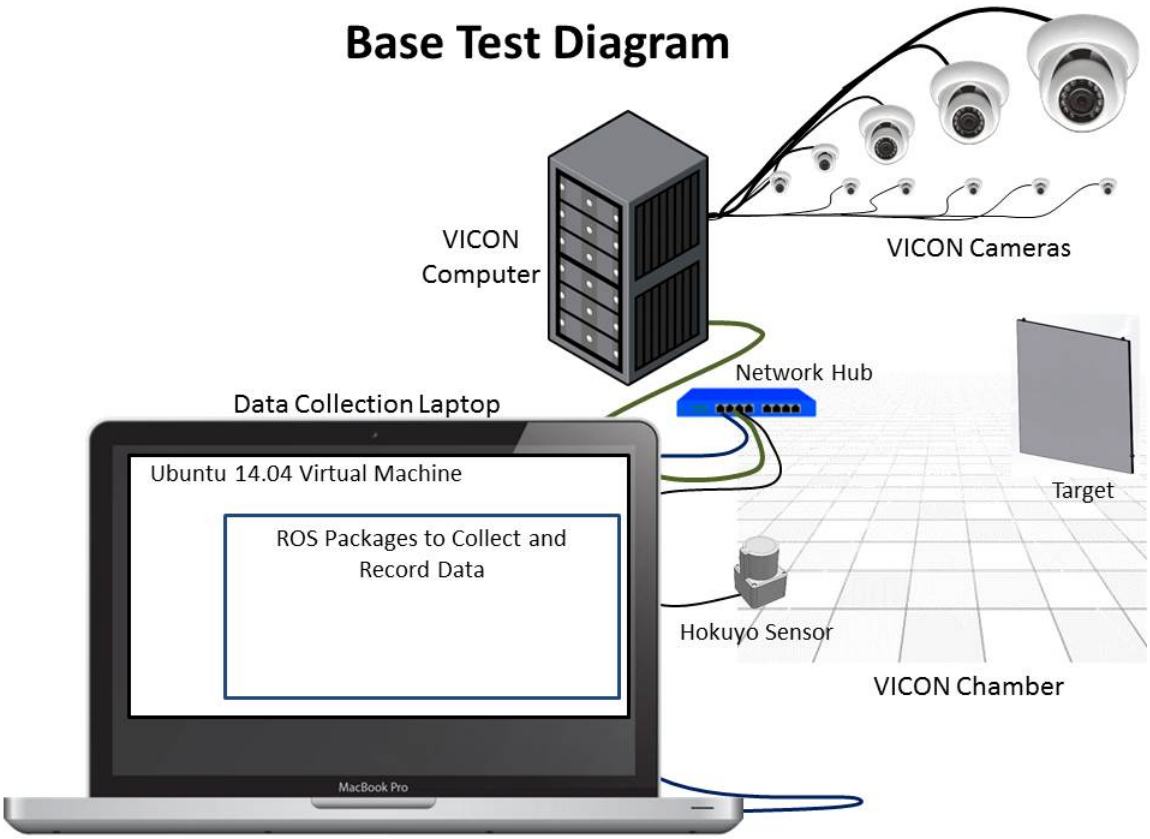


Figure 18. Base Test Diagram

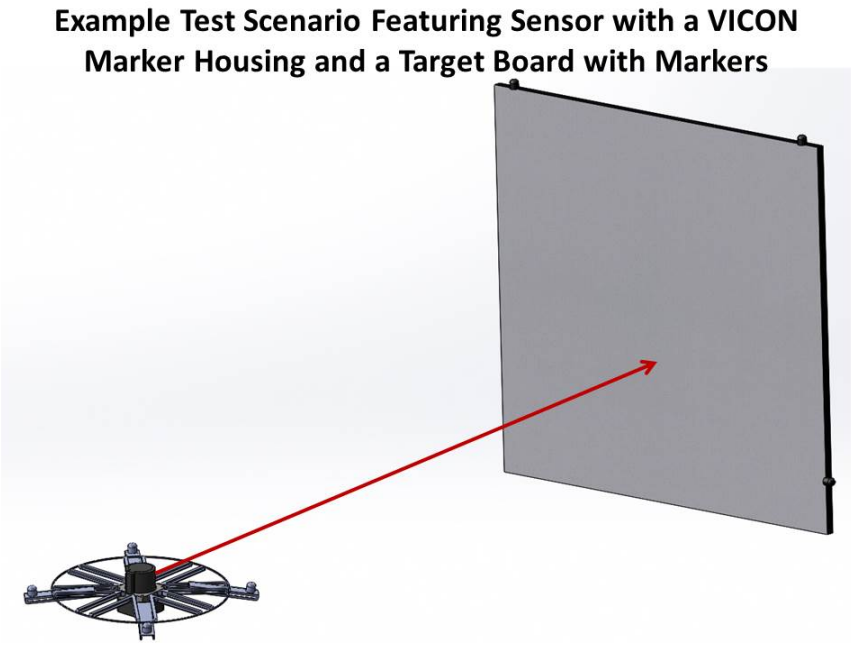


Figure 19. Example Sensor and Target with VICON Markers Attached

14.04 [88] virtual machine (VM) through the use of Parallels software [89]. ROS packages were installed within Ubuntu. The next step was to bring in and set up a device to provide "truth" data. The truth data is data that is, for all intents and purposes, known to be accurate with previously known statistics. For this effort a VICON chamber was used to provide truth data. A VICON chamber is a motion capture system that uses IR sensors and reflective markers, which are placed on the objects you wish to track in the chamber. A case study portraying the implementation of a VICON chamber can be seen in [90]. The benefit of using a VICON chamber is that once an object is created for the sensor and target the object's location and frame can also be published in ROS. Expected ranges between the sensor and target can be calculated using this published data in order to help evaluate what is being published by the sensor itself. An example of a VICON chamber can be seen in Figures 20, 21, and 22.

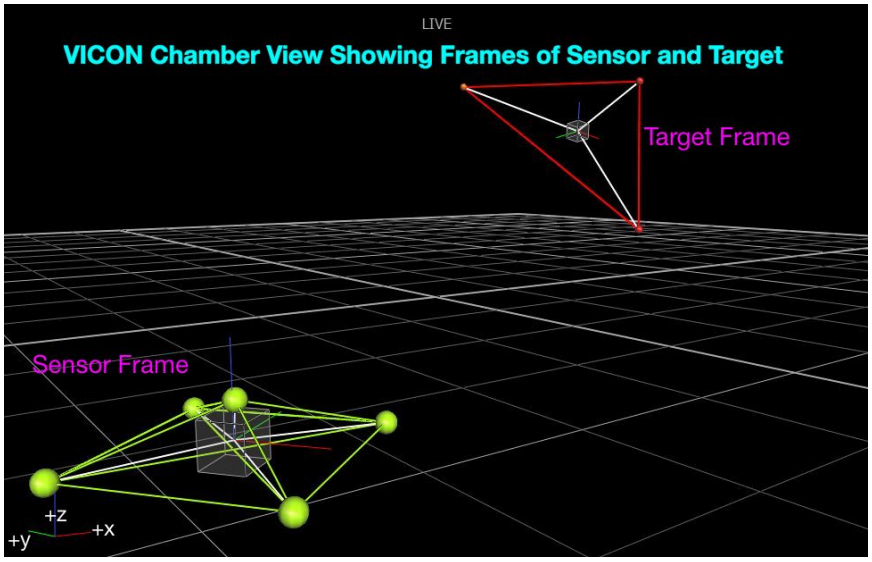


Figure 20. VICON Object Display

Code was written in C++ to create subscribers to the published data by both the Hokuyo sensor, and the VICON chamber. To collect the relevant data it was necessary to add additional code to publish data to a comma separated variable (CSV) file to be processed by MATLAB. The structure of the final output can be seen in Figure 24 and makes available

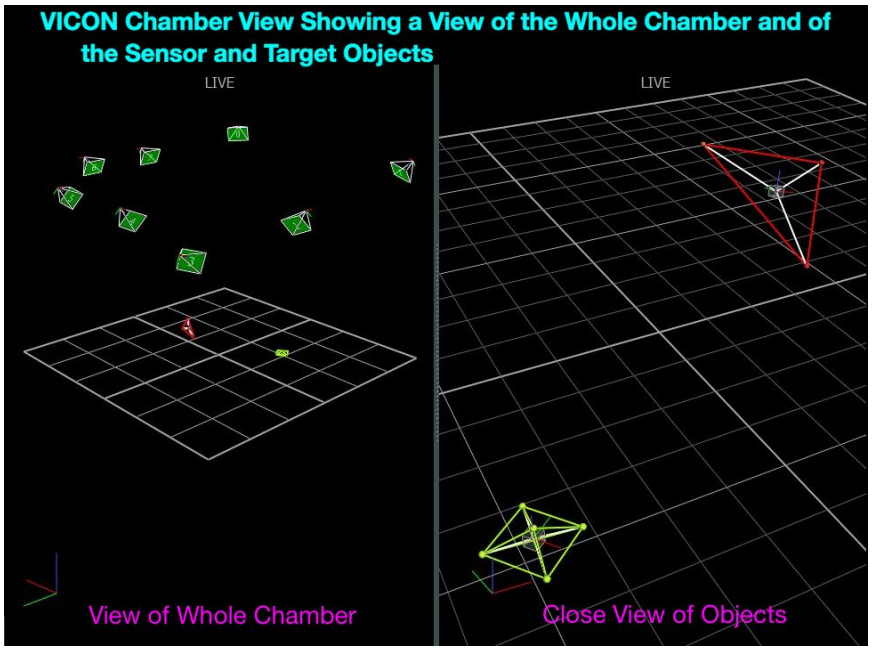


Figure 21. VICON Object and Camera Layout

to all subscribers the ROS time stamp, the test trial currently being run, the base angle of the laser beam being propagated from the sensor, the corresponding element number of the published range, the range measured by the sensor, and the position of the target board relative to the sensor frame. To calculate the position of the target board required a transformation between the world frame as published by the VICON chamber and the sensor object frame through the use of the ROS TF package [91] which performs the transform similar to the one shown later in Section 3.5. An additional transformation between the sensor object and the ideal point of origin of the sensor beam is added to eliminate any offset introduced by the sensor object as a collection of VICON markers which are necessarily non-symmetrical.

The additional frame to the sensor point of origin is developed by setting up the test layout as seen in Figure 25. Using the longest distance that could be reliably measured in the VICON chamber, 4.00m was chosen as the target distance. The target board location was measure as shown in Figure 25, and the center of the beam pattern was marked on paper attached to the surface of the target board. It was possible to view the sensor beam

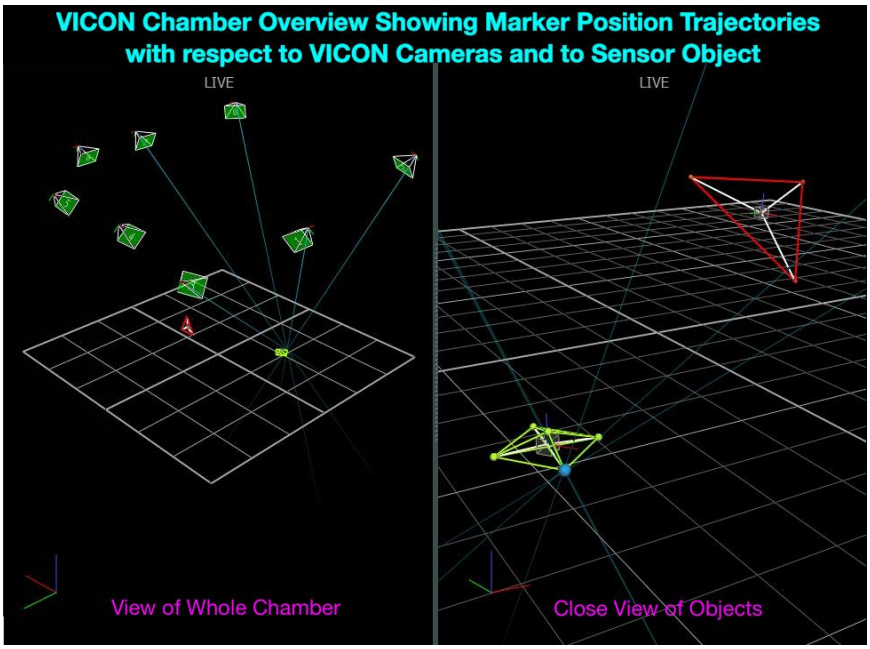


Figure 22. Example VICON Marker Trajectories

Hokuyo Sensor Output Structure

Timestamp	Sequence Number	Header Stamp	Frame ID	Minimum Angle	Maximum Angle	Angle Increment	Time Increment	Ranges	Intensities

Figure 23. Hokuyo Sensor Output Data Structure

pattern through the use of an IR scope as shown in Figure 26. This particular IR scope allows a user to view near IR between 350nm-2000nm, which is perfectly suited to view the sensor beam propagated at 905nm. Using this information an initial board position relative to the sensor is gathered. Rotation and translation parameters are then calculated using Equations 44 through 47 shown later in Section 3.5, and then the base frame with respect to the sensor origin is created with the new parameters.

Each test is conducted in the same general fashion, as seen in Figure 19, with the data being routed and manipulated as shown in Figure 18. One limiting factor on the Hokuyo

VICON Chamber Output Structure

Timestamp	Time Trial	Angle	Element	Range	P1	X(1)	Y(1)	Z(1)	P2	X(2)	Y(2)	Z(2)	P3	X(3)	Y(3)	Z(3)

Figure 24. VICON Chamber Output Data Structure

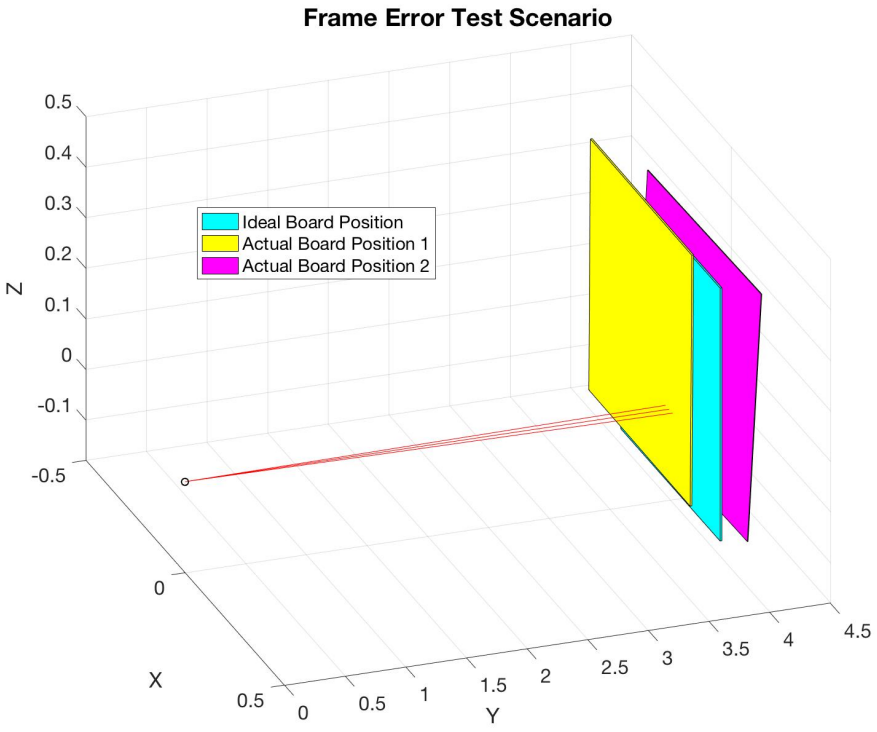


Figure 25. Example Scene Showing Target Position Error Due to Frame Transformation Errors

sensor hardware is that it is not possible to only turn on only one beam at a time therefore multiple beams must be used at all times. This shows up during the beam divergence test with the physical representation of the beam as an array of three elements shown at once, illustrated in Figure 27, instead of a single beam spot as seen in Figure 39. Another critical element is lining up the beam path to intercept the target board, which was done using the IR scope.

Figure 26. Newport IRV2-2000 IR Scope

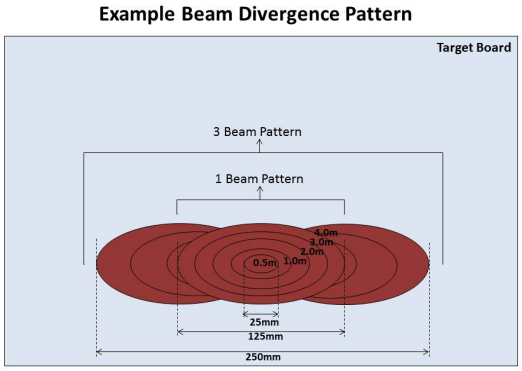


Figure 27. Beam Spot Size at Varying Distances

Range Test.

To begin the sensor characterization, a test needed to be conducted to evaluate the mean, variance, and standard deviation of the range measurements published by the Hokuyo UST-20LX scanning laser rangefinder within a standard set of operating ranges expected to be encountered in an indoor operating environment. For our case the average range to be seen by the sensor is expected to be 1m-4m as a typical indoor environment. This test was designed based on the basic setup described in Section 3.1 and is illustrated in Figure 28. At each distance measured, the target board is perpendicular to the beam path and the distance of the target board is adjusted until the 100 sample mean of the sensor output range is measured to be $\pm 1.0mm$ from the target distance. The Hokuyo sensor was set to only output the three central scanning locations, 0.25°, 0°, -0.25°. Once the initial setup was complete the data collection procedure was run until 10,000 samples of each scanning location were collected. This data was imported into MATLAB, and then the mean and standard deviation was calculated.

Once the mean was established, this was compared to the physically measured distance to the target board location at each range via the use of a tape measure. The expected tolerance for the tape measure $\pm\frac{1}{16}$ inch $(\pm 0.7937mm)$.

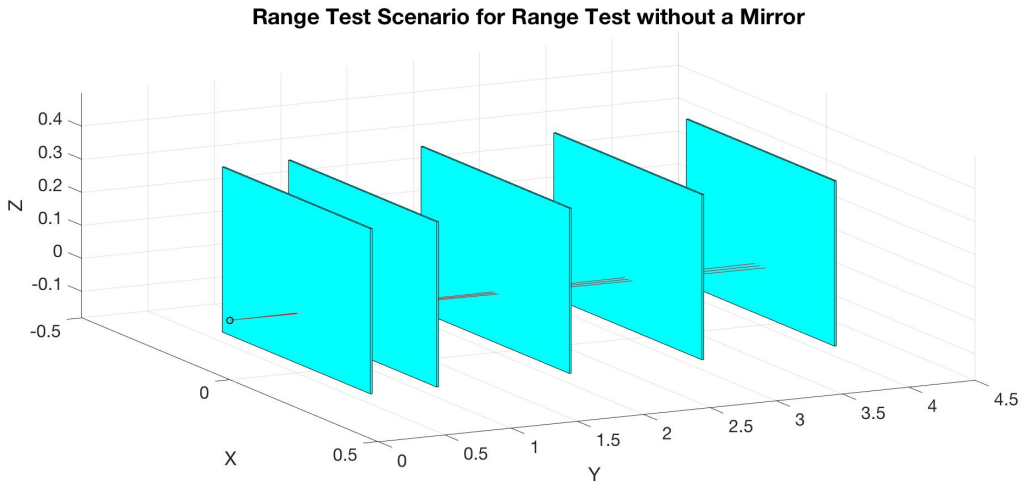


Figure 28. Illustration of Target Positions for Range Test Relative to Laser Source

Beam Divergence Test.

The next test in the sensor characterization process was to measure the beam divergence as a function of distance from the sensor. This test had exactly the same setup as the range rest in 3.1 and was conducted during the range test procedure. Once the data was collected at each distance in the range test, a white piece of paper was secured onto the target board and then the VICON chamber and the rooms lights were turned out. Once the chamber was sufficiently dark the IR scope in Figure 26 was used to hand trace the projected beam pattern onto the target board which is estimated to look similar to the illustration in Figure 27. The beam pattern was traced at each distance in turn.

Gray Level Test.

Different range returns were achieved between the use of a black colored target board and a white colored target board, and therefore it was deemed necessary to try and characterize this difference. This test was conducted similarly to the range test. At each given range, as seen in Figure 29, a small target board was placed at the range location. A pure white board was used as the initial setup at each range position to establish the baseline for

comparison with other colors. The range of colors were limited to the grayscale due to the magnitude of possible colors to test against. The chosen range of gray colors can be seen in Figure 30. During fabrication of the target boards, two boards were left out and unpainted (the dark wood, and the light wood) and added to the range of target boards to test against.

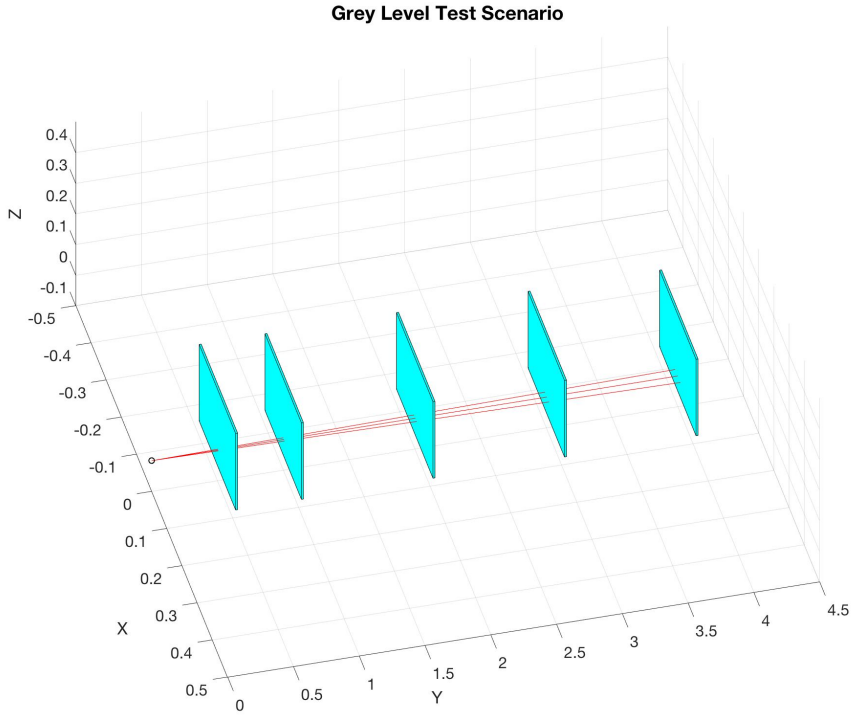


Figure 29. Illustration of Target Positions for Gray Level Test Relative to Laser Source

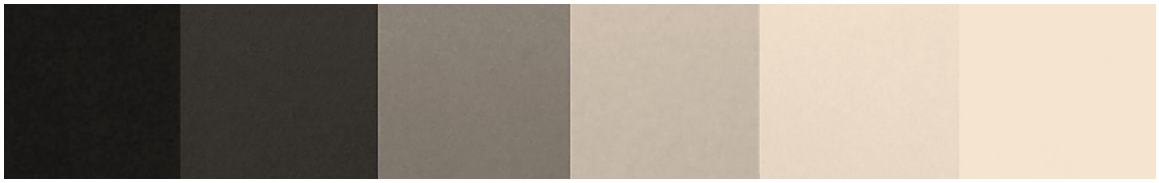


Figure 30. Gray Level Test Target Colors

For each desired range position the pure white target board is used as the initial baseline to adjust the target so that the sensor outputs a range to within $\pm 1.0mm$ of the desired target distance. Once the target distance is established a set of reference objects are placed

adjacent to the target board to be able to locate the precise position of the target board when the target board of the next color in the series of target boards are placed. For each target board 10,000 data points are collected and exported to MATLAB.

Once the target board with a returned range most closely resembling the measured range was identified the Range test and Beam Divergence test was re-conducted. We identify this target board as target board zero.

Angle Test.

Once target board zero was identified, a test to determine the effect that the angle between the normal vector of the target and the propagation sensor beam have on the error in the measured range needed to be conducted. Due to the fact that the spot size of the sensor beam is not infinitely small, and has a projected area whose size grows as a function of distance, it is necessary to characterize this effect.

The basic test layout is presented in Figure 31. The test procedure begins with a target board placed at a 4.0m distance with a 0° angle between the normal vector of the target and the center sensor beam. A minimum of 10,000 Range measurements were then taken at this target orientation. Once collected the target orientation was cycled though the range of desired angles $[\pm 5^\circ, \pm 10^\circ, \pm 20^\circ, \pm 30^\circ, \pm 45^\circ, \pm 60^\circ, \pm 80^\circ]$. This allows for enough visibility on the effect that the angle has on the sensor range return in order to derive part of the error model in Sections 3.3 and 4.6.

Figure 31 shows the sensor origin as a black point with the center array beams at -0.25°, 0.00°, and 0.25° being propagated to the target board centered at 4.0m in the y-direction and a 0.0m offset in the x-direction. The target orientations are only being rotated about the z-axis of the target board frame which is also centered at 4.0m in Y and 0.0m in X. Figure 32 shows a much closer view at the expected beam intersection point with the targets. Here it is possible to identify the target orientations individually. Using this pattern of angles

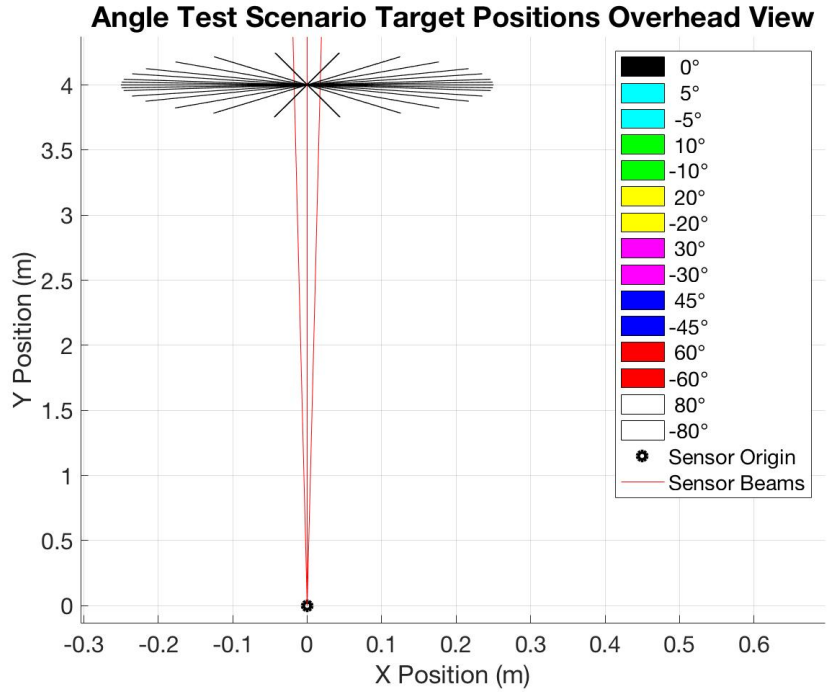


Figure 31. Angle Test Scene Overview

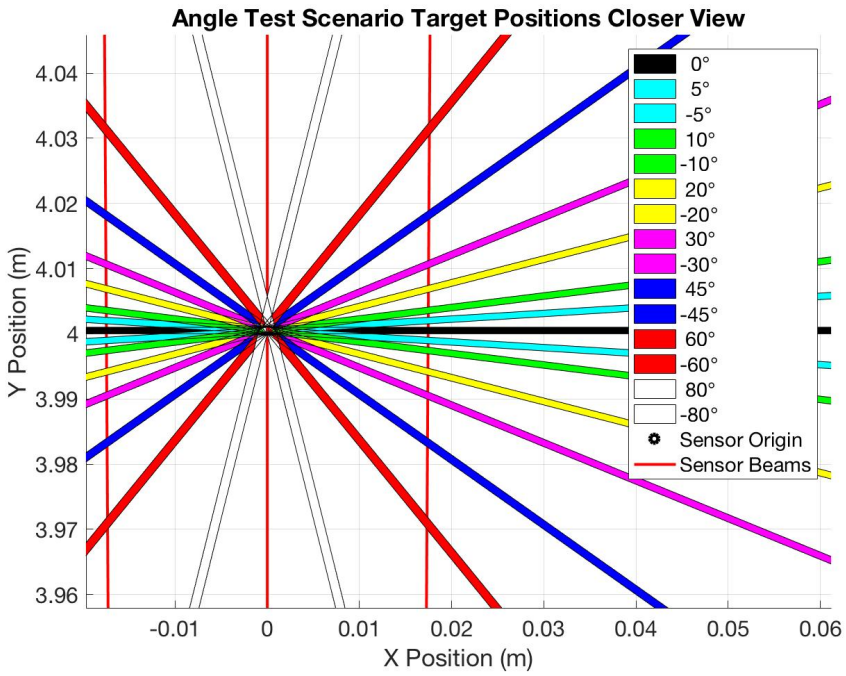


Figure 32. Angle Test Scene Zoomed in to Points of Interest

should result in a range return centered at x = 0.0m and decreasing as the angle of the target board orientation increases.

The results of this test were included as a component in the range function error model, r_{offset}.

Angular Resolution Test.

As part of the characterization of the Hokuyo UST-20LX scanning laser rangefinder, an angular resolution test was performed. Figure 33 shows the proposed test scene with the sensor origin in black, the sensor beams in red, and the two target boards in blue. The purpose of this test was to help understand the the effects of small "holes" in objects as they appear to the sensor. The test was conducted similar to the range test without a mirror. Two targets boards were used and placed perpendicular to the beam, at a target distance of about 10 meters. The targets were situated in such a way to appear as a single plane to the sensor, like viewing a wall. An initial base line of range measurements were taken, of 10,000 data points. The two targets were then separated, as seen in Figure 33, by $1/8^{th\circ}$ relative to the sensor, which equates to a gap in between the target boards of 21.82mm. Another 10,000 range measurements are taken. The procedure was completed for the angular distances of 21.82mm, 43.63mm, 65.45mm, and 87.27mm which correspond to the angular separation of 1/8°, 1/4°, 3/8°, and 1/2°.

Once the data was collected, the senor range measurements, which landed upon the target, were compared by sample means and standard deviation. The subsets comprised of the original dataset of sizes of 2, 10, 100, 1000, and then of the entire set of samples.

Intensity-Range Profiles.

An intensity-range profile test was performed to determine a more complete picture of the characterization of the Hokuyo UST-20LX sensor. This test can also be used to help

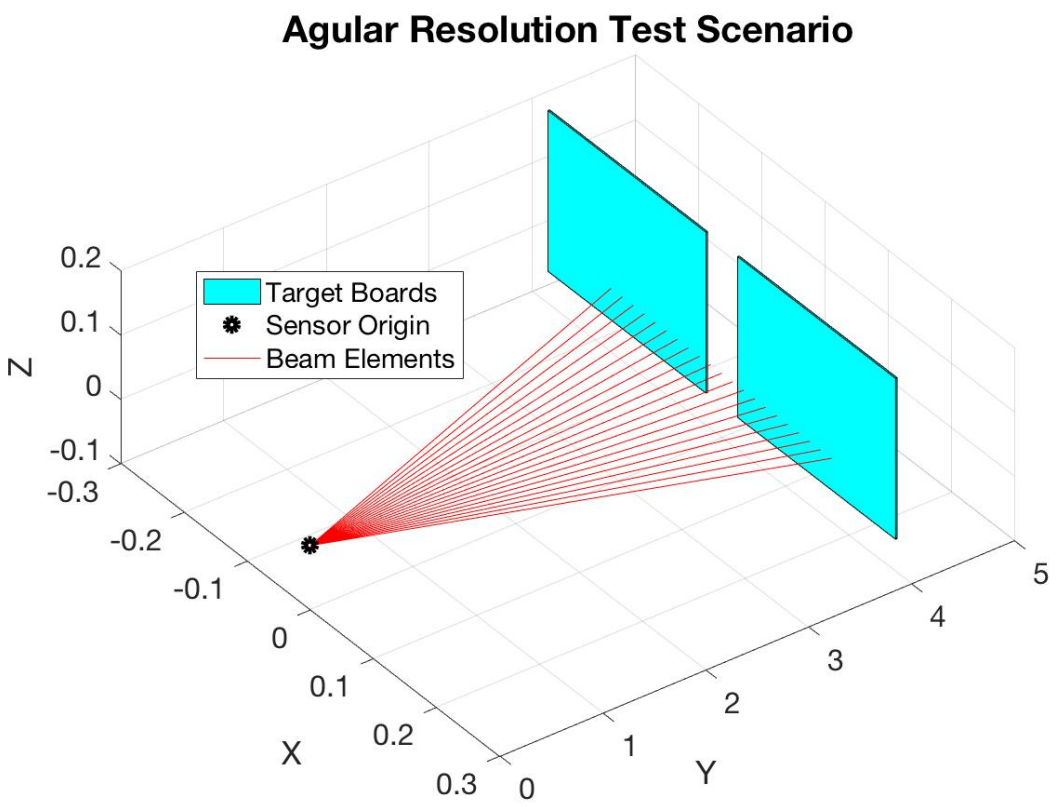


Figure 33. Illustration of Target Range for Intensity-Range Profile Relative to Laser Source

visualize the effects of adding a mirror to the propagation path.

The basic test layout, is shown in Figure 34. Test ranges of 1.00m-11.00m are marked on the floor using masking tape, with a 10.0cm gap and another mark is laid out behind each mark as seen in Figure 34. This 10.0cm known gap in the data collect location will allow for an indicator in the data during analysis to be able to compare actual ranges to the sensor readings and corresponding intensities.

Once the test area was marked and established the test was conducted. For each trial the target board was placed in the sensor propagation path at $4 \pm 1.0cm$ and slowly moved down the propagation path at a rate of about 5cm per second until reaching the first marker at 1.00m, at which point the board was removed from the beam's path. The target board was again placed in the path at the corresponding marker at 1.10m and slowly moved at the

same rate until the next marker at 2.00m was reached. The Again the board was removed and placed back into the propagation path at 2.10m. This process was repeated until all markers were reached which allowed for at least 8,000 data points to be collected and exported to MATLAB for analysis. Five total trials were completed and analyzed using target board zero.

Figure 34. Illustration of Target Range for Intensity-Range Profile Relative to Laser Source

3.2 Identify Effects of a Mirror

Once the characterization of the unmodified sensor is complete, a characterization of the sensor as it interacts with a mirror in the projected beam path is necessary due to the fact that the modification to the sensor will be comprised of an external structure with many mirrored surfaces. Recalling the potential noise sources identified in chapter 2, the optical path is a fragile system, and this system is perturbed in many unforeseen ways by adding a mirror in the sensor's propagation path. The first subsection in this section discusses some properties of mirrors such as flatness, and reflectivity for a few common materials used to fabricate mirrors. From here the range test is revisited, but with a mirror deflecting the sensor beam at 20° in elevation. After the the range test is completed, the beam divergence test and the test to determine the intensity profile as a function of range is also re-completed in the same manner.

Surface Reflection.

Adding a physical component to the propagation path of a laser beam can carry a number of unwanted effects, with the most critical to be to completely block the beam from propagating. A major step in choosing the right surface to properly deflect the previously described laser beam is to investigate how the wavelength of light (905nm) is reflected off a perspective surface. Figure 36 shows data gathered by Polyanskiy[21] which displays the reflectance one can expect from a perfectly mirrored finish on Silver, Gold, Aluminum, Platinum, and Titanium near 905nm. The mirrors used in this research were aluminum coated mirrors obtained through First Surface Mirrors [10] with the reflectivity profile in Figure 37.

A critical component of the mirror being introduced to the beam path is the "flatness" of the mirror surface. Flatness can be rated by the number of wavelengths, 1λ. For example, the mirrors used in this thesis were rated at $1/\lambda$ per inch whereas an application requiring much more precise reflection would use mirrors with a flatness rating of $\frac{1}{4}\lambda$ per inch or even $\frac{1}{20}\lambda$ per inch which climb in price quickly as seen in Figure 38, which shows current costs from Edmund Optics [92]. A sufficiently non-flat mirror can introduce aberrations

Figure 35. Example of Distorted Mirror

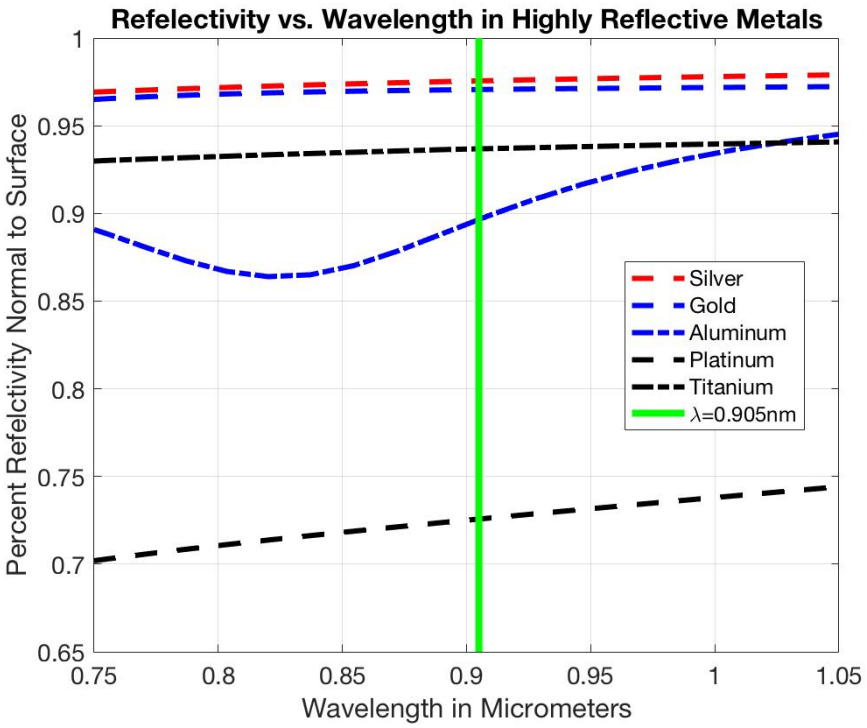


Figure 36. Expected Reflection Intensity From Common Mirrored Surfaces

to the reflected light as seen in Figure 35 which are severely aberrated and the projected reflections become interesting to look at but unusable for a laser scanning range sensor.

A final consideration of the physical property of the mirror involves edge effects at the wavelength of interest that are within the field of view of the receiving sensor. Within this research there are two primary causes of these edge effects. The first cause is due to the spread of the propagating beam in a Gaussian pattern as seen in Figure 39. The extreme tails of the beam, although unseen by the IR scope at such week intensity levels, may fall on the edge of the mirror. Due to the rough nature of the physical edge of the mirror the surface is likely to cause a "glint" of large enough magnitude to be registered by the receiving sensor as a viable return signal, and a corresponding range measurement is recorded. This range measurement is erroneous and unusable. To overcome this phenomenon, it is necessary to use a large enough mirror so that the edges are no longer in the field of view of the sensor.

The second edge effect result when the mirror is angled at small grazing angles and

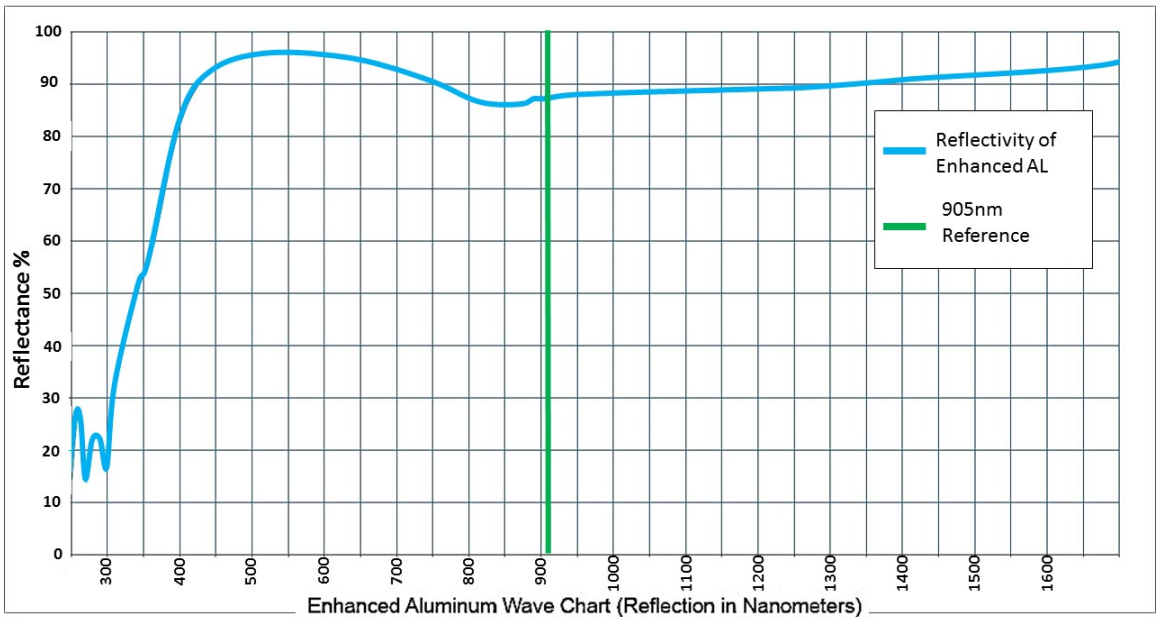


Figure 37. Expected Reflectivity from First Surface Mirrors Used in Experiments

some of light from the laser beam actually propagates along the surface of the mirror and is reflected back at the edge of the mirror. Because of these small grazing angles it would take a very large mirror with respect to the sensor size to overcome the field of view issue previously mentioned. A very large mirror is unusable in this application, and to overcome this issue beam deflection angles between 20-90 degrees are used.

Figure 40 compares a logarithmic view of the profile of the estimated Gaussian beam intensity normalized to one at 1.0cm distance from the Hokuyo sensor source. A 5% reference value is marked with respect to both graphs, which is a very low intensity value for the sensor. The logarithmic view allows one to see the intensity is not zero after reaching greater than 2mm from the center of the beam even though it would appear so. This value may not be perceptible to the human eye but the sensor can sense these values. If this was the highest return of a range measurement then a valid range value may be output as a real reading. The return caused by the glint of the edges of the mirror within the sensor's field of view may not be perceptible with the IR scope 26 used for test scenario set up, but a

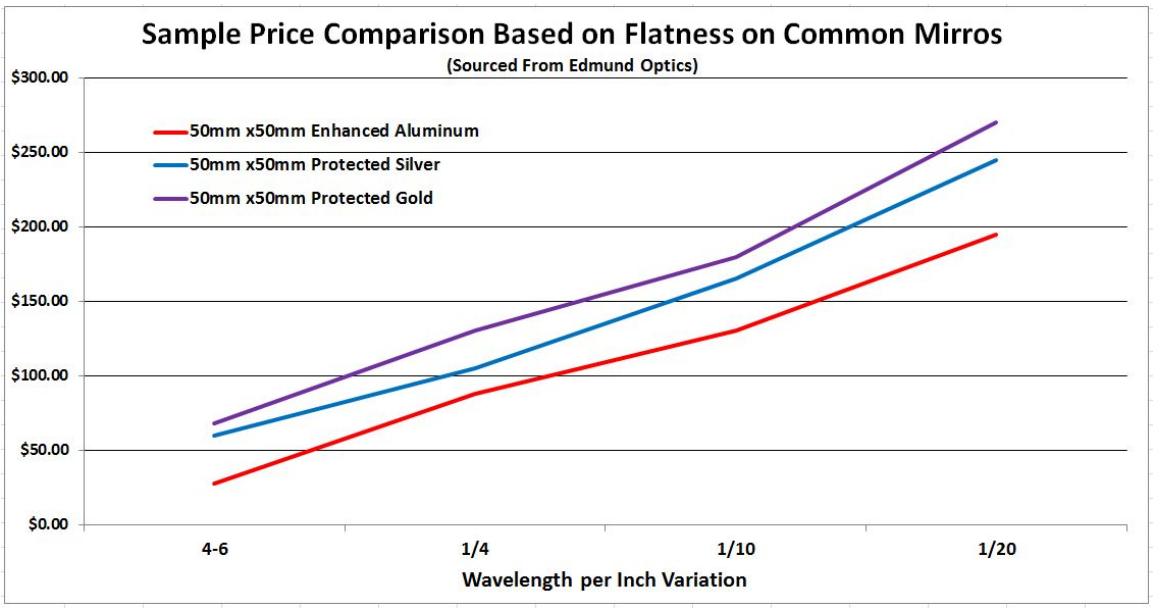


Figure 38. Price Comparison on Mirrors with Varying Flatness [10]

false return may still be sensed due to these low intensity returns.

Range Test.

This test was designed based on the basic setup described in Section 3.1 and is illustrated in Figure 28. The only difference from the basic test previously completed is that a mirror is placed in the propagation path at 4.0 cm from the sensor with a 20 degree deflection in elevation and azimuth as seen in Figure 41. This angle is chosen as it is the lowest angle in the usable set and will therefore produce the largest distortion of the beam pattern. At each distance measured, the target board was perpendicular to the beam path, and the distance of the target board was adjusted until the 100 sample mean of the sensor output range is measured to be $\pm 1.0mm$ from the target distance. The Hokuyo sensor was set to output only the three central scanning locations (0.25°, 0°, -0.25°). Once initial setup was complete, the data collection procedure ran until 10,000 samples of each scanning location were collected. This data was then imported into MATLAB and the mean, and standard deviation were calculated.

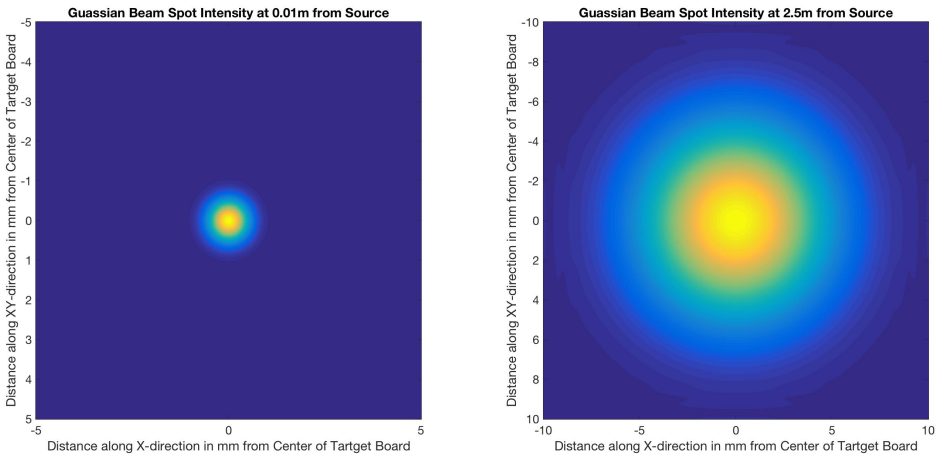


Figure 39. Estimated Gaussian Beam Spot Sizes at 1.0cm and 2.50m from Source

Once the mean was established, this was compared to the physically measured distance to the target board location at each range via the use of a tape measure.

Beam Divergence Test.

As before, the next test was to calculate the beam divergence as a function of distance from the sensor, this time with the mirror deflecting the beam. This test had exactly the same setup as the range test in 41 and was conducted during the range test procedure with the mirror deflection. Once the data was collected at each distance in the range test, a white piece of paper was secured onto the target board, and then the VICON chamber and the rooms lights were turned out. Once the chamber was sufficiently dark, the IR scope in Figure 26 was used to hand trace the projected beam pattern onto the target board. The beam pattern was traced at each distance in turn.

Intensity-Range Profiles.

The Range test layout with the deflecting mirror, as seen in Figure 41, was used to measure the combination of intensity and range values. Test ranges of 1.00m-11.00m were marked on the floor using masking tape, with a 10.0 cm gap and another mark was laid out

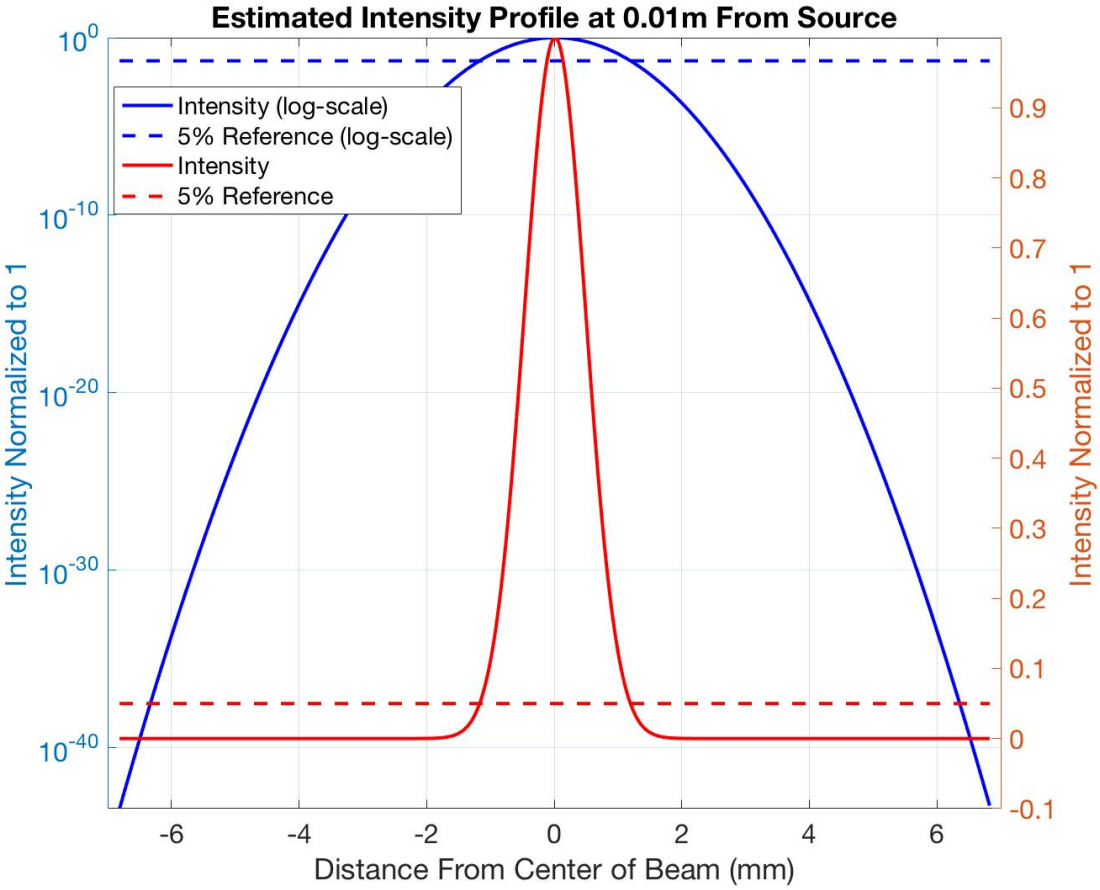


Figure 40. Side View of Gaussian Beam Spot Intensity Compared to a Logarithmic Scaling

behind each mark as seen in Figure 42. This gap will allow for an indicator in the data during analysis to be able to compare actual ranges to the sensor readings and corresponding intensities.

Once the test area was marked and established, the test was conducted. For each trial, the target board was placed in the sensor propagation path as described earlier but with a mirror in the beam propagation path. Again, five total trials were completed and analyzed using target board zero.

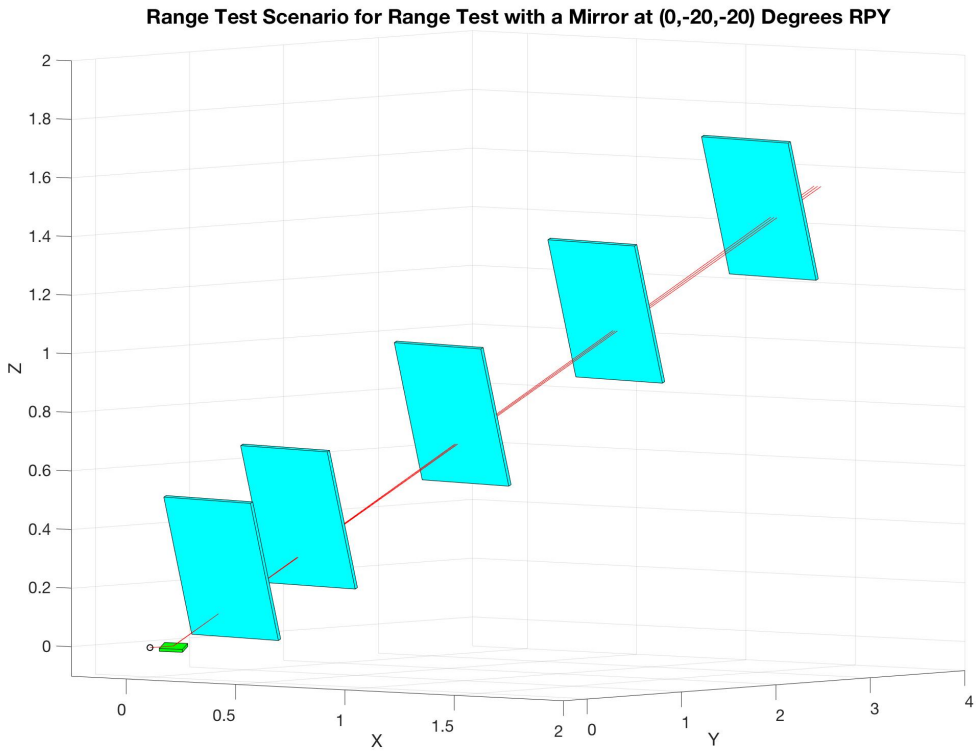


Figure 41. Illustration of Target Positions for Range Test from a Mirror Relative to Laser Source

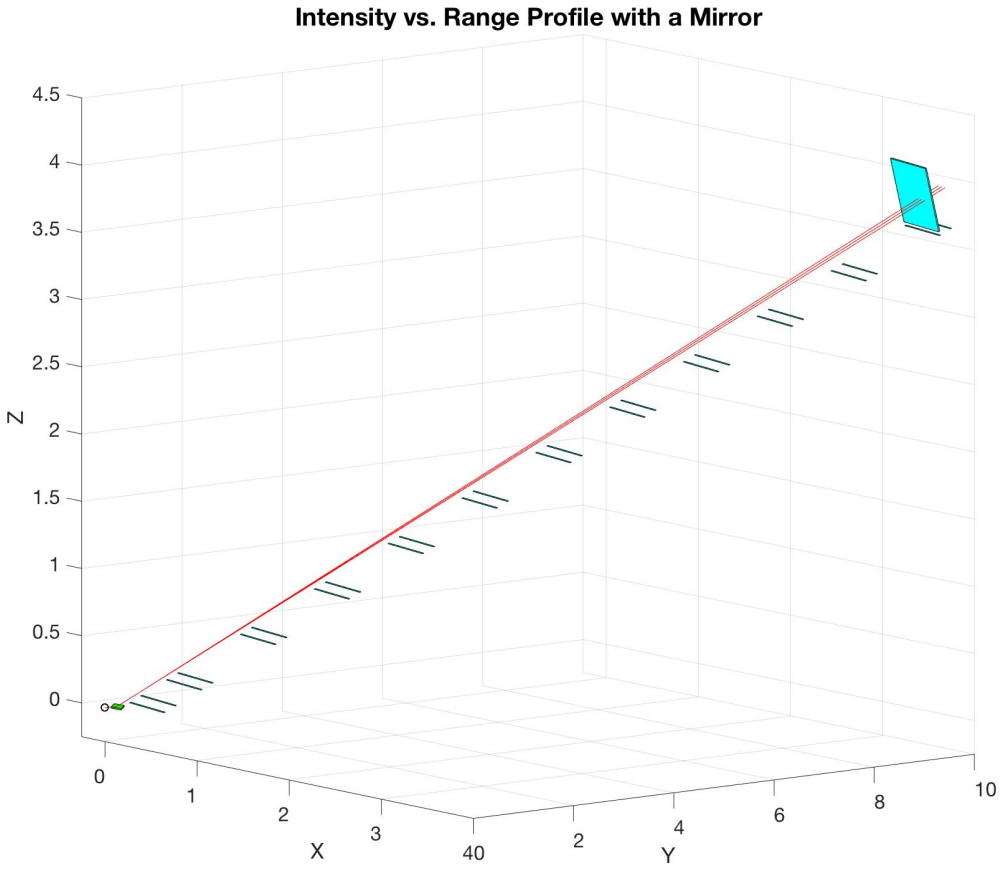


Figure 42. Illustration of Target Range for Intensity-Range Profile Reflected from a Mirror Relative to Laser Source

3.3 Range Function

Overview.

The next step in the calibration algorithm, after characterizing the sensor, was to develop a function to estimate the expected range of the sensor with a given target location in any of the previous test scenes. This is an important step in the overall calibration algorithm, and is the cornerstone of the Recursive Least Squares algorithm developed later on to estimate the extrinsic parameters of the modified sensor. The range function algorithm flow chart can be seen in Figure 43 and only works within the following set of assumptions:

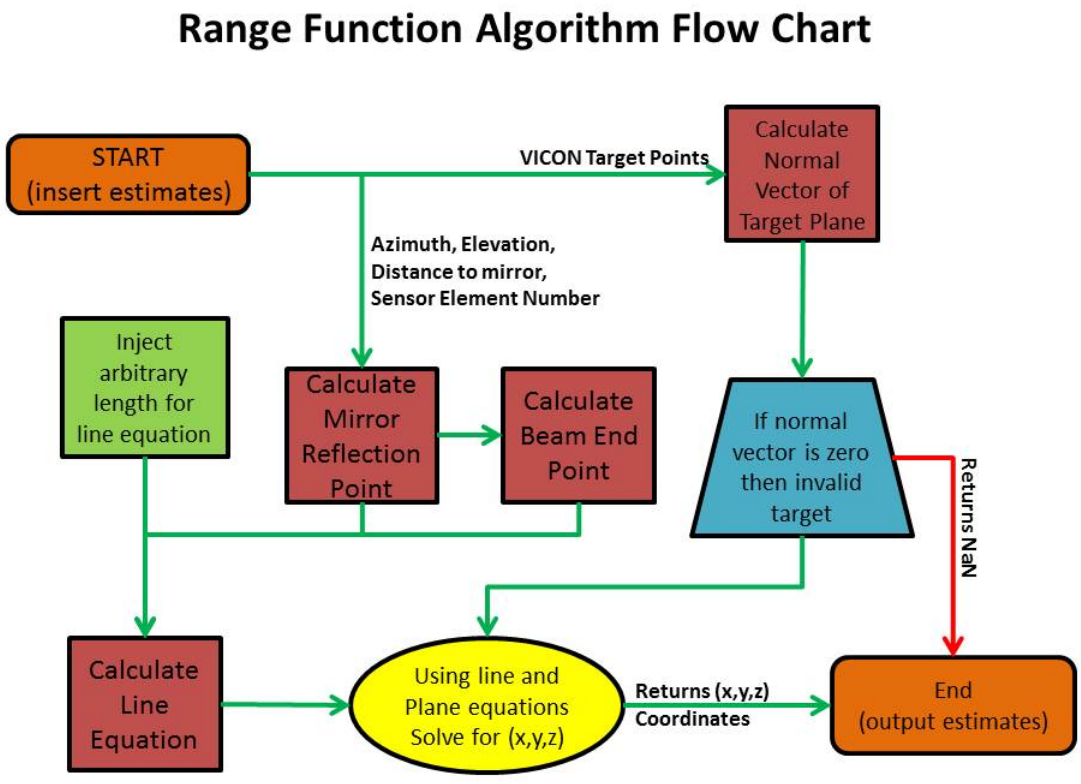


Figure 43. Range Function Algorithm Flow Chart

- The target board position is known and stable around the time of position measure-

ment.

- The azimuth and elevation of the mirror are known and stable around the time of position measurement.
- The Azimuth of the originating sensor beam is known.
- The distance between the sensor origin and the point of deflection on the mirror is known and stable around the time of position measurement.
- There is only a maximum of one deflection point in the beam propagation path prior to hitting the target board.
- The target board is flat with a rough surface producing a diffuse reflection pattern.

Using these assumptions, the range function in Equation 19 first takes the target coordinates in Equation 20, then combines this with the plane equation in Equation 23, and the estimated pointing vector of the individual beams in Equation 29. Solving for the the point where the plane and line are equal results in the estimated target intercept point from Equation 31 which is then used to calculate the estimated range in Equation 32.

The range function, seen is Equation 19, is a function of the target board coordinates, P_P, the base azimuth of the beam from the sensor origin, θ_B, the azimuth of the deflection angle due to the mirror, θ_a, the elevation of the deflection angle, ϕ, the distance to the deflection point on the mirror from the sensor origin, d, and some unknown noise, ν. The result is the estimated range, $r(\tilde{x})$.

$$r(\tilde{x}) = f(P_P, \theta_b, \Delta\theta, \phi, d) + \nu \tag{19}$$

Figures 44 and 45 represent a planar view of the X-Y, and X-Z planes respectively. For the purposes of this research, the orientation of the sensor frame is described such that the X-Y plane is the coincident with the sensor 2D scanning plane. The z-direction is up in

the sensor frame, and the Y-direction corresponds to the front of the sensor and a sensor azimuth of 0°. The illustration in Figure 44 shows the base azimuth angle, θ_B, in green that ends at point P_{L1}, which is the starting point of the intercept vector of the target. P_{L1} is also the point of deflection at distance d from the origin. The deflection angles, θ_a, and ϕ are shown in Figures 44 and 45. P_{L2} is used to create a vector from P_{L1} which intercepts a target board described by P_{P1}, P_{P3}, and P_{P3} at an unknown point P_{target}.

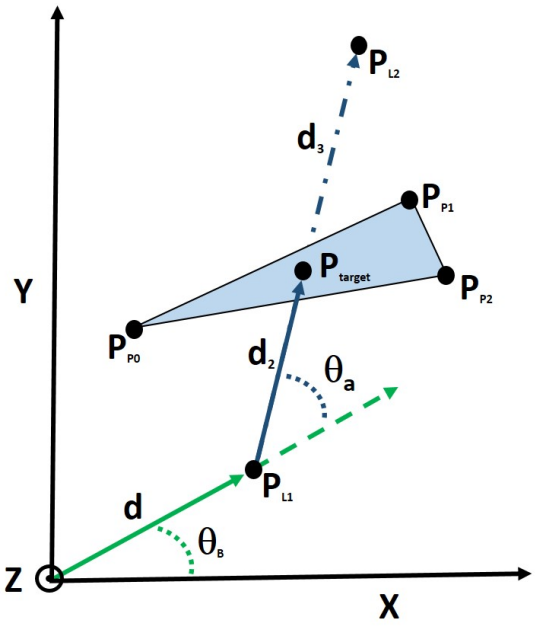


Figure 44. X-Y Frame (Horizontal) View of Target Intercept

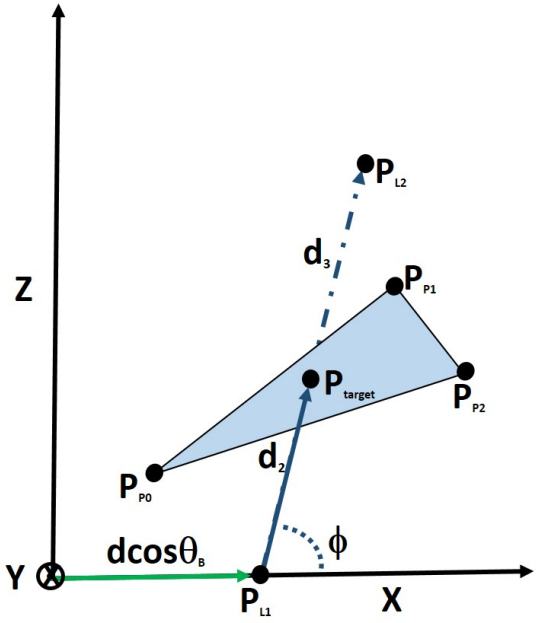


Figure 45. X-Z Frame View of Target Intercept (Z is the sensor frame vertical

In order to calculate the intercept point, P_{target}, we start with set of three points located on the target board's planar surface, P_{P0}, P_{P1}, and P_{P2} which are collectively described as P_P:

$$P_P = [P_{P0}, P_{P1}, P_{P2}] \tag{20}$$

which represent individual X-Y-Z coordinates for each point on the target board,

$$P_{P0} = (x_{P0}, y_{P0}, z_{P0}), \quad P_{P1} = (x_{P1}, y_{P1}, z_{P1}), \quad P_{P2} = (x_{P2}, y_{P2}, z_{P2}) \tag{21}$$

Next, the normal vector from the surface of the target board can be calculated by using the cross product of the differences of two subsets of the points [93]:

$$\vec{N_P} = (a, b, c) = (P_{P2} - P_{P1}) \times (P_{P3} - P_{P1}) \tag{22}$$

Once the normal vector components a, b, and c are calculated, they then can be used in the plane equation.

$$0 = a\,(x - x_{P2}) + b\,(y - y_{P2}) + c\,(z - z_{P2}) \tag{23}$$

The line equation of the anticipated intercept between the laser beam can be calculated concurrently with the plane equation using θ_a, ϕ, and an arbitrary initial line length, ρ_d, where the end points are defined as in Equations 24, 25, and 28.

$$P_{L1} = (x_{L1}, y_{L1}, z_{L1})\,, \quad P_{L2} = (x_{L2}, y_{L2}, z_{L2}) \tag{24}$$

Where P_{L1} can be calculated by:

$$P_{L1} = [dcos(\theta_b), dsin(\theta_b), 0] \tag{25}$$

and ρ_d is defined as:

$$\rho_d = d_2 + d_3 \tag{26}$$

Now, at this point both d_2, and d_3 are unknown but the total length, ρ_d can be set to any length using an arbitrarily long distance that is guaranteed to intercept the target board coordinates (as long as the vector is pointed towards the target board coordinates). In this environment let ρ_d be 5 meters. The end point of the simulated laser beam can be calculated

with Equation 28 using Equation 27 for the total azimuth of the LiDAR beam θ:

$$\theta = \theta_B + \theta_a \tag{27}$$

$$P_{L2} = [\rho_d cos(\phi) sin(\theta)), \rho_d cos(\phi) cos(\theta)), \rho_d sin(\phi)] + P_{L1} \tag{28}$$

The x,y,z components of P_{L1} and P_{L2} are then used to calculate the vector, V_L, in Equation 29:

$$(P_{L2} - P_{L1}) = [(x_{L2} - x_{L1}), (y_{L2} - y_{L1}), (z_{L2} - z_{L1})] = [x_{VL}, y_{VL}, z_{VL}] = \overrightarrow{V_L} \tag{29}$$

Next the parametric equation of a line is introduced in Equation 30, and then we combine Equation 30 with Equation 23 to get Equation 31 in order to help solve for the intercept point coordinates x_x, y_y, and z_z with respect to the sensor origin:

$$\begin{bmatrix} x_x \\ y_y \\ z_z \end{bmatrix} = \begin{bmatrix} x_0 + at \\ y_0 + bt \\ z_0 + ct \end{bmatrix} = \begin{bmatrix} x_{L1} - x_{VL}t \\ y_{L1} - y_{VL}t \\ z_{L1} - z_{VL}t \end{bmatrix} \tag{30}$$

$$0 = a\,(x_{L1} - x_{VL}t - x_{P2}) + b\,(y_{L1} - y_{VL}t - y_{P2}) + c\,(z_{L1} - z_{VL}t - z_{P2}) = P_i \tag{31}$$

Solving for t in Equation 31 and then placing the determined value of t back into Equation 30 results in the expected intercept coordinates between the target board and the propagation laser beam.

Once the coordinates are calculated the estimated range can be calculated by taking the square root of the sum of squares of the difference between the coordinates and P_{L1}, and adding it to the distance to the reflection point, d, as in Equation 32

$$r(\tilde{x}) = \sqrt{(x_x - x_{L1})^2 + (y_y - y_{L1})^2 + (z_z - z_{L1})^2} + d \tag{32}$$

Additionally, two correction factors can be created from the results of the range test, beam width test, and the gray level test and tested within the range function. Using the data from the gray level test a curve fitting process can be used to find the relationship of the measured offset between target board zero and the sensor reading. This general offset, r_{gray}, is added or subtracted from the estimated range based on the results of the gray level test. The second filter can use the beam spot size as a function of distance, and the vector between the normal vector of target board zero and the beam propagation as seen in Figure 46 to help compensate for a shorter than expected range return due to the angle of the target board as r_{offset}.

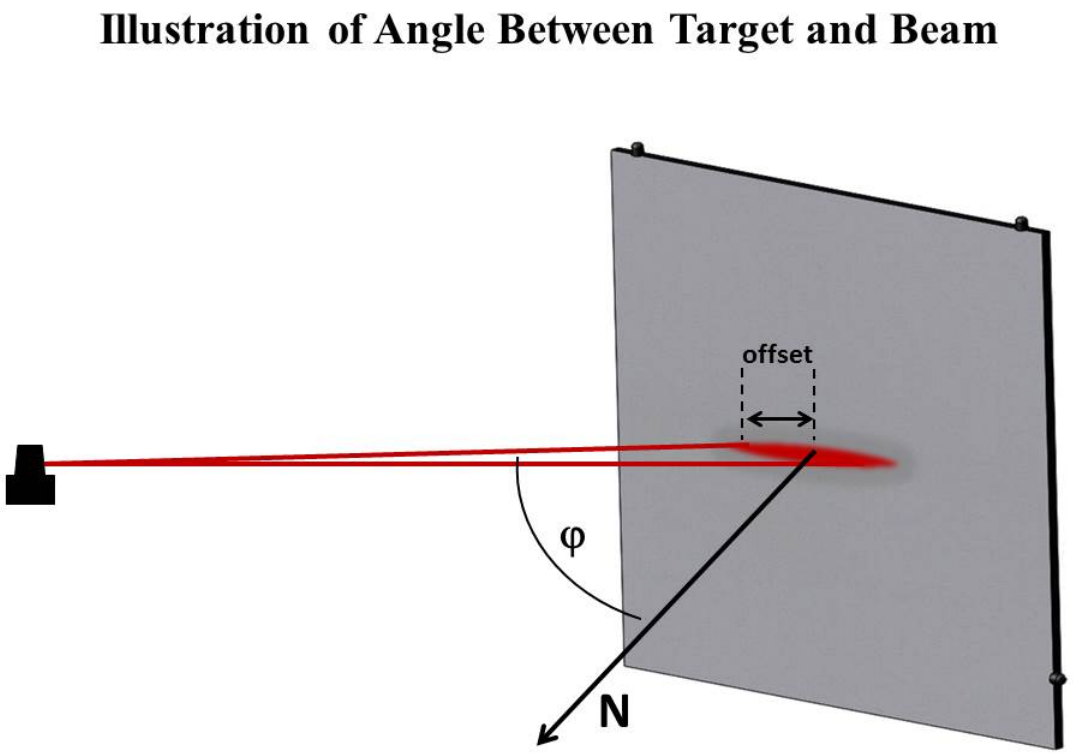


Figure 46. Illustration of Test Scene Showing the Angle between the Beam and Normal Vector of Target

Using this correction term that is a function of both the range to the target and the angle between the normal vector of the target and the beam along with the general offset

discovered by the color of the target we can then calculate a more accurate range estimation as seen in Equation 33.

$$r(\hat{x}) = r(\tilde{x}) - r_{gray} - r_{offset} \tag{33}$$

where $r(\hat{x})$ is the error corrected result of the range function estimate, $r(\tilde{x})$.

Data Collection Procedure.

Using the VICON chamber, target board zero as seen in Figure 18 and the Hokuyo UST-20LX, it was possible to collect truth data from the VICON chamber in scenarios close to those laid out in the Range test, and the Beam Divergence procedures illustrated in Figures 28, and 41 with, and without mirrors in the propagation path. The target board is a 2ft x 2ft x 3/4in square of medium density fiberboard (MDF), where the three target board points are located near three of the corners and situated such that the VICON chamber is less likely to be confused on the orientation of the target board during testing. The location of the VICON markers can be seen on the edges of the target board in Figure 46.

For the first set of data collection, the scenario shown in Figure 28 was set up. The Hokuyo sensor was laid out and marked with a series of range positions at 1m, 2m, 3m, and 4m similarly to the intensity profile scene as seen in Figure 34. Once the VICON chamber and the ROS publishers were all running, the target board was placed at the first range position. The physical range was measured with a measuring tape, the azimuth and elevation to the beam spot on the target board was also measured by measuring the x, y, and z distances to get $\theta = tan^{-1}(\frac{x}{y})$, and $\phi = tan^{-1}(\frac{z}{\sqrt{x^2+y^2}})$. The orientation was initially perpendicular to the beam propagation path. The sensor range, and board positions are recorded for 2,000 data points. Once this was collected, the board orientation was set to an arbitrary but unique position with the target board centered at the marked range position. 2,000 data points are then collected and recorded. One more arbitrary but unique board orientation was set and data was again recorded. This procedure was repeated for the rest

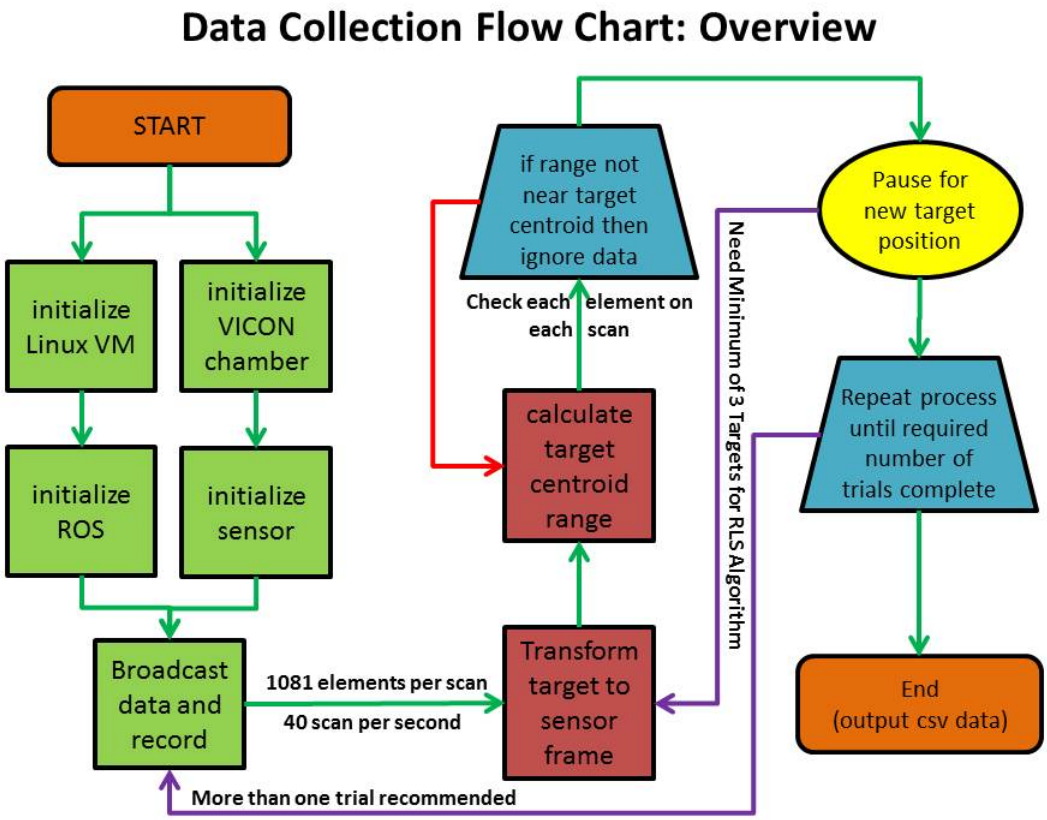


Figure 47. Overall Data Collection Flow Chart

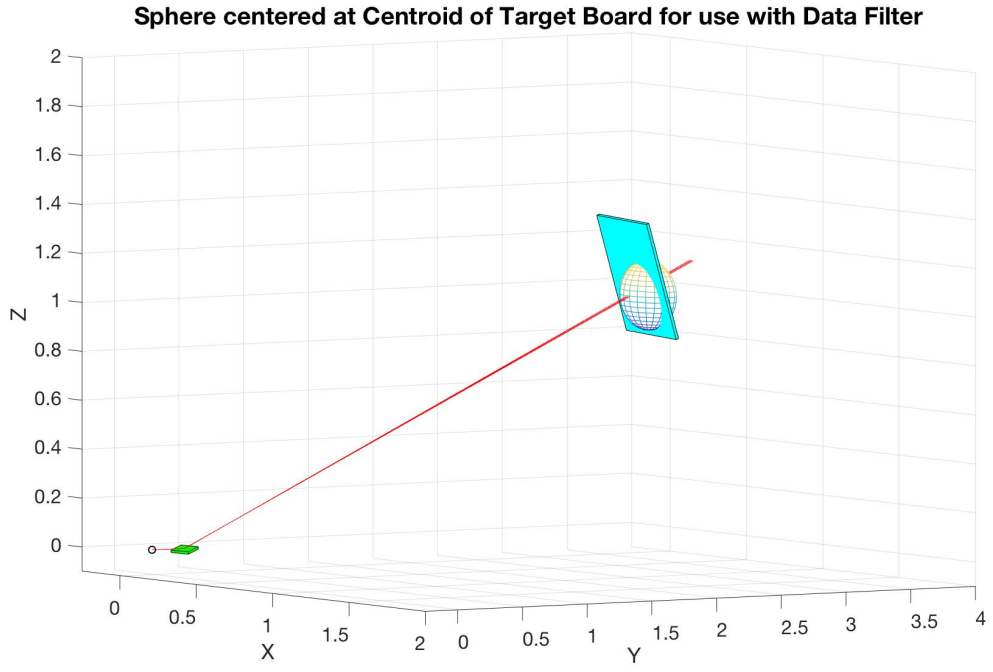


Figure 48. Illustration Showing Centroid Pre-Filter

of the range cases and the data was similarly recorded. An example of the scene showing possible target board zero orientations can be seen in Figure 50.

A small filter is used during data collection by the VICON chamber before the position data is sent out to be recorded. The centroid of the VICON target board object was calculated via Equation 34 and an estimated range is calculated between the origin of the VICON sensor object and the centroid of the target board using Equation 35. If the difference between the range to the target's centroid is greater than a certain tolerance then that measurement is ignored. This filter allows only those measurements that land near the center of the target as seen in Figure 48 and helps minimize erroneous or extremely noisy range measurements.

$$P_c = \frac{\sum_{n=1}^{3}(P_{Pn})}{n} = [P_{cx}, P_{cy}, P_{cz}] \tag{34}$$

P_c is the point in space of the centroid center, $[P_{cx}, P_{cy}, P_{cz}]$ are the x,y,z components

respectively, and P_{Pn} are the target board points P_{P1}, P_{P2}, and P_{P3}. The range to the center of the centroid from the sensor origin is then calculated using

$$r_c = \sqrt{P_{cx}^2 + P_{cy}^2 + P_{cz}^2} \tag{35}$$

where r_c is the estimated range to the centroid point. The set tolerance of the pre-filter was adjusted such that any targets that landed within 10 inches of the center of the target board are used.

3.4 Recursive Least Squares Function

After the estimated range function, and the corresponding error model is developed, which is able to more closely represent the actual measurements from the LiDAR sensor, an algorithm can be developed. This algorithm is needed to estimate the extrinsic parameters of the system, $\Delta\theta$, and ϕ, for each sensor beam element in the scanning array of the LiDAR sensor. A recursive least squares approach is undertaken due to the account of the expected noise in the measurements, whereas in an ideal environment with no noise an exact solution could be modeled using a more simple approach.

Overview.

The Recursive Least Squares (RLS) function uses the methodology as seen in the flow chart in Figure 49 to iteratively converge on a possible solution for the azimuth and elevation angles from the point of deflection of the sensor beam using an initial guess that is with in a given tolerance. The steps to the algorithm are described here:

1. First pick approximate values for the azimuth, elevation, distance to reflection point as seen in Figure 50.

2. Next calculate the estimated range using the ranging function described in Equations

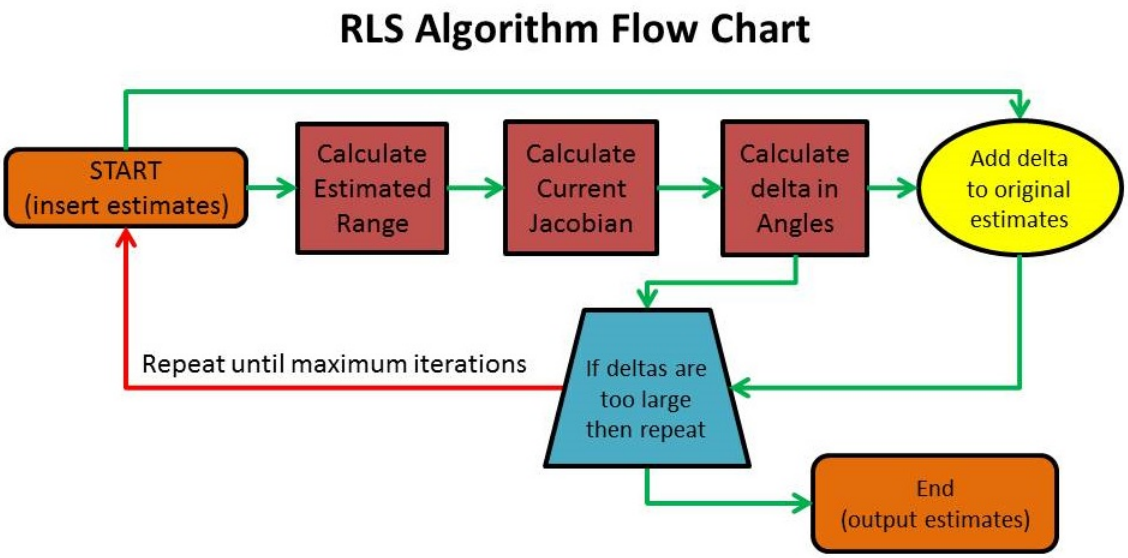


Figure 49. RLS Algorithm Flow Chart

21 through 33.

3. Calculate the current range Jacobian, H, based off of the target board position using either Equation 36 or Equation 37 depending on whether or not the beam deflection point distance, d, is being estimated. For ease of implementation, these are calculated numerically by inducing small perturbations in the range function parameters through $\delta\theta$, $\delta\phi$, and δd.

$$H = \begin{bmatrix} \frac{\partial r}{\partial \theta_1} & \frac{\partial r}{\partial \phi_1} \\ \frac{\partial r}{\partial \theta_2} & \frac{\partial r}{\partial \phi_2} \\ \vdots & \vdots \\ \frac{\partial r}{\partial \theta_n} & \frac{\partial r}{\partial \phi_n} \end{bmatrix}, \begin{bmatrix} \frac{\partial r}{\partial \theta_n} \\ \frac{\partial r}{\partial \phi_n} \end{bmatrix} \simeq \begin{bmatrix} \frac{r_n(\hat{\theta}+\delta\theta,\hat{\phi})-r_n(\hat{\theta}-\delta\theta,\hat{\phi})}{2\delta\theta} \\ \frac{r_n(\hat{\theta},\hat{\phi}+\delta\phi)-r_n(\hat{\theta},\hat{\phi}-\delta\phi)}{2\delta\phi} \end{bmatrix} \tag{36}$$

$$H = \begin{bmatrix} \frac{\partial r}{\partial \theta_1} & \frac{\partial r}{\partial \phi_1} & \frac{\partial r}{\partial d_1} \\ \frac{\partial r}{\partial \theta_2} & \frac{\partial r}{\partial \phi_2} & \frac{\partial r}{\partial d_2} \\ \vdots & \vdots & \vdots \\ \frac{\partial r}{\partial \theta_n} & \frac{\partial r}{\partial \phi_n} & \frac{\partial r}{\partial d_n} \end{bmatrix}, \begin{bmatrix} \frac{\partial r}{\partial \theta_n} \\ \frac{\partial r}{\partial \phi_n} \\ \frac{\partial r}{\partial d_n} \end{bmatrix} \simeq \begin{bmatrix} \frac{r_n(\hat{\theta}+\delta\theta,\hat{\phi},\hat{d})-r_n(\hat{\theta}-\delta\theta,\hat{\phi},\hat{d})}{2\delta\theta} \\ \frac{r_n(\hat{\theta},\hat{\phi}+\delta\phi,\hat{d})-r_n(\hat{\theta},\hat{\phi}-\delta\phi),\hat{d}}{2\delta\phi} \\ \frac{r_n(\hat{\theta},\hat{\phi},\hat{d}+\delta d)-r_n(\hat{\theta},\hat{\phi},\hat{d}-\delta d)}{2\delta d} \end{bmatrix} \tag{37}$$

where $\hat{\theta}$, $\hat{\phi}$, and $\hat{d}$ are the current estimates of θ, ϕ, and d, and H is the Jacobian matrix of the partial differentials relating the change in θ, ϕ, and d to the estimated range value.

4. Now calculate the $\Delta\theta$, $\Delta\phi$, and/or Δd from the current estimates and the error between the estimated range and the measured range from the sensor using either Equation 38 (no Δd) or Equation 39 (with Δd) where $r(\hat{x})$ are the error corrected range measurements given by the sensor and $(z - r(\hat{x}))$ are the residuals between the estimated range and the sensor measured range.

$$\Delta x = \begin{bmatrix} \Delta\theta \\ \Delta\phi \end{bmatrix} = \left(H^T H\right)^{-1} H^T \left(z - r(\hat{x})\right) \tag{38}$$

$$\Delta x = \begin{bmatrix} \Delta\theta \\ \Delta\phi \\ \Delta d \end{bmatrix} = \left(H^T H\right)^{-1} H^T \left(z - r(\hat{x})\right) \tag{39}$$

5. Take $\Delta\theta$, [incremental correction in ϕ in the current iteration of the RLS algorithm]$\Delta\phi$, [incremental correction in d in the current iteration of the RLS algorithm]Δd and add it to the original estimates of θ, ϕ, and d to get a new estimate, $\hat{\theta}_{new}$, $\hat{\phi}_{new}$, and $\hat{d}_{new}$:

$$\begin{bmatrix} \hat{\theta}_{new} \\ \hat{\phi}_{new} \\ \hat{d}_{new} \end{bmatrix} = \hat{x}_{n+1} = \hat{x}_n + \Delta x = \begin{bmatrix} \hat{\theta} \\ \hat{\phi} \\ \hat{d} \end{bmatrix} + \begin{bmatrix} \Delta\theta \\ \Delta\phi \\ \Delta d \end{bmatrix} \tag{40}$$

6. If $\Delta\theta$, $\Delta\phi$, Δd, or Δx are larger than a desired tolerance then repeat the previous steps using the new estimates calculated in step 5.

Some variants of the RLS algorithm use a weighting scheme[94] to weight different components of the Jacobian, H, when calculating for Δx in Equations 38 and 39, which

can now be written to include the weighting matrix, W:

$$\Delta x = \begin{bmatrix} \Delta\theta \\ \Delta\phi \\ \Delta d \end{bmatrix} = \left(H^T W H\right)^{-1} H^T W \left(z - r(\hat{x})\right) \tag{41}$$

where the weighting matrix, W, is defined as:

$$W = \begin{bmatrix} \frac{1}{\sigma_1} & 0 & \dots & \dots & 0 \\ 0 & \frac{1}{\sigma_2} & 0 & \dots & 0 \\ \vdots & 0 & \frac{1}{\sigma_2} & & 0 \\ \vdots & \vdots & & \ddots & 0 \\ 0 & \dots & \dots & 0 & \frac{1}{\sigma_n} \end{bmatrix} \tag{42}$$

and uses the inverse variance of the sensor measurements, $1/\sigma$, of each of the targets. By weighting the solution for Δx, the algorithm may be able to take into account the effects of an extremely noisy target, which then allows the solution to focus on the measurements with less noise to converge to desired parameters. A large variance in $1/\sigma$ in one of the components in the weighting matrix artificially lowers the corresponding component in the Jacobian, H, for that target. By lowering the components in H, the iterative steps for the solution in Δx is not as affected by the noisy target, and therefore can converge using targets with less noise.

Another variant of the RLS algorithm extends the size of the Jacobian, H, by an additional set of parameters[94] to enable the iterative solution of the RLS algorithm to converge based on the parameters in H in conjunction with the new parameters in ϵ as seen in:

$$\Delta x = \left[\begin{bmatrix} H & 0 \\ 0 & \epsilon \end{bmatrix}^T \begin{bmatrix} H & 0 \\ 0 & \epsilon \end{bmatrix} \right]^{-1} \begin{bmatrix} H & 0 \\ 0 & \epsilon \end{bmatrix} \begin{Bmatrix} z - r(\hat{x}) \\ \delta\epsilon \end{Bmatrix} \tag{43}$$

The extended Jacobian parameters represented by ϵ are specific to the implementation and may not result in an effective solution if an appropriate one cannot be identified. The goal is to use a definition of ϵ in such a way that the components in H does not rely on ϵ, but ϵ can be dependent on the the components of H. Deriving an ϵ in this way allows the RLS algorithm to converge to unrealistic results in ϵ in an attempt to converge on better results based on H. $\delta\epsilon$ now acts as a garbage collector for the junk terms resulting from ϵ.

RLS Function Assumptions.

For the RLS algorithm to run smoothly, as implemented here, certain assumptions are given:

- The initial estimates of $\hat{\phi}$ and $\hat{\theta}$ are within 10° of the true angles, and $\hat{d}$ is within 5mm within the true value to prevent the algorithm from converging on local minimums not near the true value.
- The range function is correct in Equations 21 through 33.
- The variety of target board orientations allow for enough excitation in the range Jacobian in Equations 36 and 37 to prevent a near zero matrix result.
- No two combinations of $\hat{\phi}$, $\hat{\theta}$, and $\hat{d}$ will produce the same exact range measurement within the desired residual tolerance used in Equations 38 or 39.

Data Collection Procedure.

With an initial setup similar to Figures 41 and 42 a mirror is placed in the path of propagation with an arbitrary azimuth, elevation, and distance from sensor. The mirror will not be moved for this procedure. Next, target board zero is stationed at an arbitrary orientation and length away such that the beam intercepts the target within a tolerance to the center of mass of the target as illustrated in Figure 48. Data is collected and then the

target is set to a new unique and arbitrary orientation. This is repeated until a sufficient amount data from a sufficient number of orientations are collected as seen in Figure 50. This collection is repeated across 10 independent trials with the same sensor, and mirror positions. The collections were considered independent in that the target board positions were different positions for all orientation within a data collect, and across all 10 trials. The results of the repeatability are then measured across the 10 trials and evaluated.

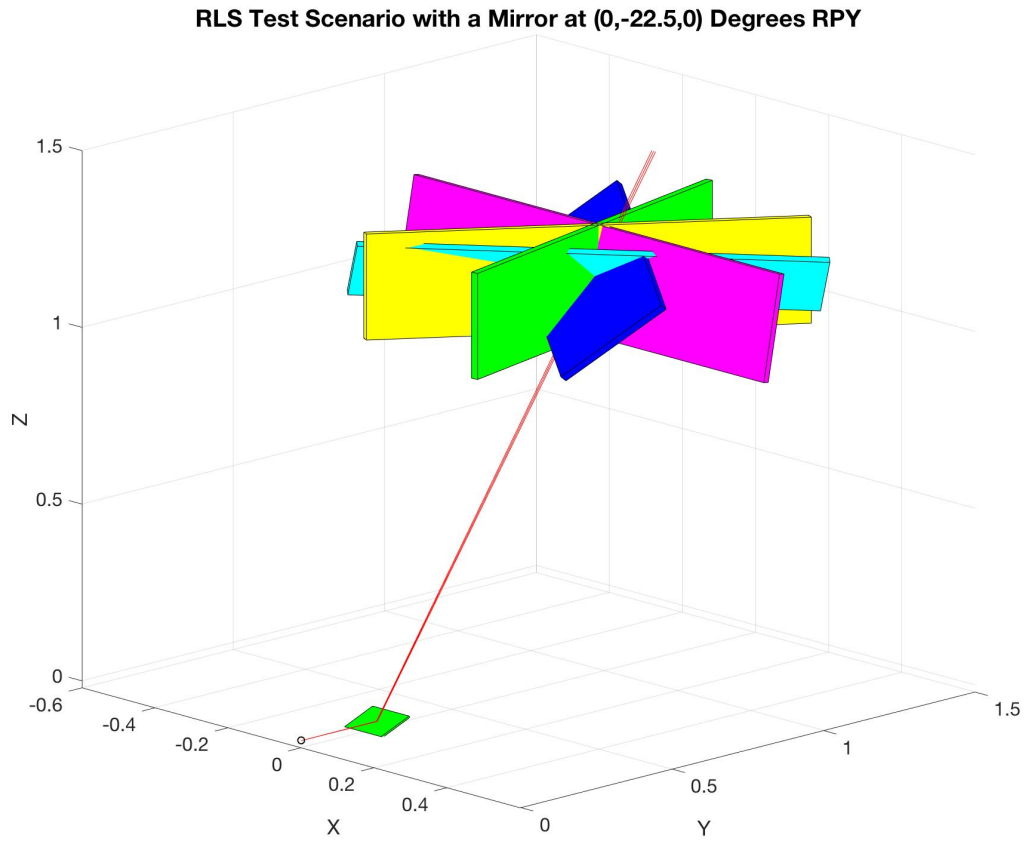


Figure 50. Illustration of Target Positions for RLS Test from a Mirror Relative to Laser Source

Error In RLS Results.

As a conclusion to the sensor characterization, range function, and RSL algorithm testing, a final test was designed to indicate the accuracy of the calibration efforts. A similar

experiment to the scene in Figure 50 was designed. In this instance, a small circle the size of a VICON marker is laid out in the surface of the target board as shown in Figure 51, and indicated by the yellow circle. The VICON marker is 12mm in diameter, and by using the VICON chamber the exact location of the circle can be tracked. The previous repeatability test using 10 trials was repeated, but in this test the target board was placed so that the beam always falls within the center of the yellow circle on the target board. The target board positions were limited between a range of 1.5m to 2.0m meters from the sensor. At a range beyond 2.0m the sensor beam pattern was too large to fully fit inside the target circle.

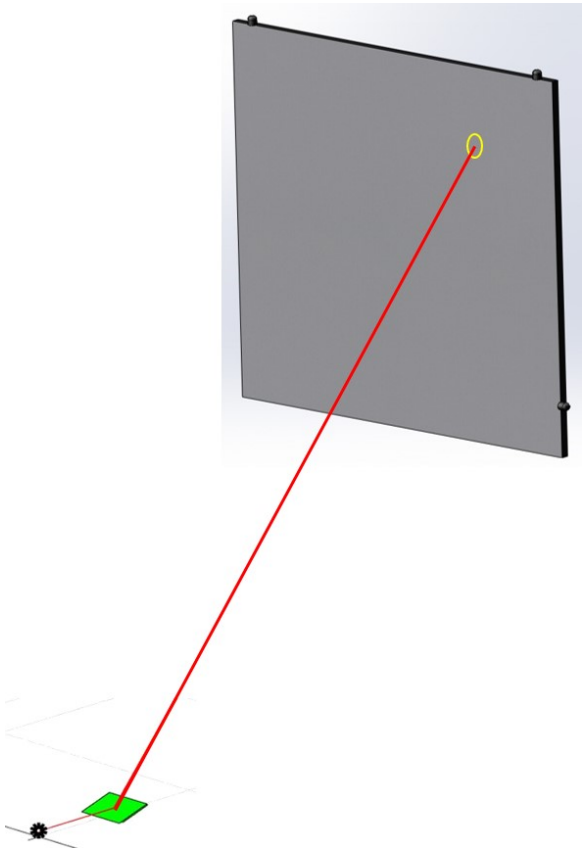


Figure 51. Illustration of Target Location to Measure Error In Calibration Results

3.5 Algorithm to Implement Transformation

Overview.

The Hokuyo UST-20LX Scanning Laser Range sensor produces a data output structure as seen in Figure 23, that includes a range measurement, and intensity array across 1081 elements. Each element is assigned a sequence number on the output and correlates with the $\frac{1}{4}$ degree increments in the angle between -135° and 135° and is assumed to be consistent. This data must be converted to a coordinate system to perform mapping or position estimation within the environment. Assuming there are no highly reflective surfaces, the return range measurement can be converted to a two dimensional coordinate system, holding z constant, using Equation 25 where θ_b are the incremental angles between -135° and 135° and where d is originally the measured reflection point distance we can replace it with the measured range values to get Equations 52.

This equation can be used without any prior knowledge of the environment, all that is required are azimuth of each beam, and the range measurement itself. All the points can be gathered and plotted as a point cloud as seen in Figure 52, which illustrates a 2D planar view of the environment that is highly dependent on the orientation of the sensor itself.

Once highly reflective surfaces are added to the propagation path of the sensor, the previously mentioned 2D planar view is no longer valid. Additional information is now required which can be provided by the RLS algorithm results, and Equation 53 is used to generate a three-dimensional point cloud, PC_1. Equation 53 is used to calculate a transformation at each range element is to translate the sensor range measurement into the new 3D environment using the $\hat{\phi}$, $\hat{\theta}$, θ_b, and $\hat{d}$ taken from the results in the RLS algorithm at each sensor beam element.

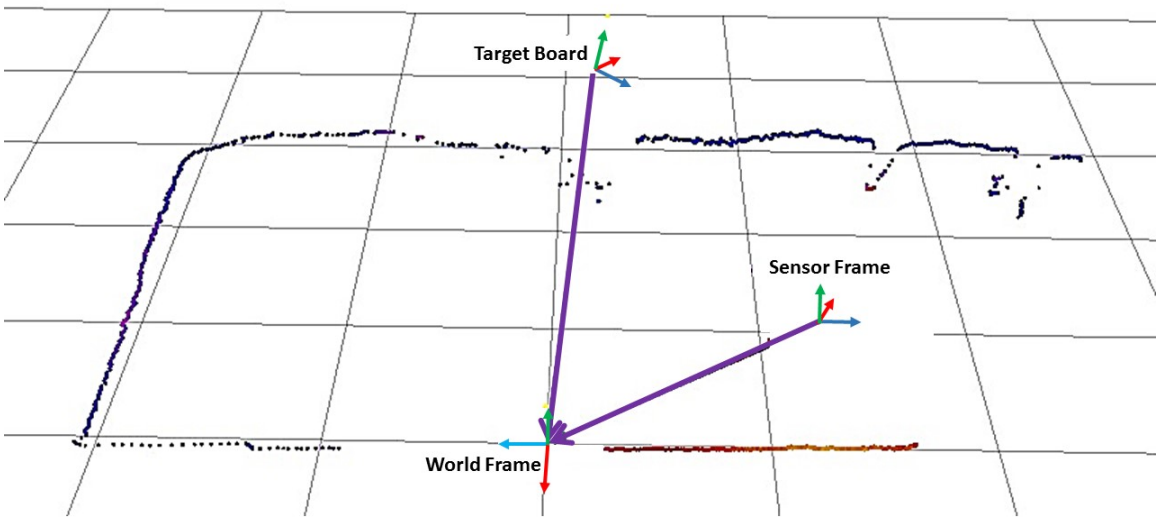


Figure 52. Illustration of 2D Point Cloud View of Returned Ranges from Hokuyo Sensor

Transformation.

Transformations have two major components, the rotation and the translation, to establish the difference between two frames [95]. Equation 46 shows a series of rotation matrices about the x, y, and z axis and then a full 3-axis rotation matrix in Equation 47 that follows a sequence of rotations of z-y-x.

$$R_z = \begin{bmatrix} cos(\theta_3) & -sin(\theta_3) & 0 \\ sin(\theta_3) & cos(\theta_3) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (44) \qquad R_y = \begin{bmatrix} cos(\theta_2) & 0 & sin(\theta_2) \\ 0 & 1 & 0 \\ -sin(\theta_2) & 0 & cos(\theta_2) \end{bmatrix} \quad (45)$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta_1) & -sin(\theta_1) \\ 0 & sin(\theta_1) & cos(\theta_1) \end{bmatrix}, \quad R_x R_y R_z = R_{xyz} \quad (46)$$

$$R_{xyz} = \begin{bmatrix} C(\theta_2)C(\theta_3) & C(\theta_1)S(\theta_3) + S(\theta_1)S(\theta_2)S(\theta_3) & S(\theta_1)S(\theta_3) - C(\theta_1)S(\theta_2)C(\theta_3) \\ -C(\theta_2)S(\theta_3) & C(\theta_1)C(\theta_3) - S(\theta_1)S(\theta_2)S(\theta_3) & S(\theta_1)C(\theta_3) - C(\theta_1)S(\theta_2)S(\theta_3) \\ S(\theta_2) & -S(\theta_1)C(\theta_2) & C(\theta_1)C(\theta_2) \end{bmatrix} \quad (47)$$

where θ_1, θ_2, and θ_3 are the rotation angle about the x, y, and z axis respectively.

The appropriate translation can be calculated using the results of RLS algorithm to find the estimates of the azimuth, and elevation of the reflected beam, and the distance to the reflection point for each element of the Hokuyo scanning laser rangefinder. This can be accomplished using only two rotations involving $\hat{\theta}$ for the rotation about the z-axis and $\hat{\phi}$ for the rotation about the y-axis in a 3-2 rotation sequence to get the rotation matrix in Equation 48. This means that a yaw movement is conducted first and then followed by a pitch maneuver.

$$R_{32} = R_{zy} = \begin{bmatrix} cos(\theta_2)cos(\theta_3) & -cos(\theta_2)sin(\theta_3) & -sin(\theta_2) \\ sin(\theta_3) & cos(\theta_3) & 0 \\ sin(\theta_2)cos(\theta_3) & -sin(\theta_2)sin(\theta_3) & cos(\theta_2) \end{bmatrix} \tag{48}$$

Substituting $\hat{\theta}$ and $\hat{\phi}$ for θ_3 and θ_2 we get

$$R_{32} = R_{zy} = R_{\hat{\theta}\hat{\phi}} = \begin{bmatrix} cos(\hat{\phi})cos(\hat{\theta}) & -cos(\hat{\phi})sin(\hat{\theta}) & -sin(\hat{\phi}) \\ sin(\hat{\theta}) & cos(\hat{\theta}) & 0 \\ sin(\hat{\phi})cos(\hat{\theta}) & -sin(\hat{\phi})sin(\hat{\theta}) & cos(\hat{\phi}) \end{bmatrix} \tag{49}$$

The next step after calculating the rotation using $R_{\hat{\theta}\hat{\phi}}$, is to calculate the corresponding translation. For our purposes the translation between the origin of the sensor beam and origin of the reflected beam is equivalent to the Cartesian coordinates of the beam deflection point as calculated by

$$t_0^1[n] = [d(n)sin(\theta_b(n)), d(n)cos(\theta_b(n)), 0] = P_{L1}(n) \tag{50}$$

where $t_0^1[n]$ is the translation of the n^{th} beam from the origin to the 1^{st} frame, which represent the 3d point cloud to be generated. $d(n)$ is the estimated distance of the nth beam to the

reflection point. In the non-reflected case only one translation is necessary and is calculated across each element of the range sensor output to get the point cloud, PC_0, comprised of an array of 1081 points in Equation 51 where each point is described in Equation 52.

$$PC_0[\ldots] = [P_0(1), P_0(2), \ldots, P_0(1080), P_0(1081)] \tag{51}$$

The 2D planar representation of of the sensor range measurements for the nth element in P_0 is represented as

$$P_0[n] = [r(n)sin(\theta_b(n)), r(n)cos(\theta_b(n)), 0] \tag{52}$$

$r(n)$ is the measured range value on the sensor output. From here, the rotation and translation for each element is implemented to get the Cartesian coordinate components as seen in Equation 53 for a 3D representation of the sensor range measurements.

$$PC_1[n] = \left[R^n_{\hat{\theta}\hat{\phi}} \left(P_0[n] - P_{L1}[n] \right) \quad + \quad P_{L1}[n] \right] = \left[P_{x1}[n], \quad P_{y1}[n], \quad P_{z1}[n] \right] \tag{53}$$

where $R^n_{\hat{\theta}\hat{\phi}}$ is the rotation matrix in a 3-2 sequence for the nth beam, $P_0[n]$ is the coordinate of the nth range element, and $P_{L1}[n]$ is the translation of the nth range element from the sensor beam frame to the new frame. $PC_1[n]$ corresponds to the n^{th} point of the resulting 3D point cloud, PC_1, relating to the n^{th} beam element of the sensor range returns.

3.6 Calibration Procedure

The calibration procedure is the consolidation of the results of the previous steps in the methodology to include the characterization of the sensor, data filtering, and the validation of the range and RLS functions. The algorithm is intended to wrap all the previous steps into a generalized pattern to be able to replicate the process with new and unique external

mirror designs from what is discussed in this thesis.

Calibration Algorithm.

1. Take the fabricated mirror structure and attach to the sensor. Arrange a placement of VICON markers on the structure such that minimal or no erroneous reflections will be seen on the sensor. Create a VICON object from these points using the corresponding chamber and markers.

2. Take initial measurements of each expected reflection point, d, of the laser on the structure. It is preferable to take range measurements before any reflective surfaces are attached to the structure and then to add an offset to the returned range measurements due to the thickness of the reflective surface being added.

3. Run data collection (10k+ data points) in a known environment using the estimated d, ψ, and θ to calculate the initial range vector measurements, r. Eliminate any range scan elements that return with abnormal or inconsistent range statistics.

4. Establish the transformation between the sensor object inside the fabricated mirror structure and the point of origin of the laser source (laser base frame).

5. Convert target board coordinates with respect to the laser base frame.

6. Place target in desired location that allows for the first set of elements in the sensor laser beam to intercept the target near the center of mass. Collect range and position data. Adjust target to a new unique orientation such that the same elements of the sensor laser beam fall near the center of mass of the target. Repeat for a minimum of three target positions.

7. Move target to a new location such that the next series of elements in the sensor beam intercept near the target's center of mass. Collect data.

8. Adjust target to a new unique orientation such that the same elements of the sensor laser beam fall near the center of mass of the target. Repeat for a minimum of three target positions then move to the next series of elements in the sensor beam.

9. Repeat the previous step until all desired elements of the laser scan range (default is 270°) have data collected as intercepted at a minimum of three targets in unique positions.

10. Run RLS algorithm, separately for each beam to calculate the estimated $\vec{\psi}$, and $\vec{\theta}$.

11. Calculated the transformation for each scan element using the measured d, and the calculated ψ, and θ from the RLS algorithm.

12. Implement the transformations calculated and verify the expected output. Original 2-dimensional planar point cloud results from the sensor output should be realized as the 3-dimensional point cloud of the target environment.

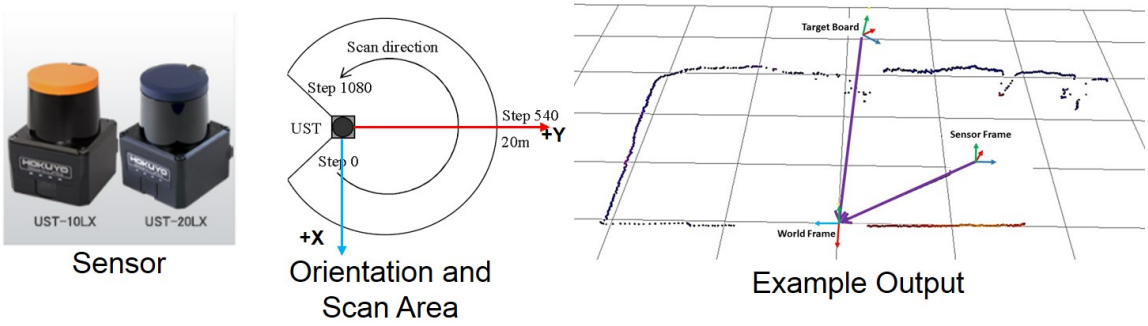
IV. Results

The results presented are the final results of numerous iterations of testing scenarios with the final test scenarios and corresponding results presented here in a logical order. The actual time-line of events for this investigation took a highly circuitous route which often required repeated tests. The reason for this was due to the effect of noise sources, or other optical phenomenon, not anticipated during the original test design phases.

As a precursor to the results presented in this chapter, a summary figure of the sensor, and of its orientation is presented in Figure 53. The sensor used in this research is the Hokuyo UST-20LX scanning laser rangefinder, also known as a 2D scanning LiDAR sensor. The sensor output is comprised of 1081 beam elements oriented along the X-Y plane between -135° to +135° at 1/4° increments to create a potential point cloud similar to the Example Output in Figure 53. At a published scan rate of 40 times per second, this sensor can effectively measure the ranges of a diffuse white target in the operating range between 0.06m - 20.0m. In cases where the range is measured outside of these bounds the sensor may return an error indicator instead of a range value.

In addition to the sensor's 270° sweep angle, the field of view of the receiving sensor is also 270°, which means that for every laser pulse sent out, the sensor listens to the entire sweep area for a return pulse. This is an important characteristic, since this sensor can be affected by multi-path effects or other environmental factors induced by a potential test scenario.

The small size of the sensor, at 130g, illustrates the size of the sensor and the inherent usefulness in small-scale UAV applications. The additional weight of the current 3D printed superstructure, to include the mirrored surfaces, only weight about 300g as discussed later in Section 4.8.



Weight :	130g (0.286lbs)
Output:	1081 element arrays for range, and intensity
Scan Rate:	40/sec
Sweep Angle:	270°
Receiver Field of View:	270° Horizontally

Figure 53. Hokuyo UST-20LX Sensor Summary

4.1 Range Test

Range Test Data Without Mirror.

The first set of data to discuss are the range test results. This test, as described in Section 3.1, is designed to describe the results of the sensor range measurement as compared to the actual range. Figures 54 and 55 illustrate the actual positions of the target boards relative to the Hokuyo laser rangefinder. The target board is a 2ft-by-2ft-by-3/4in medium density fiber (MDF) board. The material at this size is fairly resistant to warping, and if warping occurs it is indicated by cracking or material loss which can be replaced if that should occur during testing. The physical benefit of this helps to minimize warping of the target board plane and minimize potential error caused by this. The target board is tracked by the VICON chamber at the three corners, and the range measurement and estimation functions assume a flat planar surface. If this surface was warped then an unknown range error offset would be introduced into the testing data. The blue triangles in Figures 54 and 55 represent three corners of the target board, the blue dotted line is the initial line vector to calculate the

target intercept point, and the red line is the intercept line calculated by the range function.

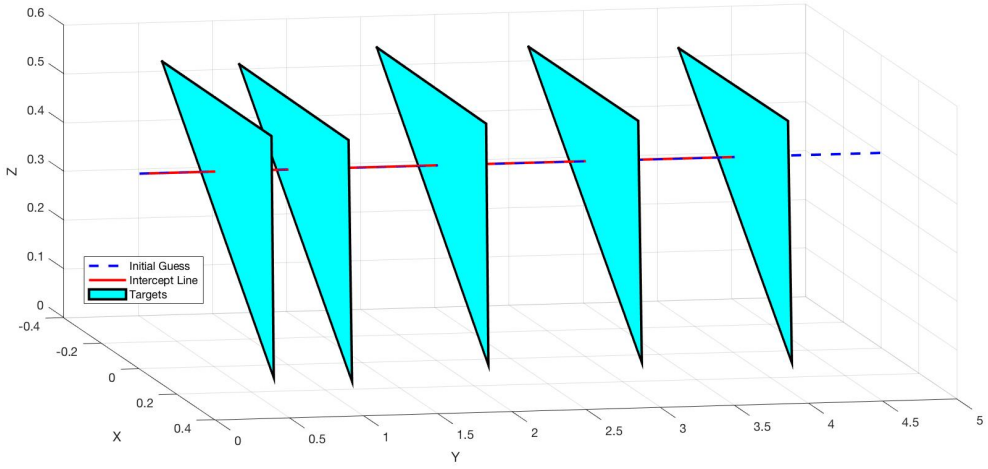


Figure 54. Illustration of Range Test Relative to Laser Source

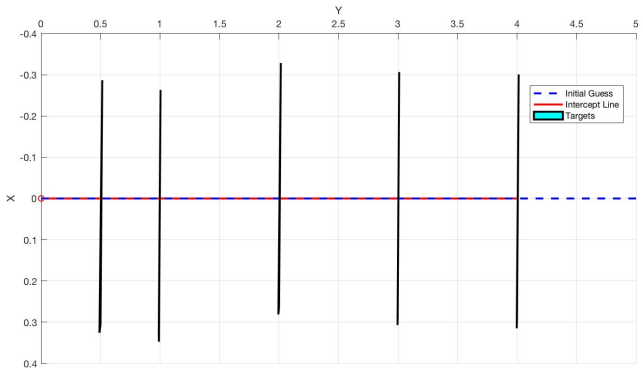


Figure 55. Overhead View of Range Test Relative to Laser Source

The results of the initial range test can be seen in Table 1. The data was collected across three angles from the sensor, -0.25°, 0.00°, and +0.25°and the center beam at 0.00° is illustrated in Figure 56 across all the test ranges. It can be seen that, when the mean error is plotted in Figure 60 with respect to distance, it is similar to a sine function and the magnitude of mean error peaks near 2.00m. Interestingly enough the standard deviation of the mean error between the sensor range return and the actual return is minimized at the 2.00m distance and grows larger everywhere else in the measured distance space.

Table 1. Mean Error and Standard Deviation From Measured Range w/o a Mirror

Range	0.50m		1.00m		2.00m		3.00m		4.00m	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
-0.25°	-0.0017	0.0041	0.0044	0.0043	0.0070	0.0038	0.0032	0.0043	0.0008	0.0047
0.0°	-0.0032	0.0041	0.0026	0.0042	0.0058	0.0039	0.0020	0.0042	-0.0004	0.0046
+0.25°	-0.0036	0.0042	0.0019	0.0042	0.0054	0.0039	0.0016	0.0042	-0.0007	0.0046

Figure 56 illustrates a histogram of the returned range from sensor based on more than 10k measurements at each range. The yellow vertical bar in each figure represents the measured target distance from the sensor origin which is overlaid on top of a black vertical bar indicating the measurement tolerance. The distance was measured with a tape measure and allows for a tolerance of $\pm\frac{1}{8}in$ or $\pm 3.2mm$. The histogram of each distance is estimated as a Gaussian distribution with the given mean and standard deviation as shown in the subplot titles of Figure 56 and is illustrated in each subplot as the red line which encompass the distribution of the sensor range returns appropriately. It shows that the larger the distance the closer the distribution matches a Gaussian curve. At a distance of 0.50m the distribution shown in blue roughly follows the Gaussian fit in red but at a distance of 4.00m the distribution almost follows the Gaussian curve exactly.

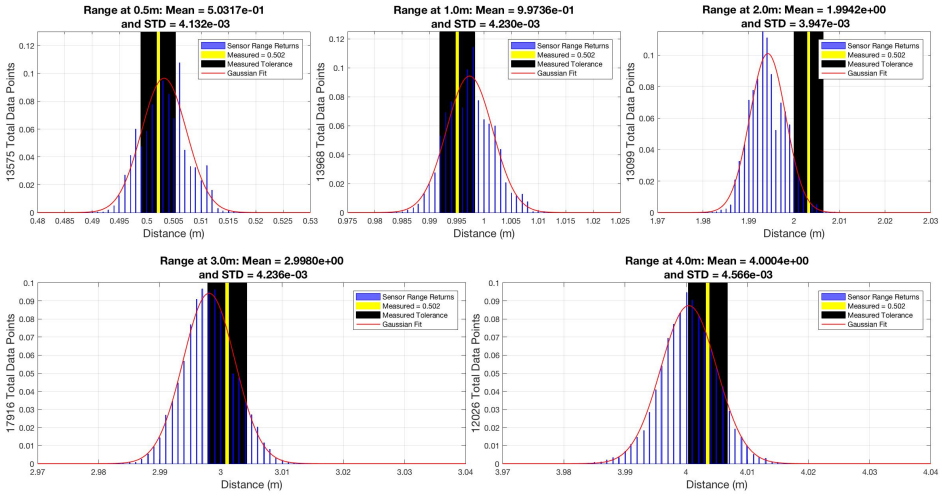


Figure 56. Range Return w/o a Mirror Compared to the True Range

The standard deviation of the target position along each x-y-z component is seen in Ta-

ble 2 at each range respectively. This is proportional to the actuals values used for weighting in the Weighted Recursive Least Squares function discussed later in this section. Due to the extremely low STD shown in Table 2 the estimated range function has the same exact STD due to the fact that the only variation at each range function calculation is the board position. This creates a function return that is practically constant with sub-millimeter consistency at each range distance. Now, assuming the target position is constant an error metric between the estimated range via the range function and the sensor return can be calculated. The physical targets were stationary with no noticeable external environmental disturbances, and the noise in this case is most likely due to the measurement noise of the VICON chamber used to capture the position data.

Later in this section the error correction factor to the range function will be derived. A comparison between the raw range data and the data error corrected is shown side-by-side in Figure 62 and numerically in Tables 4 and 5. The correction factor includes an offset due to the error model describing the estimated error between the sensor range return and the estimated range. It also includes an offset derived from the gray-level test which compares the range return across different shades of gray, and an offset based off of the error introduced to the sensor return as the angle between the sensor and the normal vector of the target board.

Table 2. Standard Deviation of Target Boards for Range Test

	Standard Deviation		
Distance (m)	X	Y	Z
0.50	0.45mm	0.19mm	0.41mm
1.00	1.21mm	1.60mm	3.65mm
2.00	1.25mm	1.36mm	5.62mm
3.00	2.29mm	0.64mm	4.60mm
4.00	3.39mm	0.81mm	5.91mm

Tables 4 and 5 show the mean and standard deviation of the error for each distance. Again, the corrected range values are displayed for ease of comparison. The mean error is

plotted in Figure 60 as the blue circles with a best fit line displayed in red. It is interesting to note that the corresponding standard deviation in Table 4 is identical between the raw data and the corrected values in which both seem to slightly increase with target range.

Table 3. Mean Error and Standard Deviation From Range Function w/o a Mirror

Table 4. Uncorrected for Range Error

Range		**-0.25°**	**0.0°**	**+0.25°**
0.5m	Mean	-0.0037	-0.0052	-0.0055
	STD	0.0041	0.0041	0.0042
1.0m	Mean	0.0043	0.0027	0.0021
	STD	0.0043	0.0042	0.0042
2.0m	Mean	0.0075	0.0065	0.0063
	STD	0.0038	0.0039	0.0039
3.0m	Mean	0.0050	0.0040	0.0040
	STD	0.0043	0.0042	0.0042
4.0m	Mean	0.0050	0.0042	0.0044
	STD	0.0047	0.0046	0.0046

Table 5. Corrected for Range Error

Range		**-0.25°**	**0.0°**	**+0.25°**
0.5m	Mean	0.0015	-0.0000	-0.0004
	STD	0.0041	0.0041	0.0042
1.0m	Mean	0.0045	0.0027	0.0020
	STD	0.0043	0.0042	0.0042
2.0m	Mean	0.0026	0.0014	0.0010
	STD	0.0038	0.0039	0.0039
3.0m	Mean	-0.0004	-0.0016	-0.0019
	STD	0.0043	0.0042	0.0042
4.0m	Mean	0.0014	0.0002	-0.0001
	STD	0.0047	0.0046	0.0046

Range Test With a Mirror.

The next step is to repeat the range test with a mirror placed at about 0.20m distance from the sensor. The mirror deflects the beam at 20° in azimuth and 20° in elevation from the y-axis as illustrated in Figures 57 and 58. In this test scenario, the tolerance of the measured range measurement increased from $\pm 3.2mm$ in the case without a mirror to $\pm 5mm$ due to the possible error in measuring the distance along a diverted path. The same measuring tape was used, but more potential error is introduced when bending the measuring device to follow the sensor propagation path. Again, the blue triangles in Figures 57 and 58 are the positions of the targets based on location of three corners of the target board, the blue dotted line is the initial line vector to calculate the target intercept point with, and the red line is the intercept line calculated by the range function.

The results of this test is similar to the results of the range test conducted without a mirror. The statistics are displayed in Table 6 showing an exception from the original range

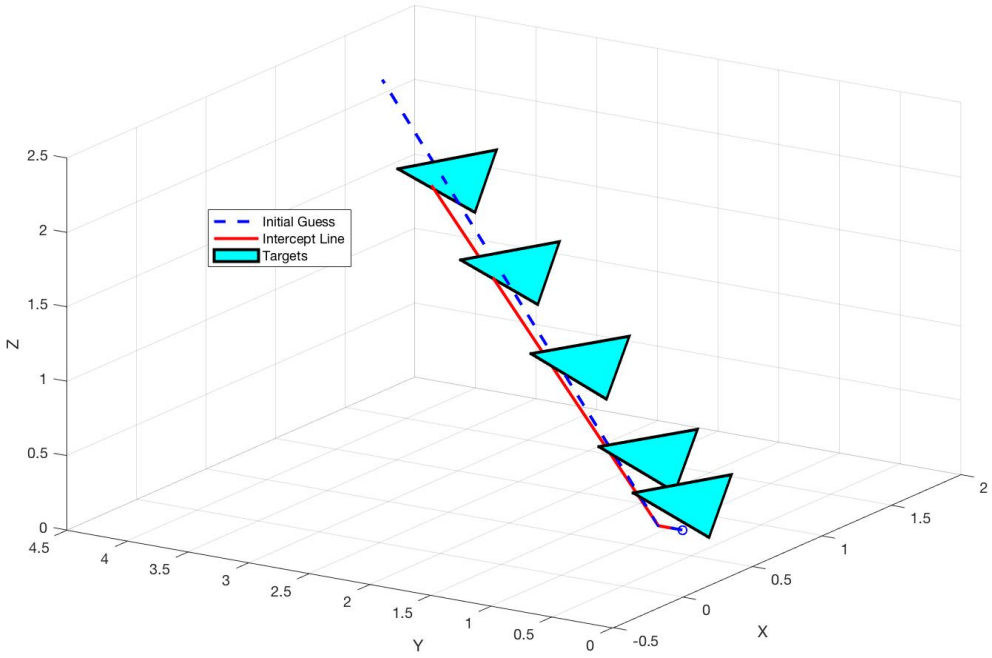


Figure 57. Illustration of Range Test Deflected with Mirror Relative to Laser Source

test, in that the mean error is not as constant as one would expect. The measurements appear to have a different overall shape as can be seen in Figure 61 when compared to Figure 60. These measurements also exhibit a larger standard deviation as a group by 0.5mm. Independent of the overall mean and standard deviation increase, there was also an anomaly in the range return at a distance of 2.00m and the -0.25° scan element. This was seen initially as a bad test collect, but this view was reconsidered after researching what may have caused this anomaly. It was found that at this data point the sensor beam may have been experiencing the edge effects discussed earlier in Section 2.2, and the range return over time was an assortment between two types of returns. About 25% of the range returns were centered at 1.995m which appears as valid measurements, but the next had a Gaussian distribution centered at 1.380m. This may have been caused by edge effects, or there was potentially another target within the the sensor's field of view due to the augmented field of view caused by the mirror. A secondary target may explain the range returns at 1.380m.

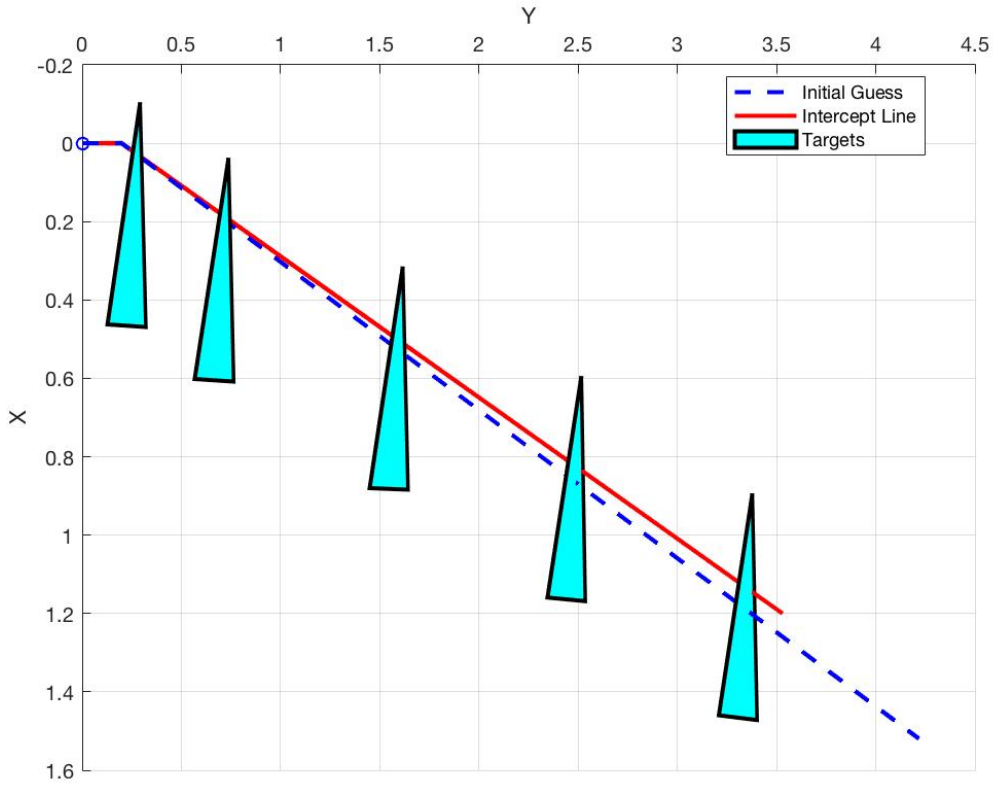


Figure 58. Overhead View of Range Test Deflected with Mirror Relative to Laser Source

Due to the low intensity, the possible path was not traceable with the IR scope used, and no potential secondary target was identified.

Table 6. Mean Error and Standard Deviation From Measured Range with a Mirror

	0.50m		1.00m		2.00m		3.00m		4.00m	
Range	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
-0.25°	-0.0012	0.0046	0.0028	0.0052	0.6272	0.6117	-0.0026	0.0053	-0.0028	0.0060
0.0°	-0.0028	0.0045	-0.0010	0.0049	-0.0015	0.0046	-0.0013	0.0054	0.0014	0.0053
+0.25°	-0.0036	0.0045	-0.0036	0.0045	-0.0035	0.0044	-0.0006	0.0054	0.0032	0.0051

The variation from the range test with a mirror is unlikely to be caused by the noise of the target positions. The STD in the target positions can be seen in Table 7 and it can be noted that the standard deviations in each position are smaller than in the previous range test without a mirror. The only other difference is the addition of the mirror deflecting the beam in the propagation path, and at this point logic dictates that the mirror must be having a perturbation effect on the sensor range measurements. A possible cause is due to the flatness of the mirror which may cause a slight divergence in the sensor beam. If the the beam diverges in an unknown manner, the low intensity areas surrounding the high intensity spot of the beam may be warped in such a way as to allow for greater variation in the range returns.

Table 7. Standard Deviation of Target Boards for Range Test

	Standard Deviation		
Distance (m)	X	Y	Z
0.50	0.47mm	0.24mm	0.39mm
1.00	0.41mm	0.30mm	0.41mm
2.00	0.65mm	0.30mm	0.42mm
3.00	1.19mm	0.48mm	0.68mm
4.00	1.38mm	0.52mm	0.81mm

The range test data is visualized as before in Figure 59 which shows the distribution of range measurements at each target distance. The black bar represents the tolerance of the measured range with the yellow bar indicating the actual measured range via the measuring

tape. The red line is the estimated Gaussian distribution based on the mean and standard deviation of the sensor return. The mean and standard deviation for each range is displayed above each subplot in Figure 59 for reference. At farther distances it can be seen that the range returns follow the Gaussian profile much more closely than at distances less than 1.00m.

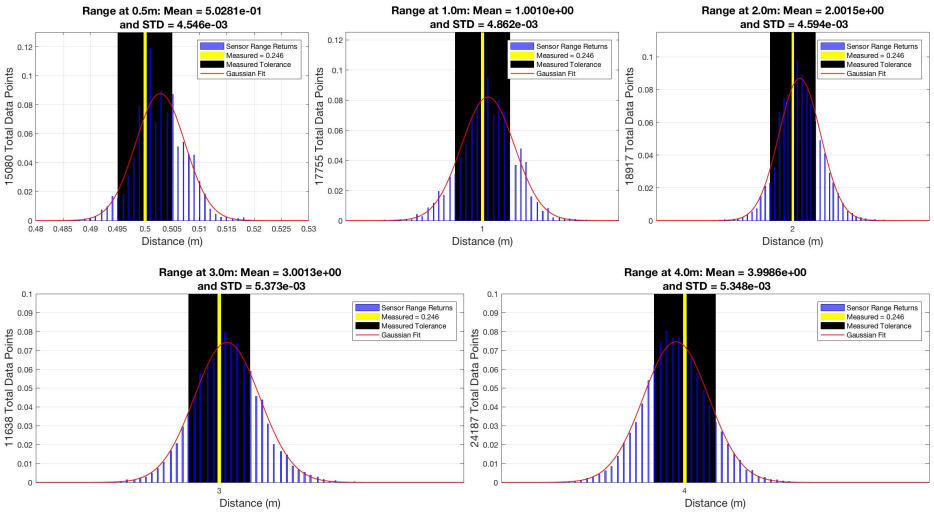


Figure 59. Range Return w/ a Mirror Compared to the True Range

Table 8. Mean Error and Standard Deviation From Range Function with a Mirror

Table 9. Uncorrected for Range Return Error

Range		-0.25°	0.0°	+0.25°
0.5m	Mean	0.0098	0.0081	0.0072
	STD	0.0046	0.0045	0.0045
1.0m	Mean	0.0181	0.0142	0.0113
	STD	0.0052	0.0049	0.0045
2.0m	Mean	0.6355	0.0067	0.0045
	STD	0.6117	0.0046	0.0044
3.0m	Mean	0.0203	0.0215	0.0219
	STD	0.0053	0.0054	0.0054
4.0m	Mean	0.0080	0.0122	0.0137
	STD	0.0060	0.0053	0.0051

Table 10. Corrected for Range Return Error

Range		-0.25°	0.0°	+0.25°
0.5m	Mean	0.0011	-0.0006	-0.0015
	STD	0.0046	0.0045	0.0045
1.0m	Mean	0.0035	-0.0004	-0.0032
	STD	0.0052	0.0049	0.0045
2.0m	Mean	0.6157	-0.0132	-0.0154
	STD	0.6117	0.0046	0.0044
3.0m	Mean	0.0010	0.0022	0.0027
	STD	0.0053	0.0054	0.0054
4.0m	Mean	-0.0082	-0.0040	-0.0024
	STD	0.0060	0.0053	0.0051

After data collection for this test, the range function results were compared against the range test results to get the error results shown in Tables 9 and 10 displaying the mean error,

and the standard deviation of that error for each target range. Keep in mind the anomalous data point at 2.00m target distance, which does not appear to affect the other data points. The standard deviation in Tables 9 and 10 does not change as in the range test without a mirror. The mean error at each range is larger than in the previous case by more than 0.55mm on average. This is interesting because the difference is not a constant deviation but appears to be function of distance as with the magnitude of the standard deviation.

Range Test Correction Factor.

Once the range test for the case with and without a mirror is completed an error model can be calculated due to the error in the sensor returns based on the measured target range as a function of range. Figures 60 and 61 show in blue circles the actual mean error at each distance based on the information in Tables 4 and 9 . An interpolated line is indicated by the dotted line to extrapolate more data points within the given ranges. A best fit can be calculated using the interpolated line which is shown in red. In this case what was found to be most representative of the error is a two term Fourier series shown in Equations 54 and 55 and corresponds to Figures 60 and 61 respectively.

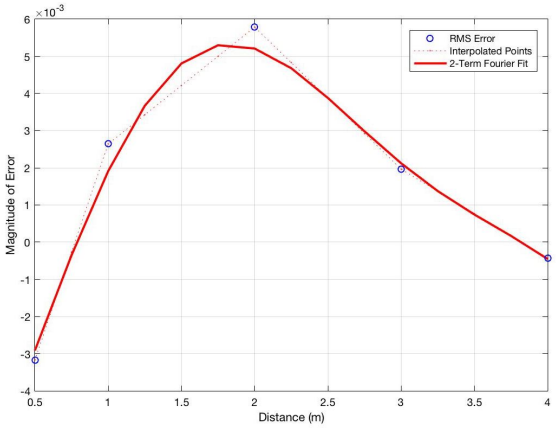


Figure 60. Mean Error At Each Range in No-Mirror Case

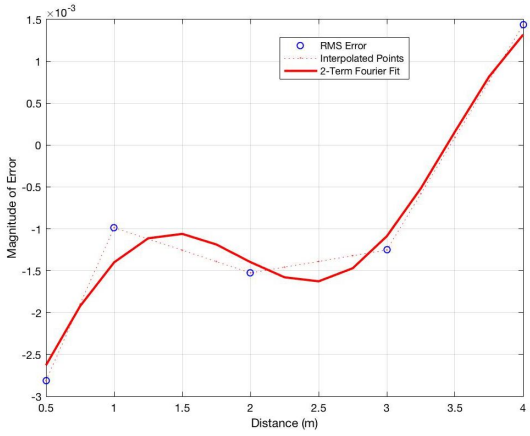


Figure 61. Mean Error At Each Range in Mirror Case

$$E_{r_{nm}} = -2.191 \times 10^{-03} - 4.31 \times 10^{-03} cos(0.897r) + 6.113x10^{-03} sin(0.897r) \\ -1.376 \times 10^{-03} cos(1.795r) + 1.746 \times 10^{-03} sin(1.795r) \tag{54}$$

$$E_{r_m} = -1.543 \times 10^{-03} - 2.148 \times 10^{-03} cos(0.897r) - 8.863 \times 10^{-05} sin(0.897r) \\ -4.445 \times 10^{-04} * cos(1.795r) + 1.49 \times 10^{-03} sin(1.795r); \tag{55}$$

where $E_{r_{nm}}$ and E_{r_m} is the error introduced by the sensor at a given range r for a case without a mirror and with a mirror.

This error model is only representative within a given target range between 0.50m and 4.00m. It is intriguing that introducing a mirror to the sensor propagation path distorts the expected error to an almost completely different solution. The only point where the two error models seem to agree are at the 0.50m mark which shows a mean error near -3.0mm.

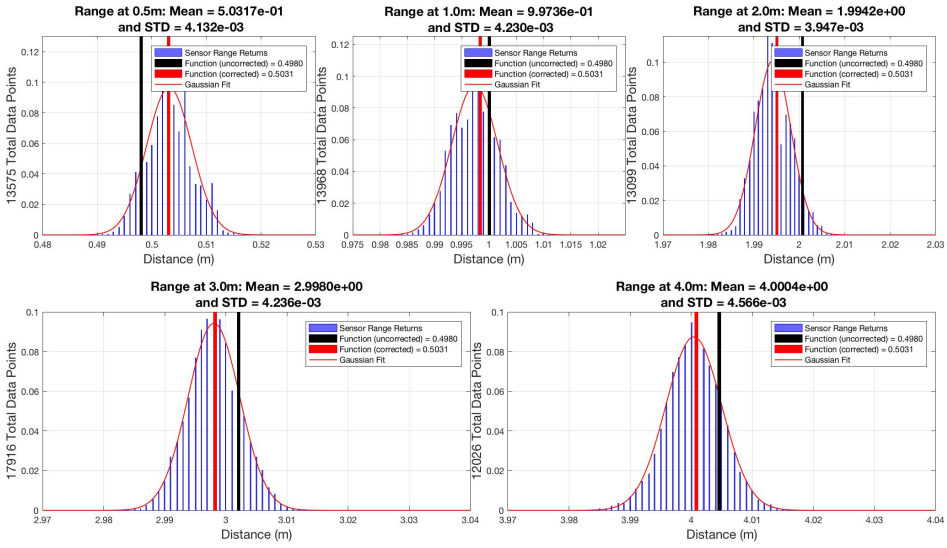


Figure 62. Range Return Compared to the Range Function Return

Additionally, it was possible to visualize the change in the error estimates by using the error correction term in the range estimation function. Figure 62 show the previous displayed range data as a Gaussian distribution. It also shows the original range function

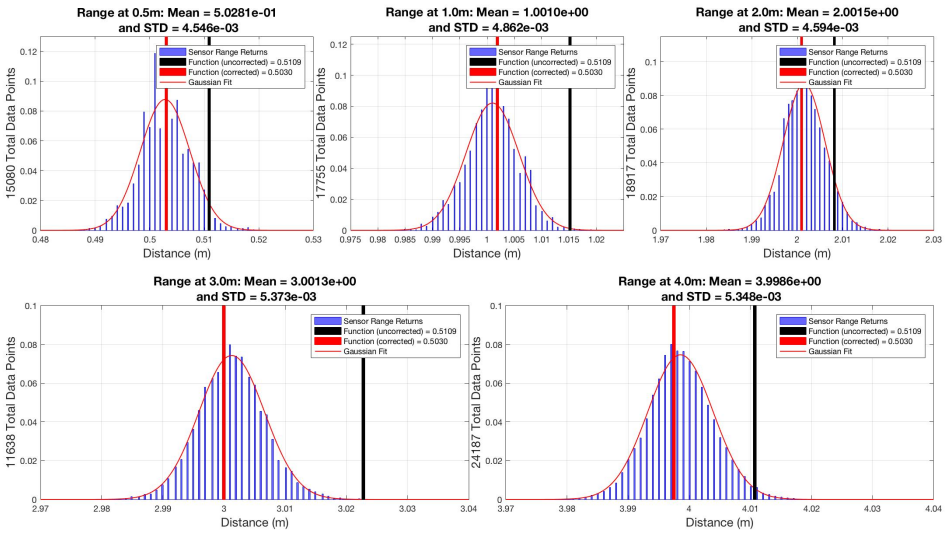


Figure 63. Range Return Compared to the Range Function Return with a Mirror

estimate as a black vertical bar. The range function estimate using the error correction term in Equation 54 is displayed as a red bar for each range. It can be seen that the corrected range function does give a closer estimate to the mean of the ranges returned by the sensor. The same is done for the range test data with a mirror, and can be seen in Figure 63. Similar results showing a closer approximation to the mean of the sensor return is achieved by the estimated range function by using the described error correction in Equation 55. From here then next step in characterizing the sensor is the beam divergence test.

4.2 Beam Divergence Test

The beam divergence test was conducted as described in Section 3.1 and in all practicality the data was collected during the corresponding range tests. Once data was collected for the range test, the external light sources were turned off for maximum visibility with the IR scope and the beam pattern was traced out by hand. The test scene used is shown in Figure 64 for the case without a mirror and the test scene in Figure 65 describe the test case with a mirror. Tables 12 and 13 display the results.

Table 11. Beam Divergence Results

Table 12. Beam Size at Multiple Distance w/o Mirror

Distance (m)	Height (mm)	Width (mm)
0.50	3.2	15.9
1.00	4.0	33.3
2.00	4.8	55.6
3.00	5.6	98.4
4.00	6.3	136.5

Table 13. Beam Size at Multiple Distance w/ Mirror

Distance (m)	Height (mm)	Width (mm)
0.50	3.2	19.0
1.00	4.0	31.8
2.00	5.2	66.7
3.00	7.9	106.4
4.00	9.5	139.7

Figure 64 also depicts an artists rendition of the propagated beam pattern on the target boards. This beam pattern was seen via the use of the IR scope but it had no recording capabilities and the beam pattern needed to be hand traced. Figure 65 also displays an artists rendition o the projected beam pattern being deflected off of a mirror. It can be seen that the intensity across the beam is not of a consistent pattern like the unmolested beam in Figure 64.

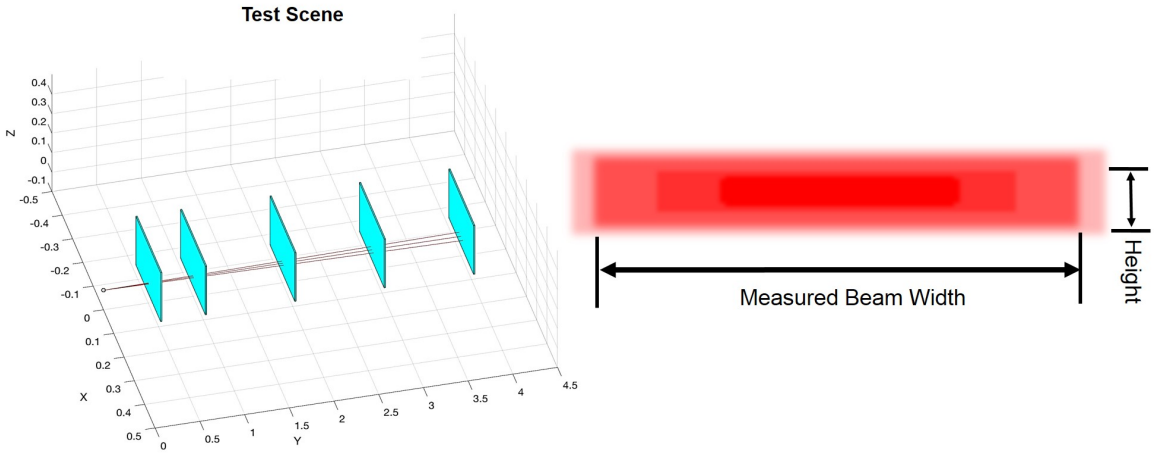


Figure 64. Beam Divergence Test Scene w/o a Mirror

The beam pattern in Figure 65 can be described as having hot and cold locations, the cold location are where the intensity is low and appear almost white in the depiction. This altered beam pattern is most likely due to the flatness of the mirror used in testing. The mirror had a $1/\lambda$ flatness rating but this does not appear to be flat enough to reproduce a uniform intensity pattern. The LiDAR sensor is an incoherent source, which means that all

the photons in the beam are in random phases, and therefore the resulting intensity pattern is unlikely to be caused by constructive and destructive phase interference.

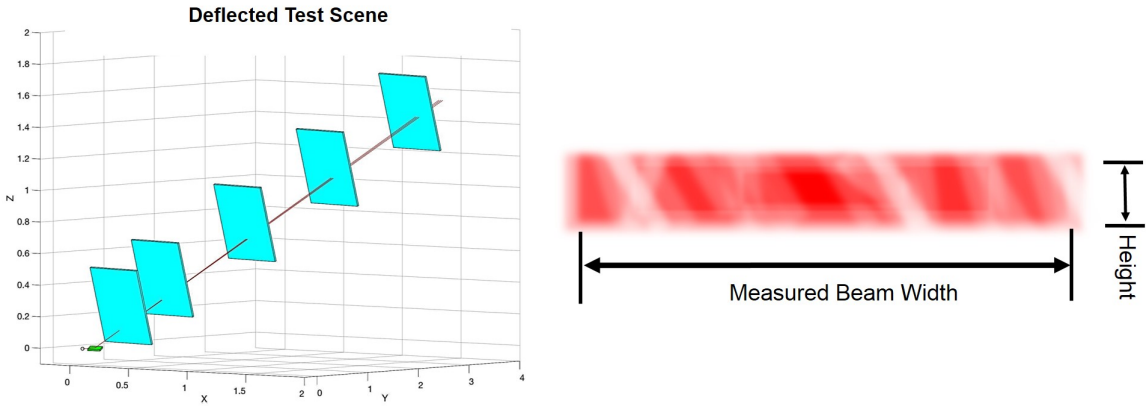


Figure 65. Beam Divergence Test Scene w/ a Mirror

The beam patterns were measured via another measuring tape, this one had a resolution down to $1/32in$ which equates to a tolerance of $\pm 0.4mm$. Keeping this in mind the beam divergence can be calculated using the data in Tables 12 and 13 and Equation 56 to get the average beam divergence listed in Table 14.

$$\Theta = tan^{-1}\left(\frac{h(n) - h(n-1)}{2(d(n) - d(n-1)}\right) \tag{56}$$

where Θ is the angle of divergence, h(n) is the height at the n^{th} distance, and d(n) is the n^{th} distance.

Table 14. Calculated Beam Divergence

	0°× 0° Deflection	20°× 20° Deflection
Height	0.413 mRad	0.900 mRad
Width	17.23 mRad	17.24 mRad

4.3 Angle Test

The actual angle-test scenario can be seen in Figures 66 and 67 from an overhead perspective. The target boards are in light blue with a black boarder. In Figure 66 the targets may just appear to be lines, as it was attempted to eliminate any possible variation in the z-direction to minimize error. From a high vantage point, as in Figure 66, the targets appear to be perfectly aligned with the z-axis. The close-up view in Figure 67 illustrates that there was still a slight variation in the z-component. The dashed blue lines represent the possible intercept vector by the sensor, while the red line represent the calculated intercept location of the farthest target in this instance as calculated by Equations 21 through 32.

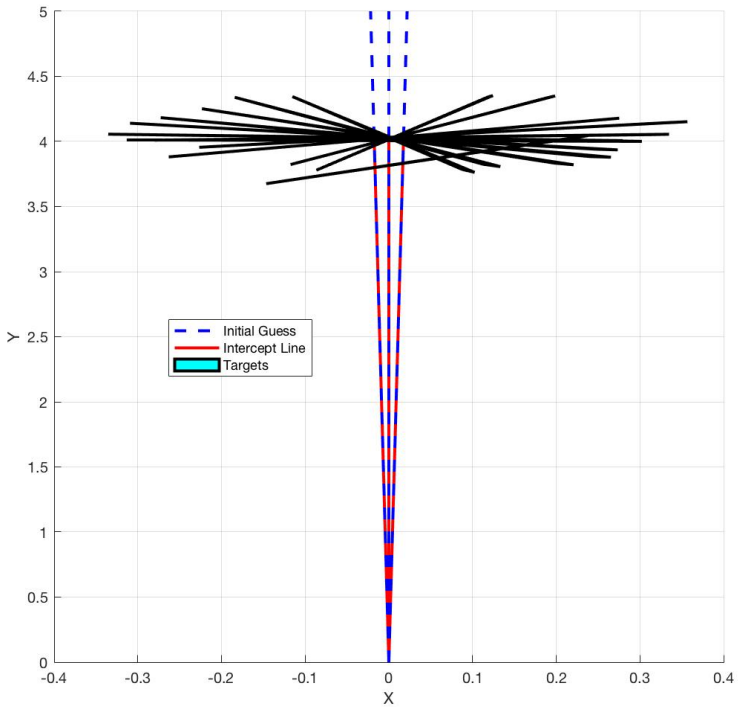


Figure 66. Target Positions for Angle Test

The proposed test scenario in Section 3.1 suggested a number of ranges to test for a more detailed picture of how the returned range is affected by the target boards orientation with respect to the propagating laser beam. Figures 68 through 73 show the sensor range

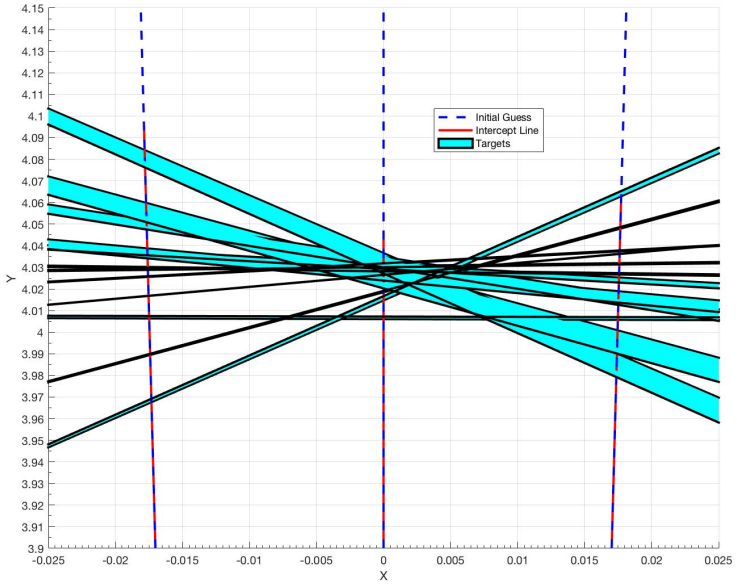


Figure 67. Close View of Target Positions for Angle Test

return in a similar fashion as seen in the range tests. The histogram of of the range returns are presented in red and a Gaussian fit envelops the histogram in blue.

Table 15. Standard Deviation of Target Boards for angle Test

	Standard Deviation (mm)		
Distance (m)	X	Y	Z
0°	8.73	1.61	11.68
5°	4.13	1.67	14.16
20°	4.67	1.44	13.55
30°	4.75	1.50	14.13
45°	99.69	611.5	174.9
60°	5.00	1.74	13.97
80°	5.19	1.41	14.29
-5°	5.26	1.58	14.31
-20°	5.34	1.58	14.43
-30°	5.48	1.56	17.33
-45°	5.68	1.69	17.95
-60°	5.67	1.56	14.52
-80°	5.68	1.58	14.39

The tested angles and the realized angles can be seen in Table 16 while the target po-

sition STD can be seen in Table 15. Looking at Table 15, first to evaluate the position error of the targets it's possible to identify any anomalies. It can be seen that the target at 45° experienced noise about 1-2 orders of magnitude larger than the rest of the target positions which had a x-y-z positional standard deviation of about 100mm, 600mm, and almost 200mm.

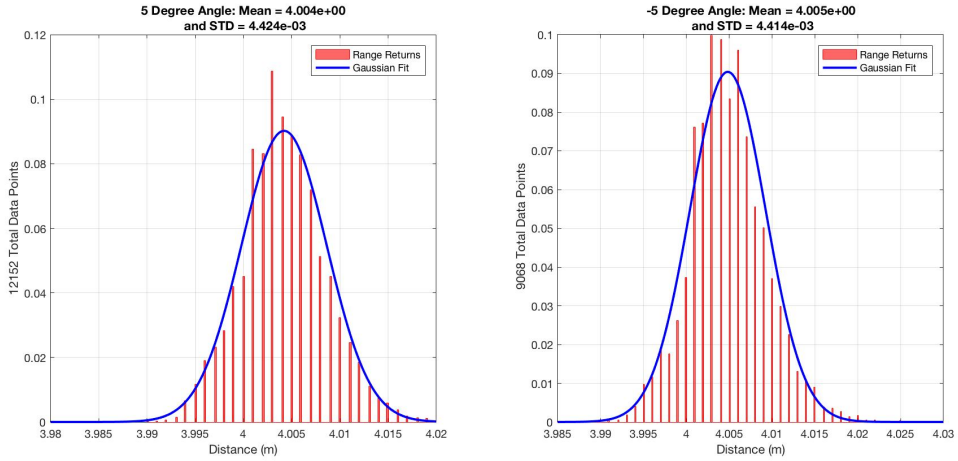


Figure 68. Angle Test Snapshot of Sensor Range Return at ±5°

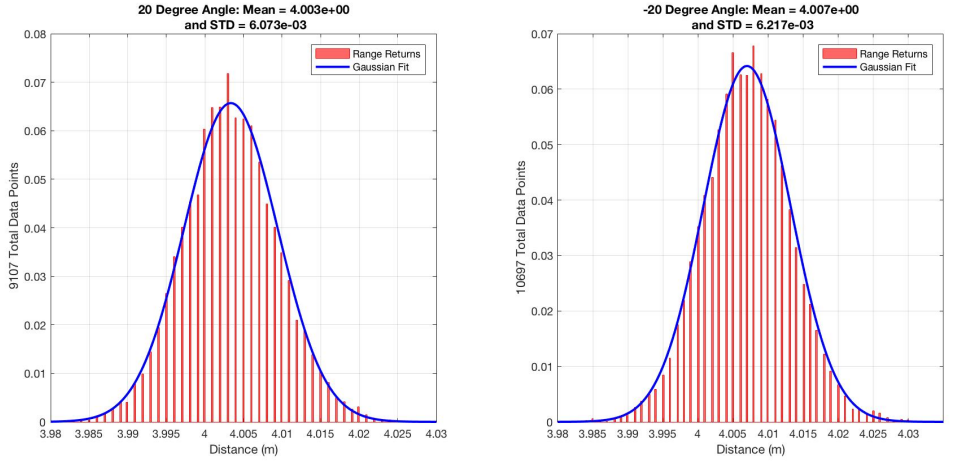


Figure 69. Angle Test Snapshot of Sensor Range Return at ±20°

Table 16 shows the measured mean and standard deviation of the sensor returned range at the different target angles. Looking at the values, it looks like the most important take

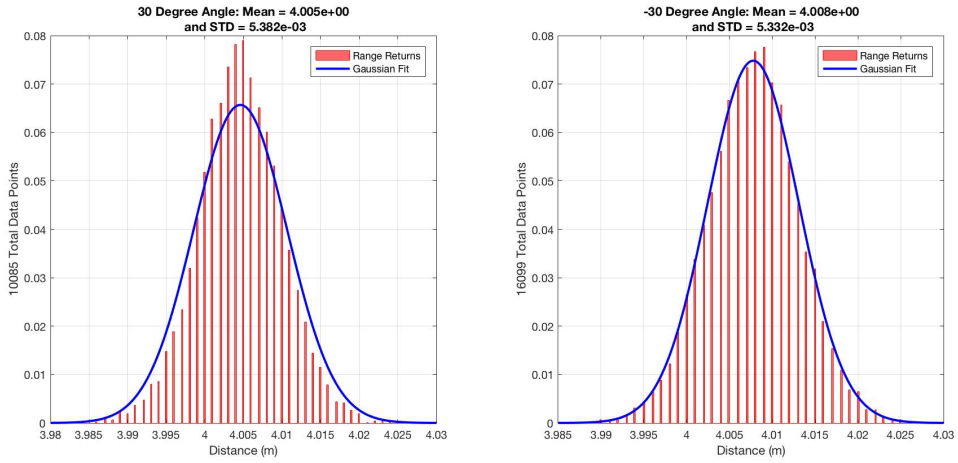


Figure 70. Angle Test Snapshot of Sensor Range Return at $\pm 30°$

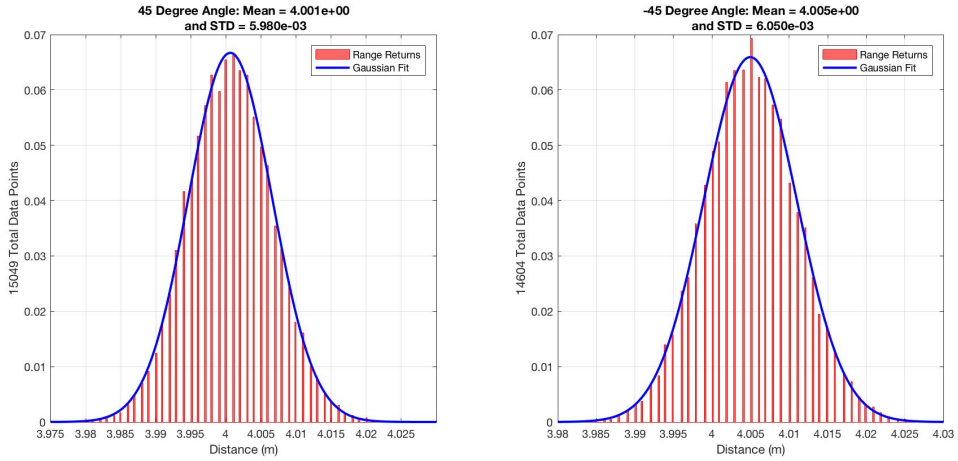


Figure 71. Angle Test Snapshot of Sensor Range Return at $\pm 45°$

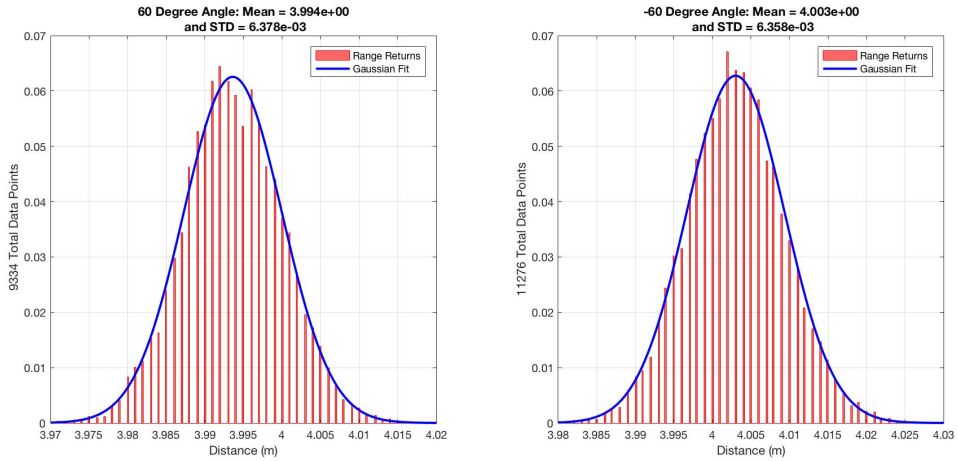


Figure 72. Angle Test Snapshot of Sensor Range Return at $\pm 60°$

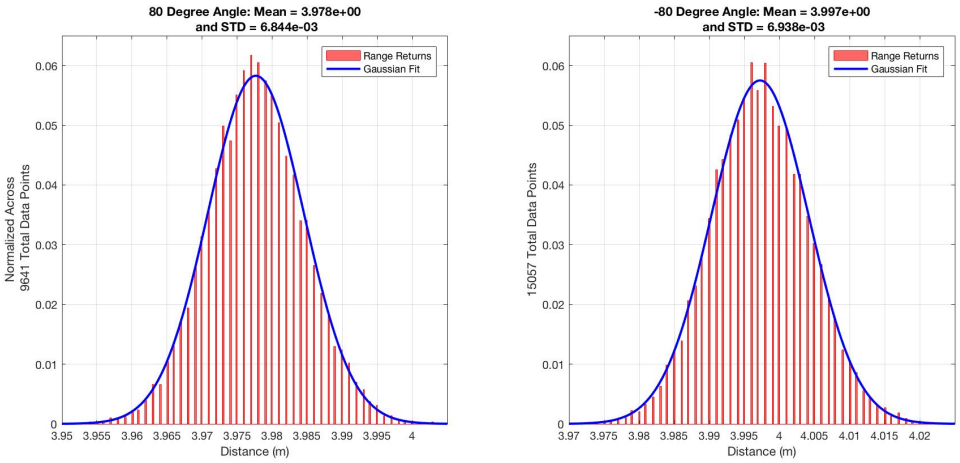


Figure 73. Angle Test Snapshot of Sensor Range Return at $\pm 70°$

away is that the standard deviation of the sensor range return increases slightly as the angle normal to the target board increases away from the sensor beam propagation path.

Table 16. Mean and Standard Deviation of Range w.r.t. Angle Test

Desired Angle	Measured Angle	-0.25°Az		0.0°Az		+0.25°Az	
		Mean	STD	Mean	STD	Mean	STD
0.0°	0.9°	4.0044	0.0048	4.0046	0.0046	4.0049	0.0045
5.0°	4.0°	4.0328	0.0044	4.0315	0.0044	4.0303	0.0044
20°	18.5°	4.0368	0.0060	4.0308	0.0061	4.0249	0.0058
30°	28.9°	4.0362	0.0052	4.0264	0.0054	4.0168	0.0053
45°	43.6°	3.8356	0.0062	3.8196	0.0060	3.8038	0.0059
60°	59.1°	4.0467	0.0061	4.0172	0.0064	3.9881	0.0059
80°	70.0°	4.0598	0.0063	4.0112	0.0068	3.9638	0.0063
-5.0°	-4.7°	4.0280	0.0043	4.0294	0.0044	4.0309	0.0046
-20°	-19.7°	4.0276	0.0061	4.0339	0.0062	4.0402	0.0060
-30°	-30.2°	4.0245	0.0054	4.0347	0.0053	4.0450	0.0055
-45°	-44.7°	4.0233	0.0060	4.0406	0.0061	4.0582	0.0062
-60°	-60.0°	4.0088	0.0060	4.0390	0.0064	4.0698	0.0060
-80°	-70.1°	3.9989	0.0065	4.0469	0.0069	4.0962	0.0065

The next set of figures are presented to illustrate the difference between the sensor range return and the position of the target at each angle. Figures 74 through 79 each compare the positive and negative data points at each set of complimentary angles. For example, Figure 74 shows the target positions at 0°, -5°, and +5°with the sensor range return. The target

positions are in light blue with a black boarder, although some might appear only as a black line which is dependent on how close to absolutely vertical the target was during testing. The sensor output is plotted in the solid blue lines while the dotted blue lines indicate the look angle between the three sensor array elements under test, -0.25°, 0°, and +0.25°azimuth from sensor center. The vertical red dotted lines in each figure indicate the range function estimate to intercept the target board. These lines end at the intercept point near the top of each figure.

Again, please note that Figure 77 does not show the +45° target due to the location of

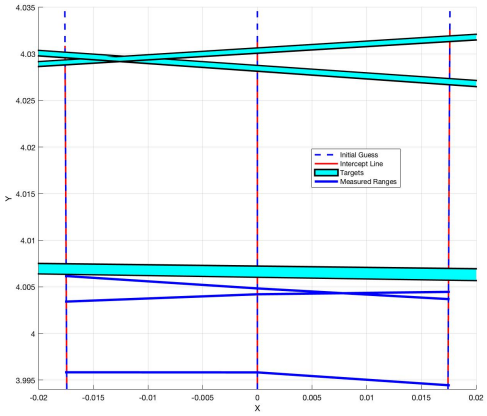


Figure 74. Angle Test: 5°

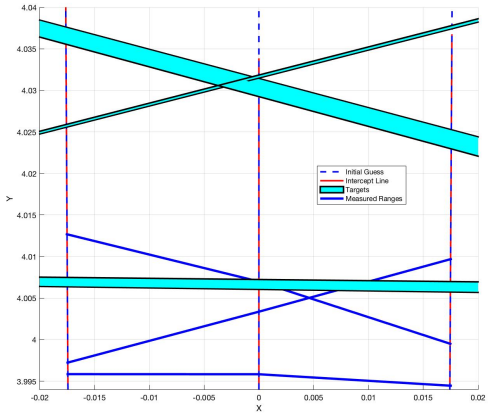


Figure 75. Angle Test: 20°

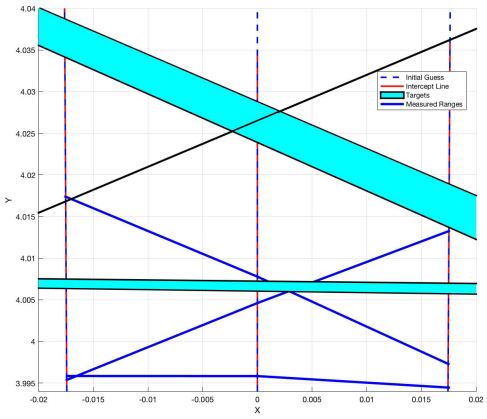


Figure 76. Angle Test: 30°

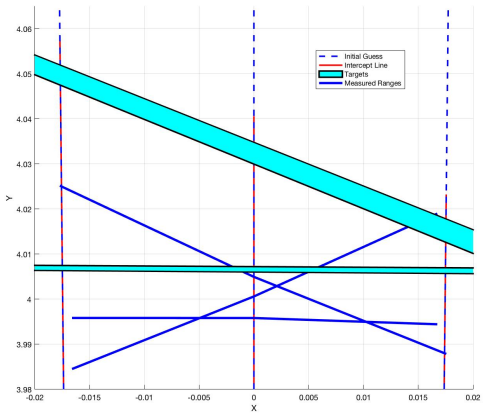


Figure 77. Angle Test: 45°

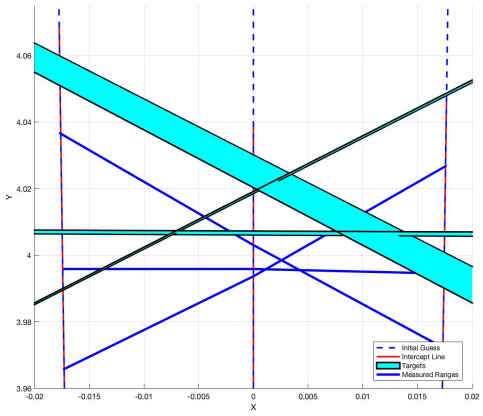


Figure 78. Angle Test: 60°

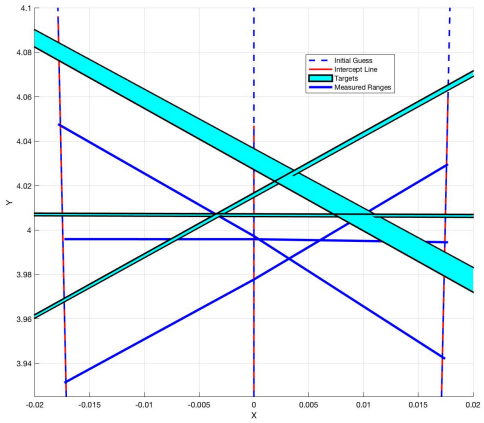


Figure 79. Angle Test: 80°

the target board relative to the intended location, as explained by the large positional STD indicated in Table 15. The possible reason for this large deviation may be due to some positional noise on the VICON markers through the VICON chamber. It is not uncommon for the system to not see the VICON marker with ”enough” IR cameras, and the resulting location will jitter between potential location solutions. This jitter is a result if the VICON system not being able to accurately solve for the position of the VICON markers within the object and tends to flip some of the coordinate pairs of the target.

In addition to the individual set of target board locations, Figures 80 and 81 are introduced to illustrate the overall sense of the pattern of error to the reader. Figure 80 presents the series of sensor range returns as a group. One would expect to see the complimentary angles to cross each other at the same x-axis location, near the 2^{nd} return element. It can be seen that the cross point veer slightly to the left and down. The fact of the crossing points decreasing in distance, moving down the y-axis, is expected as the spot of the sensor beam is likely to land on a closer point on the target as the angle of the target increases. It was, however, unexpected for the cross points to veer left on the x-axis. This may mean that the target boards varied more than intended in center position or that there is something more going on. Figure 81 is also presented as part of the series to illustrate the overall sensor

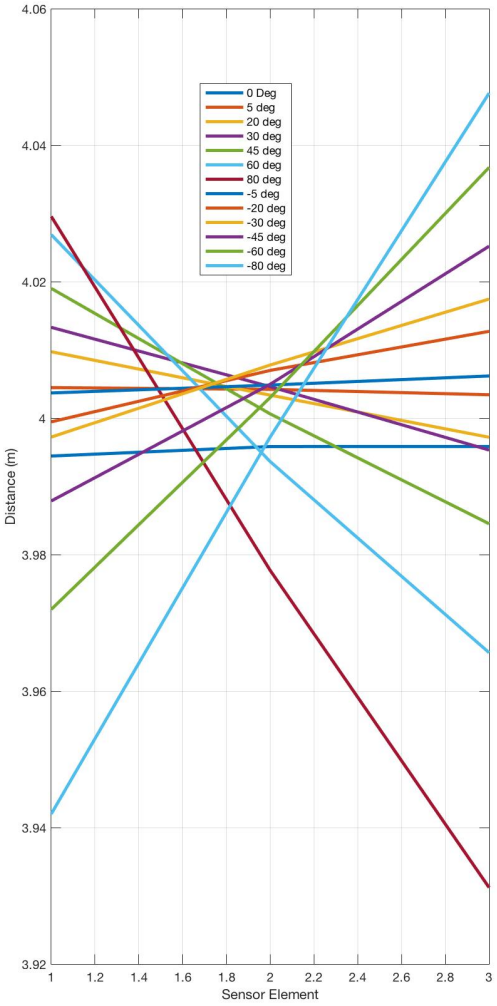


Figure 80. Angle Test Measured Ranges

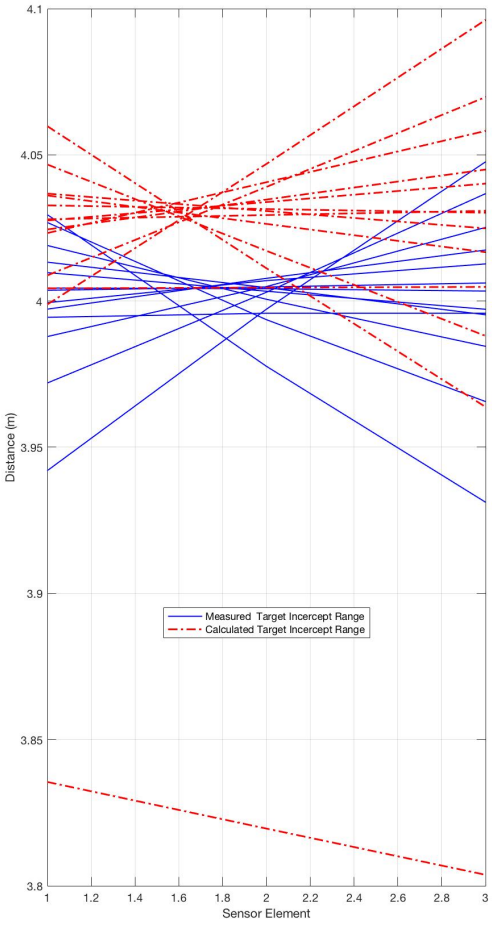


Figure 81. Angle Test Measured Ranges Compared to the Calculated Ranges

measured returns, in blue, with the calculated range function returns highlighted in red. Here it is seen again that there exists a cross point pattern that has veered left in the x-axis which corresponds with Figure 80 and indicates that there must have been an error in target placement to induce this skewed pattern in the data.

Table 17. Uncorrected Mean Error on the Three Center Beams between Range Return and Calculated Target Distances

Estimated Angle	0.0°	5.0°	-5.0°	20.0°	-20.0°	30.0°	-30.0°	45.0°	-45.0°	60.0°	-60.0°	80.0°	-80.0°	RMS
Calculated Angle	0.9°	4.0°	-4.7°	18.5°	-19.7°	28.9°	-30.2°	43.6°	-44.7°	59.1°	-60.1°	70.1°	-70.2°	Error
Beam 1 at -0.25°	0.0099	0.0283	0.0243	0.0271	0.0281	0.0229	0.0273	-0.1834	0.0354	0.0199	0.0368	0.0303	0.0569	0.0309
Beam 2 at -0.0°	0.0088	0.0273	0.0246	0.0274	0.0269	0.0218	0.0269	-0.1810	0.0357	0.0236	0.0360	0.0335	0.0497	0.0300
Beam 3 at 0.25°	0.0090	0.0268	0.0247	0.0277	0.0275	0.0214	0.0276	-0.1807	0.0330	0.0225	0.0331	0.0326	0.0486	0.0293

The mean error is consolidated into Table 17 to compare the results from Figures 74 to 79. Here it can be seen that the 45° target is the only target with a negative value mean error and the magnitude is much greater than the rest of the targets. Neglecting that target the overall RMS Error is 23.035mm, 22.048mm, and 21.248mm for the base azimuth of -0.25°, 0°, and +0.25°respectively.

Using this information it's possible to create another error model to account for the error between the measured sensor ranges and the calculated range function returns.

Angle Test Correction Factor.

The initial thought regarding the error function was to use a negative cosine function as the estimated error would be expected to be proportional to the angle with zero error at an

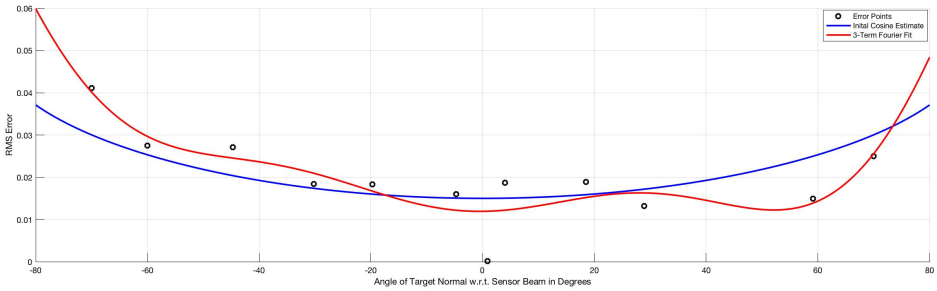


Figure 82. Angle Test Error Model Fit

angle of 0° and potentially infinite error at an angle of ±90°. Equation 57 represents the initial guess and is compared to the mean error in Figure 82, where the cosine function is in blue and the mean error are plotted using black circles

$$E_a = -0.05\sqrt[3]{cos(\theta_N)} + 0.065; \tag{57}$$

where E_a is the error due to the angle θ_N between the measured range by the sensor and the range return from the range function. An additional attempt for a curve fit was attempted using a three-term Fourier equation as shown in Equation 58 which is display in red in Figure 82.

$$E_a = 5.14 \times 10^{-02} - 5.77 \times 10^{-02}cos(0.02615\theta_N) - 5.95 \times 10^{-03}sin(0.02615\theta_N)$$
$$+2.94 \times 10^{-02}cos(0.0523\theta_N) + 6.73 \times 10^{-04}sin(0.0523\theta_N) - 1.12 \times 10^{-02}cos(0.0785\theta_N)$$
$$+1.72 \times 10^{-03}sin(0.0785\theta_N) \tag{58}$$

Using this error model, along with the previous error models and the ones still to be calculated, it's possible to reduce the expected mean error of the range function return. Table 18 shows an improvement of using the correction factor based off of the collective error models of the raw data displayed in Table 17.

Table 18. Corrected Mean Error on Three Center Beams between Range Return and Calculated Target Distances

Estimated Angle	0.0°	5.0°	-5.0°	20.0°	-20.0°	30.0°	-30.0°	45.0°	-45.0°	60.0°	-60.0°	80.0°	-80.0°	RMS
Calculated Angle	0.9°	4.0°	-4.7°	18.5°	-19.7°	28.9°	-30.2°	43.6°	-44.7°	59.1°	-60.1°	70.1°	-70.2°	Error
Beam 1 at -0.25°	0.0013	0.0196	0.0156	0.0158	0.0168	0.0082	0.0124	-0.2038	0.0144	-0.0076	0.0094	-0.0009	0.0261	0.0142
Beam 2 at -0.0°	0.0002	0.0186	0.0158	0.0162	0.0154	0.0073	0.0119	-0.2012	0.0145	-0.0037	0.0084	0.0026	0.0186	0.0127
Beam 3 at 0.25°	0.0004	0.0181	0.0160	0.0166	0.0160	0.0070	0.0125	-0.2008	0.0117	-0.0045	0.0052	0.0020	0.0172	0.0123

Once the error correction is applied to the range estimation function, a comparison between the estimated range returned by the sensor can be calculated with both the original function and the function with error correction applied. Figures 83 through 88 display the

original range returns of the sensor as seen previously but with a blue vertical bar indicating the range estimation function value, and the range estimation function value using error correction in green. It can be seen that the range estimation function results using error correction returns a range value closer to the actual mean of the sensor range returns.

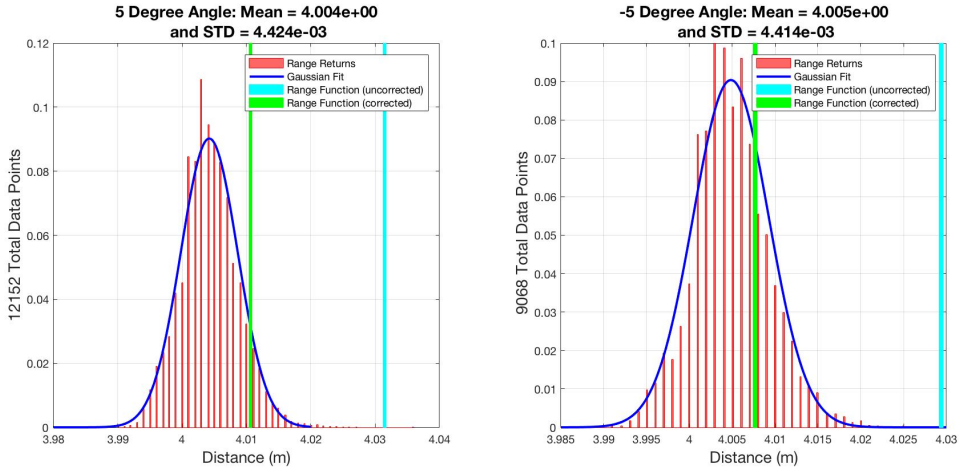


Figure 83. Angle Test Snapshot of Sensor Range Return at ±5°w/ a Mirror

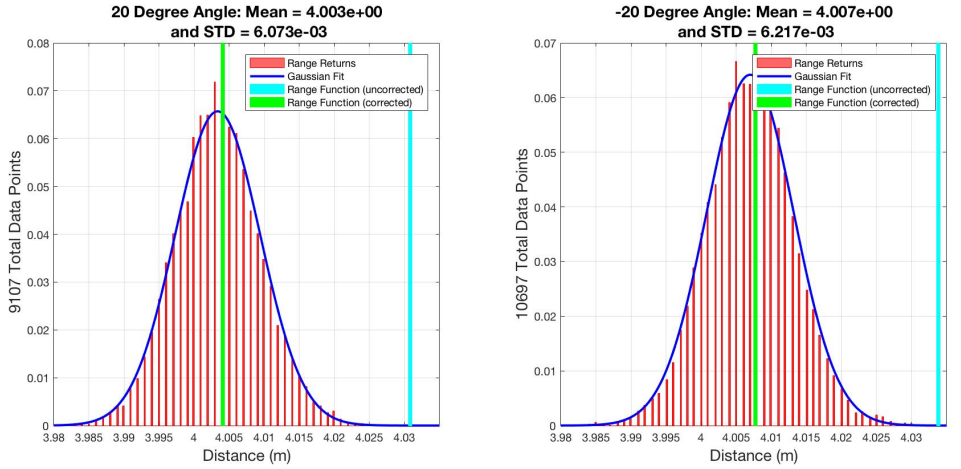


Figure 84. Angle Test Snapshot of Sensor Range Return at ±20°w/ a Mirror

As a summary, Figure 89 shows all of this angle test data in one coherent figure. All 13 test angles are portrayed in a spiral pattern with the ±70° on the top row and the 0° case in the center. The range data across the base azimuth of -0.25°, 0°, and +0.25°are shown in magenta, blue and green respectively. The vertical red bar illustrates the expected function

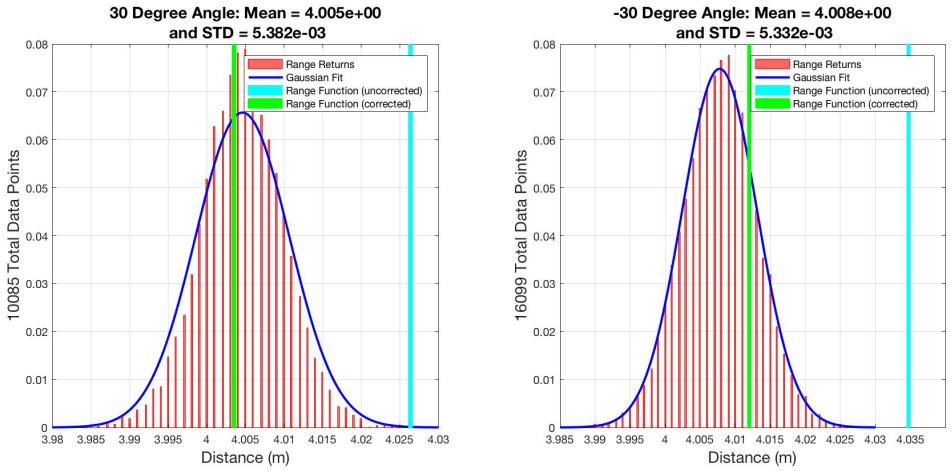


Figure 85. Angle Test Snapshot of Sensor Range Return at ±30°w/ a Mirror

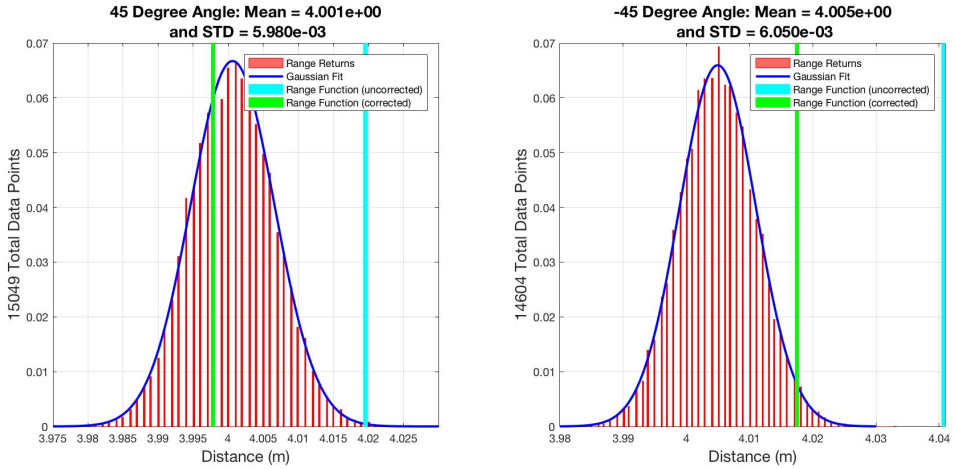


Figure 86. Angle Test Snapshot of Sensor Range Return at ±45°w/ a Mirror

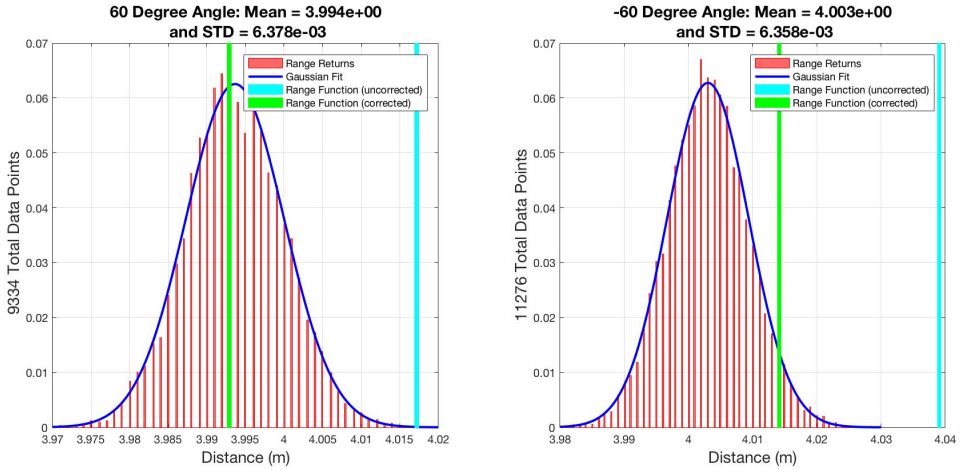


Figure 87. Angle Test Snapshot of Sensor Range Return at ±60°w/ a Mirror

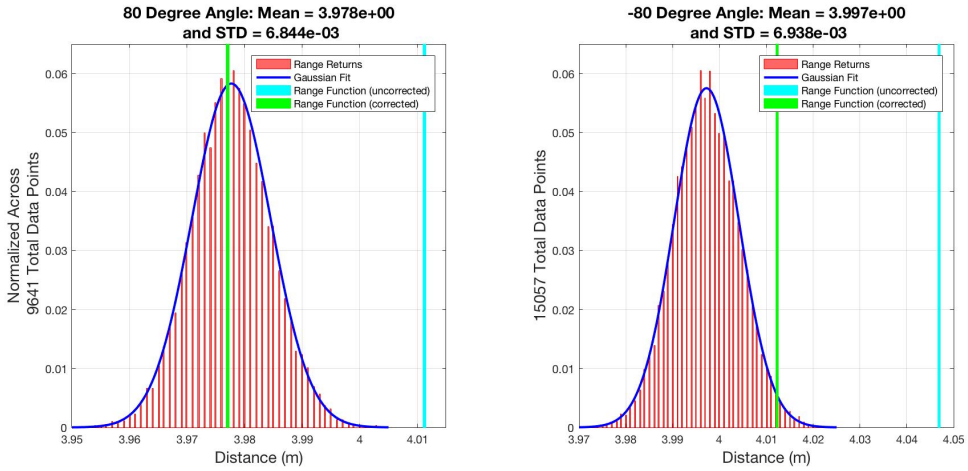


Figure 88. Angle Test Snapshot of Sensor Range Return at $\pm 70°$w/ a Mirror

return. Figure 90 shows a similar display for the angle test data with the error correction applied to the range estimation function. It is harder to see the improvement in this figure but the overall trend across all the angles is more evident.

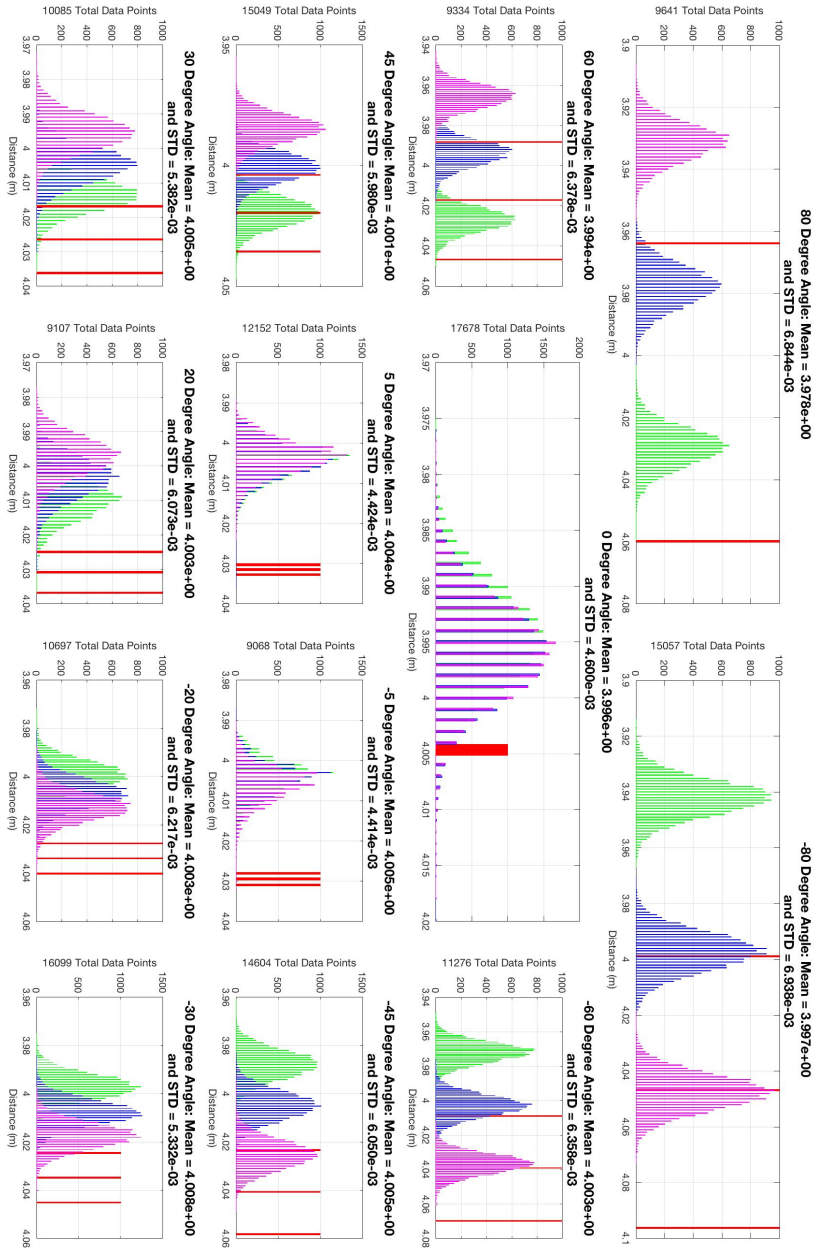


Figure 89. Angle Test Snapshot of Sensor, Measured, and the Raw Range Function Returns

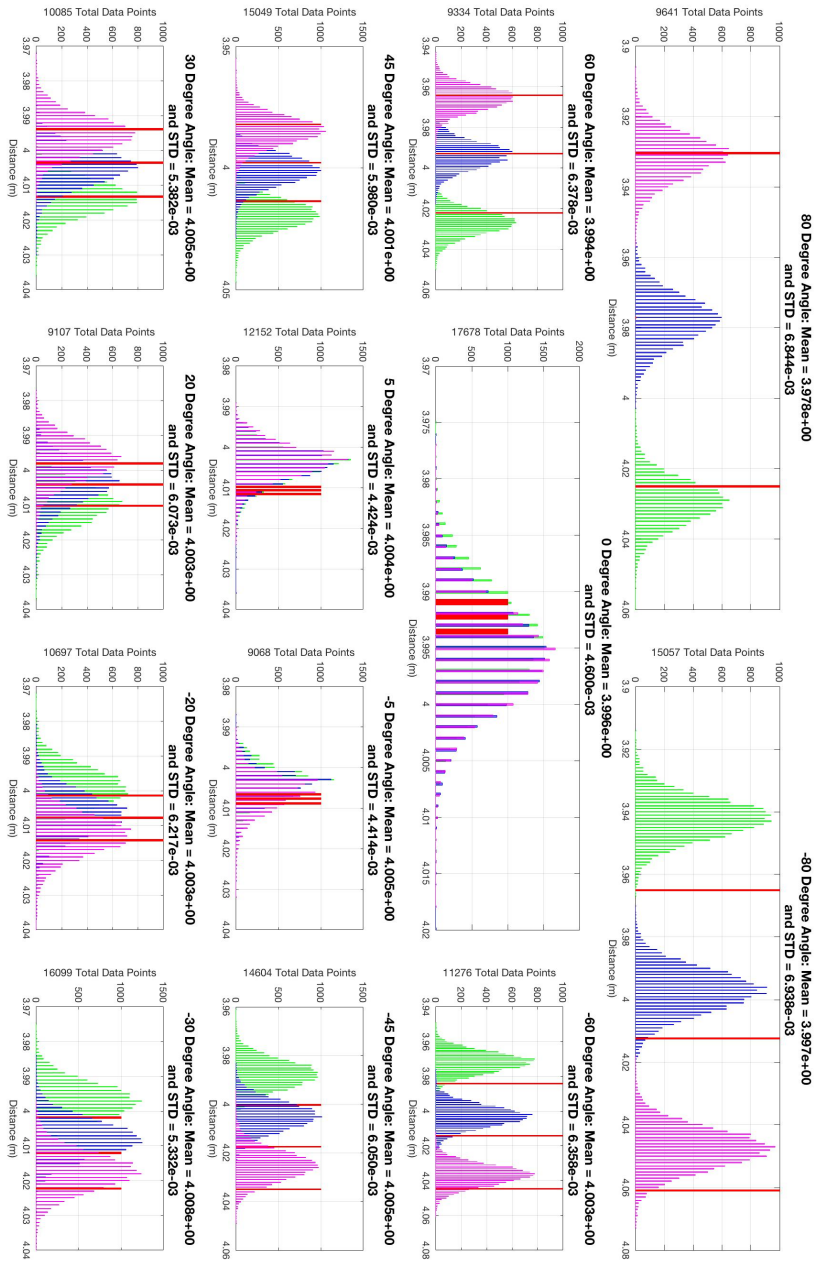


Figure 90. Angle Test Snapshot of Sensor, Measured, and the Corrected Range Function Returns

4.4 Grey Level Test

Once the error models based on the error due to the sensor range and the error due to the angle of the target board are analyzed,the next step is to evaluate how differing shades of gray affect the sensor range return. During initial testing, the target board was painted a flat matte black color, but hindsight has proven this to be an error in judgment. The gray level test was conducted as prescribed by Section 3.1 and the general board positions can be seen in Figures 54 and 55 with a few exceptions. The actual board positions are shown in Table 19 at the indicated distance. This table compares the standard deviation of the sensor range measurements as the distance and target surface color is changed. A range of 1.600m replaced the original 1.500m distance described in Section 3.1 due to some erroneous data returns at 1.500m during the test with a cause that is still unidentified.

Table 19. Standard Deviation of Sensor Returns for Gray Level Test

Distance	Pure White	Light Pine	Antique White	Pewter Grey	Dark Pine	Pencil Grey	Hematite	Jet Black
0.493m	4.39mm	3.63mm	3.96mm	4.18mm	3.90mm	3.85mm	3.93mm	5.37mm
1.011m	4.64mm	4.59mm	4.69mm	4.13mm	4.27mm	4.71mm	4.49mm	6.06mm
1.600m	3.52mm	3.46mm	3.43mm	3.72mm	3.65mm	3.96mm	5.64mm	5.29mm
2.007m	3.71mm	3.91mm	3.55mm	4.04mm	4.05mm	4.18mm	6.05mm	5.00mm
Mean	4.09mm	3.92mm	3.94mm	4.02mm	3.97mm	4.19mm	5.10mm	5.44mm

Table 19 illustrates the STD of the sensor returns for each target collected against for this test scenario. It is interesting to note a couple items that stand out: the first is that the STD at a distance at 1.011m is larger than at most other distances, with exceptions to the near black targets. Additionally, the STD for hematite and jet black are also larger than for the other gray colors. The variance, which is the square root of STD, is plotted as target board color vs. variance for the four different ranges under test. This can be seen in Figure 91, and the overall upward trend is visible here. The eight targets are plotted in order from pure white to jet black vs variance. Next, the actual range measurement at each gray color is evaluated with a summary of the two gray colors that represent the widest variation between values in Figure 92. Once the colors of pure white and jet black were

identified, an additional test at 1.50m and 3.0m was conducted to fill in the data gap.

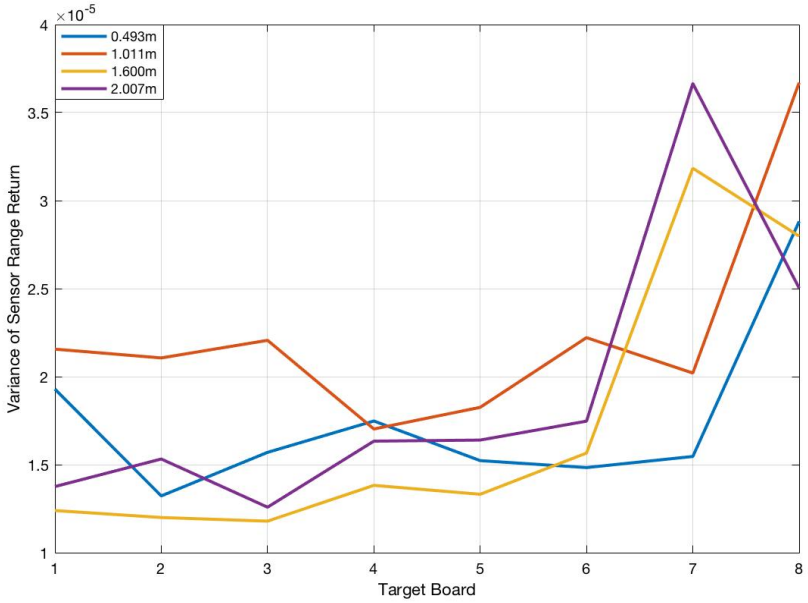


Figure 91. Gray Level Target Range Variance

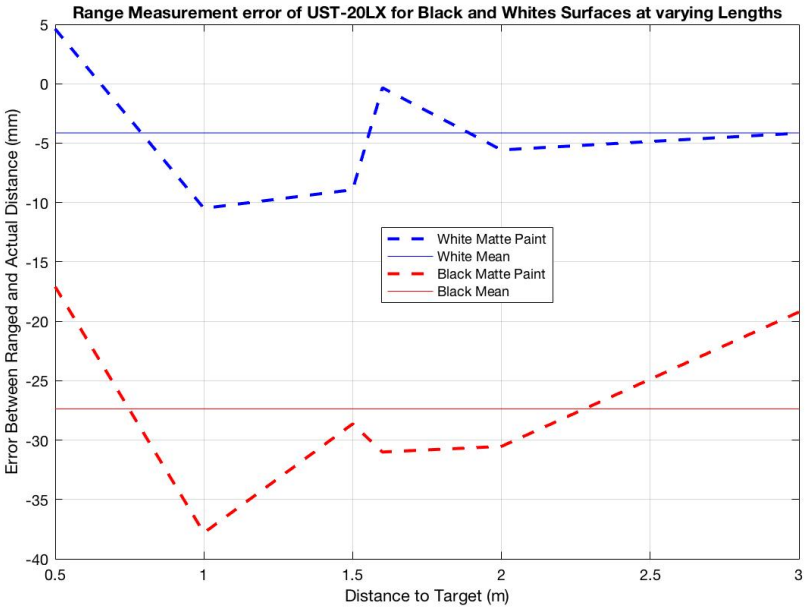


Figure 92. Gray Level Error Summary

For a more complete picture of the range return due to the gray levels of the colors

tested at each range please reference Figures 93 through 96. The thick vertical line in each figure indicates the measured range of the target via a measuring tape with a tolerance of $\pm 3.2mm$ as in the range test. Upon taking a close look at the data displayed in the figures, it can be seen that the relationship between the gray level is consistent in the fact that only the very dark shades of gray cause a noticeable effect to the range return.

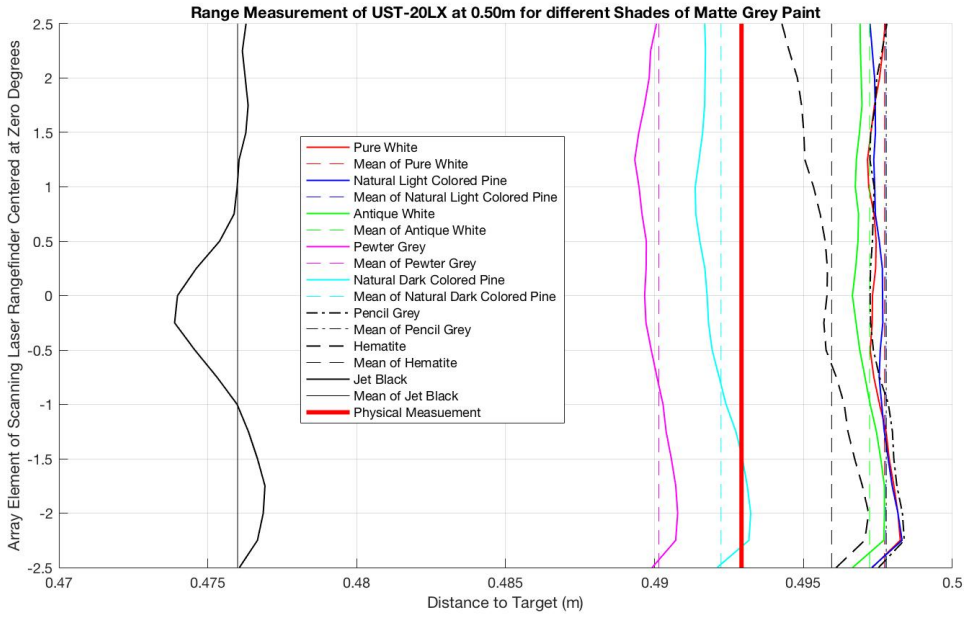


Figure 93. Gray Level Results at 0.5m

The colors of pure white, antique white, and the light colored natural pine all stay within the same general offset category, whereas the pewter gray and the dark colored natural pine cluster together at a larger offset from the previous group. Pencil gray, and hematite seem to proportionally increase in offset until the largest offset cause by jet black is achieved. Pencil gray is interesting, because the offset due to this color does not seem to be consistent. At 0.50m it shares a similar offset to the group close to pure white, at 1.60m it causes a range error close to those caused by the second group near pewter gray, but at 2.0m it pushes its range error close to hematite and jet black. Keeping these results in mind, it's possible to gain some insight by looking back at Table 19. There seems to be another underlying

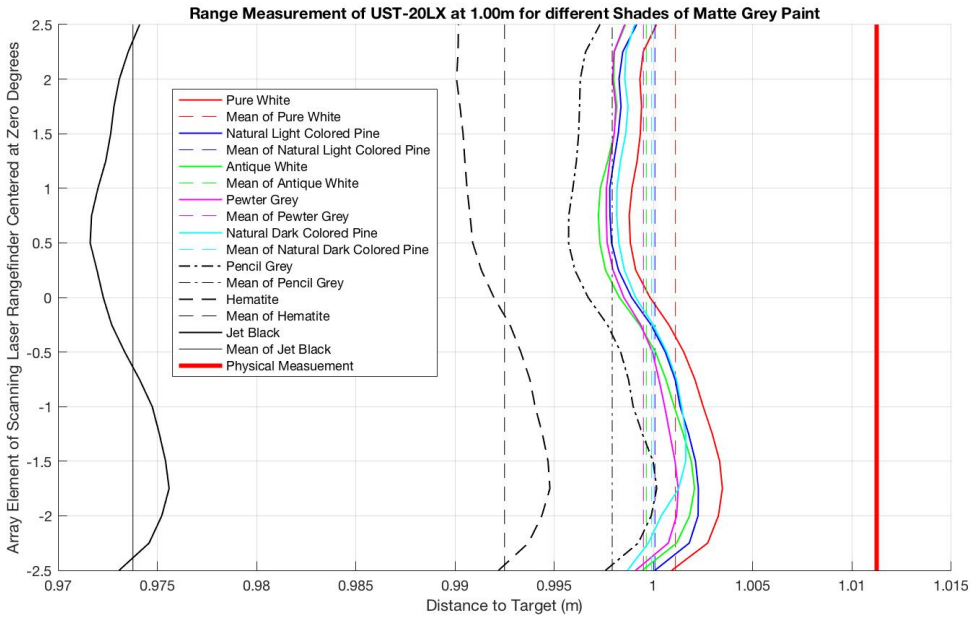


Figure 94. Gray Level Results at 1.0m

relationship between the STD of the data with the range and the color. At larger distances the variability appears to be smaller for pure white, which is the brightest color of the group, while having a much larger STD at the 0.50m distance. It seems that an unintended reflectivity component may be causing this, even though flat matte paints were used in each case. At closer distances where intensity returns are much higher, the possibility of a higher speckle components in the range return is present. Large amount of speckle would cause the STD to increase. When looking at pure white and antique white at a 2.0m distance it shows a 28.6% and 19.9% decrease in STD, respectively.

In summary of the gray level test we can again look at Figures 93 through 96 to see that white appears to offer the lowest overall sensor range error. This choice comes at a cost due to the larger STD that it comes with. Statistically the best choice appears to be the light pine, but the benefit of smaller sensor range error and the ability to cover the target surface with paint overcomes the benefit of an 8%smaller STD.

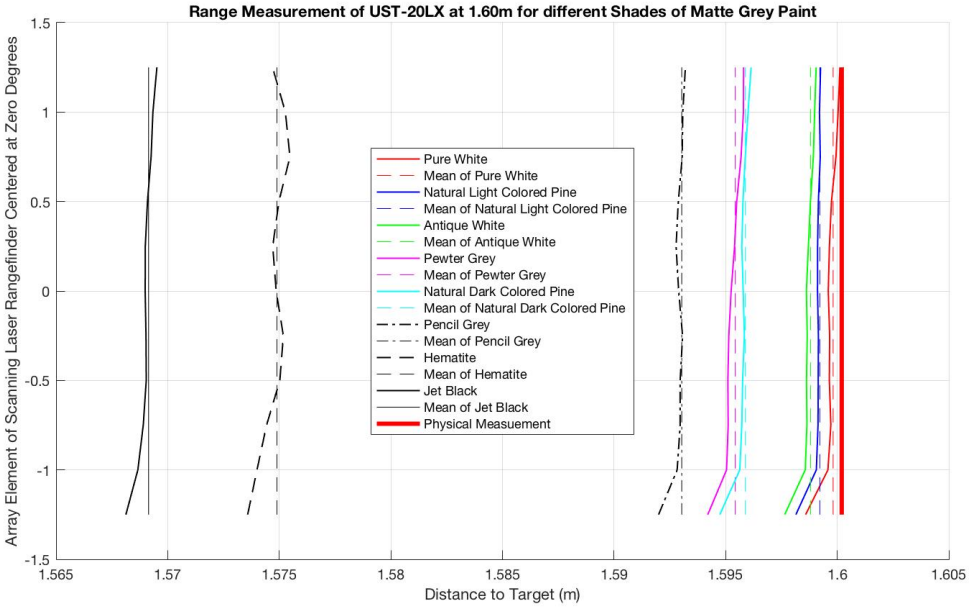


Figure 95. Gray Level Results at 1.6m

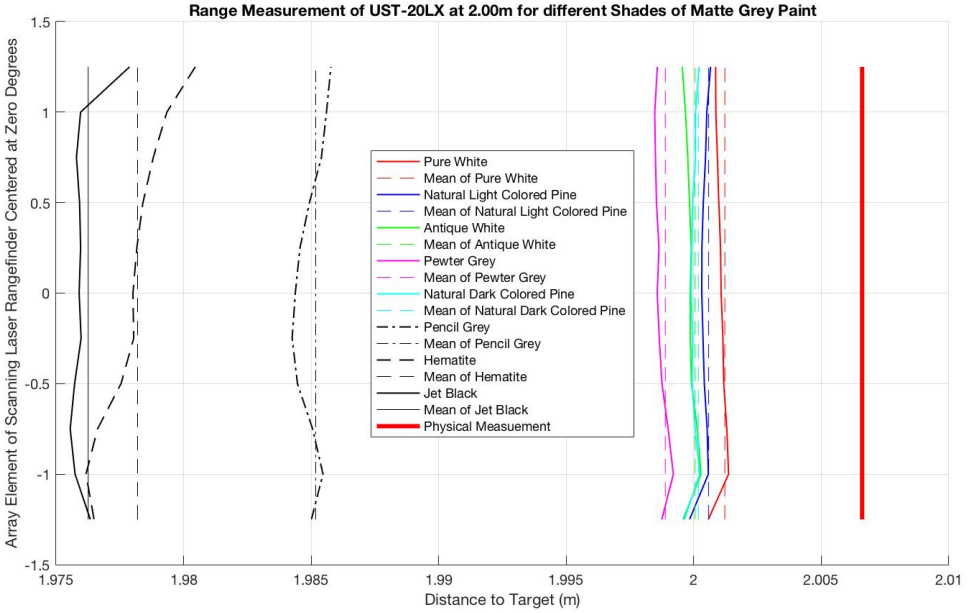


Figure 96. Gray Level Results at 2m

Gray Level Correction Factor.

Once the preferred target board is identified, the error model for that choice can be calculated. Figure 97 shows the mean error for white in blue and the mean error for black in red. They are both plotted to illustrate the similarity in range error to each other. A four-term Fourier series in Equation 59 is used to create the error model from using a pure white target board in light blue in Figure 97. A separate four-term Fourier series in Equation 60 is used to create the error model from using a jet black target board in magenta.

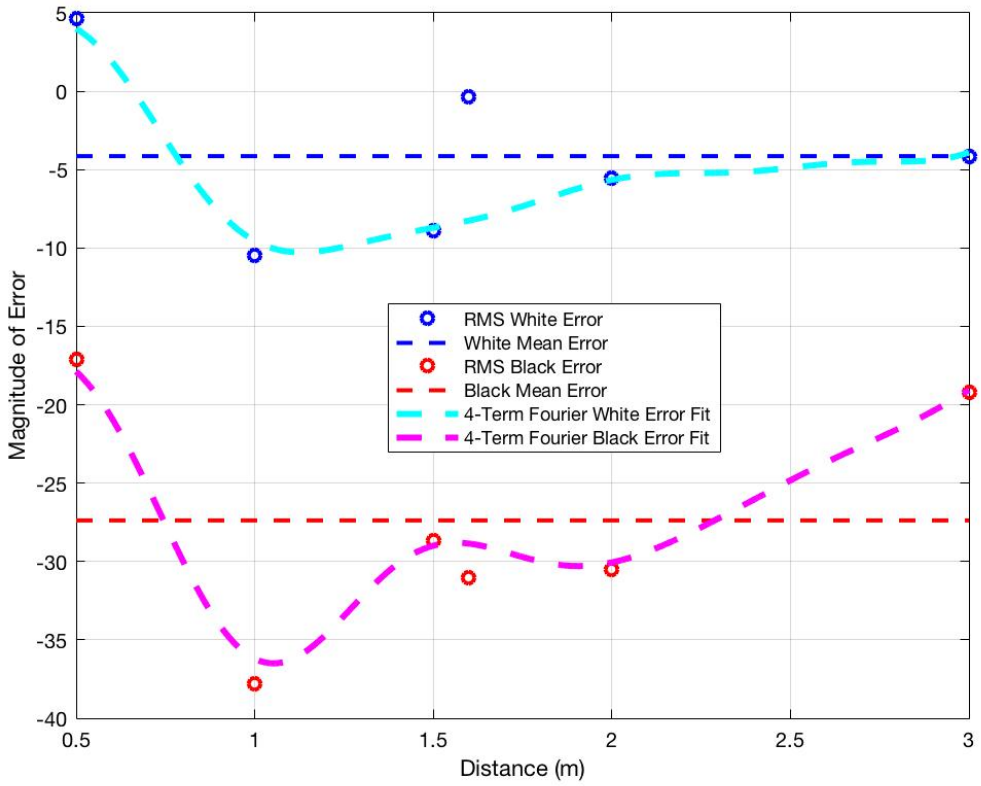


Figure 97. Error Model Fit Against Error Due to Black or White Matte Targets

Equations 59 and 60 represent the error models in Figure 97 which describe the sensor

range error from the measured ranges between 0.5m and 3.0m.

$$E_w = -4.592 + 4.287cos(2.033r) + 1.166sin(2.033r) - 0.5697cos(4.066r) + 3.295sin(4.066r) - 1.447cos(6.099r) + 0.5061sin(6.099r) - 0.5075cos(8.132r) - 0.4295sin(8.132r) \quad (59)$$

and

$$E_b = -26.6 + 6.692cos(2.183r) + 0.9485sin(2.183r) + 1.188cos(4.366r) + 3.618sin(4.366r) - 2.237cos(6.549r) - 0.3769sin(6.549r) + 0.297cos(8.732r) - 0.3783sin(8.732r) \quad (60)$$

where r is the range to the target and E_w and E_b are the estimated error due to a matte white or matte black target respectively.

Angular Resolution Test.

As part of the characterization of the Hokuyo UST-20LX scanning laser rangefinder, a test to determine the angular resolution was conducted as described in Section 3.1. A target distance of 10.0m was used in order to allow for a good resolution on the test results using uncorrected raw range measurements. A summary of the results can be seen in Figure 98, which shows the indicated separation distance between the targets board represented by gray bar plots, and the range return in the colored line plots. The summary shows the results using the mean 10,000 range measurements at each corresponding gap distance, with the beam number on the x-axis and the measured range on the y-axis. In an ideal world, the depth of the gap shown would indicate the prescribed testing distance of 2.0cm for and gap size greater than 0°. This is not the case.

Figures 99 through 103 show the progression of the gap size as the separation grows

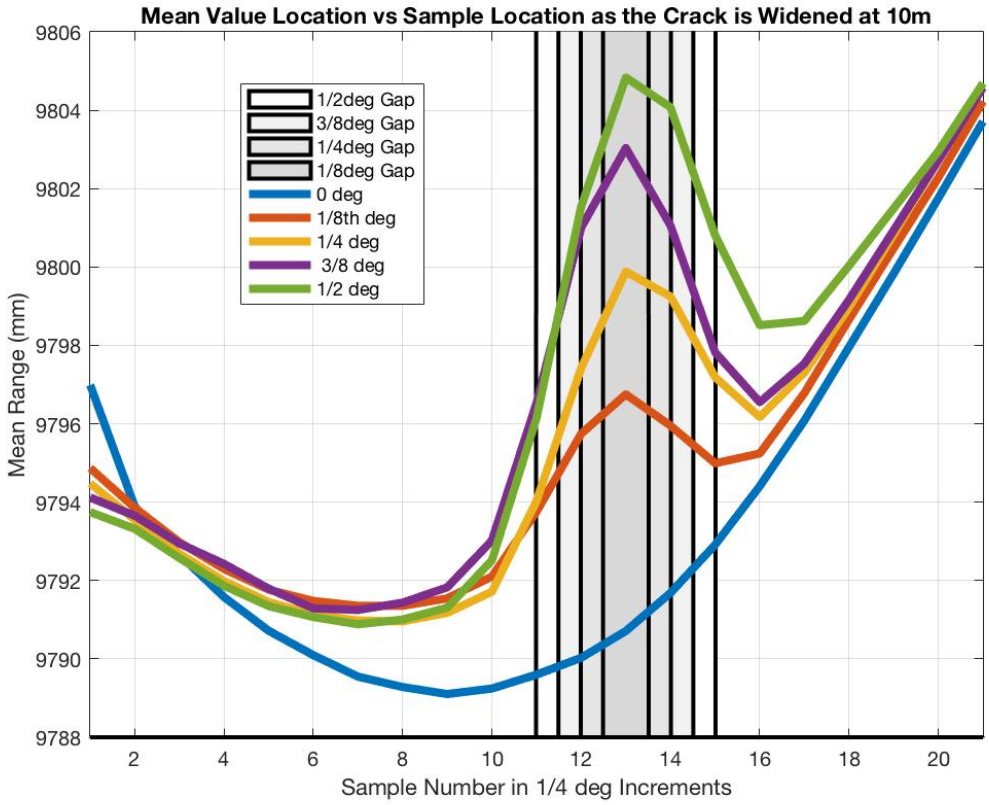


Figure 98. Mean Range Estimate for angular resolution test using 10,000 samples

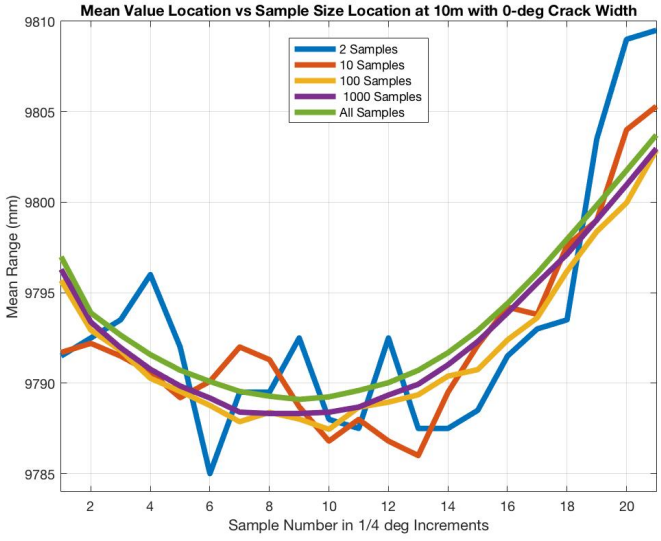


Figure 99. Mean Range Estimate for 0°of target separation

Table 20. Gap Distance Equivalent to Angular Distance from Zero to 1/2 Degrees in 1/8 Degree Increments at 10m Target Distance.

Gap Size (deg)	1/8°	1/4°	3/8°	1/2°
Gap Size (mm)	21mm	42mm	64mm	85mm

from 0° to 1/2°. In the case with no separation between the two target boards, the general shape is recognizable in each sample size. The 2-sample mean in blue shows the largest variation, with the 10-sample mean vary similar in red. For sample sizes of 100 and larger, the results show a much more consistent result, in that the target form can be readily inferred from the measurements.

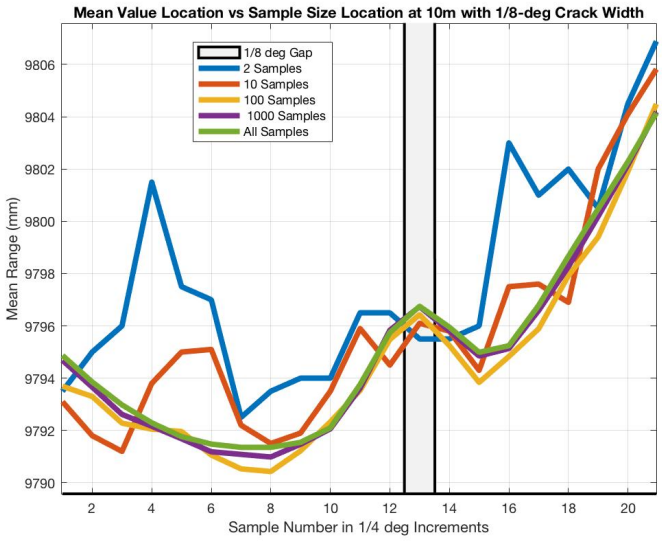


Figure 100. Mean Range Estimate for 1/8°of target separation

Figure 100 is the first figure showing the smallest separation value, of 2.1cm. At this distance, the entire spot size of the beam completely covers the gap between the two target boards. The indication of the a gap between the two target at the 13^{th} element begins to take shape, but again, this is not readily inferred until a 100 sample mean is taken. The following figure showing a 1/4° separation, Figure 101, looks very similar to the results in Figure 100.

Figures 102 and 103 show a target separation of 3/8°, and 1/2°, respectively. Looking

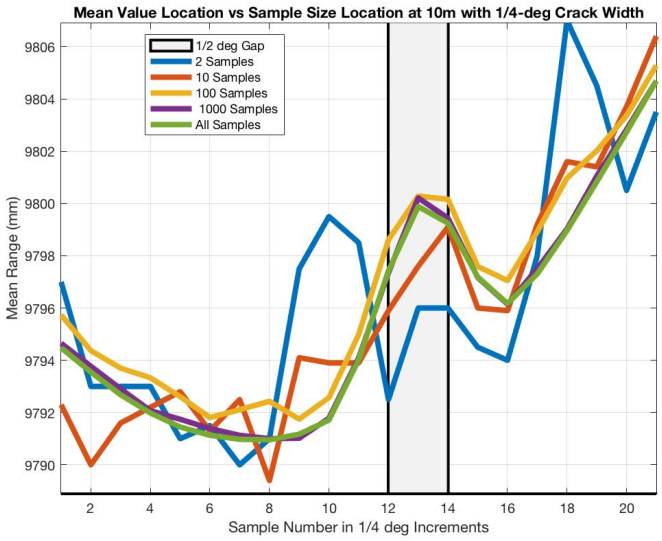


Figure 101. Mean Range Estimate for 1/4°of target separation

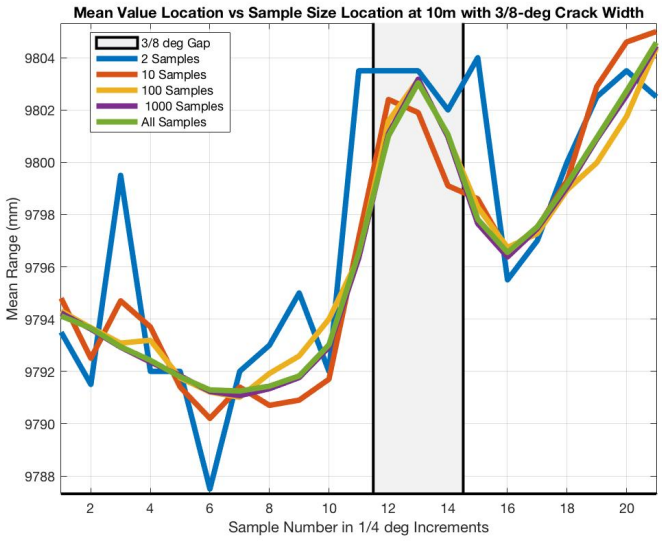


Figure 102. Mean Range Estimate for 3/8°of target separation

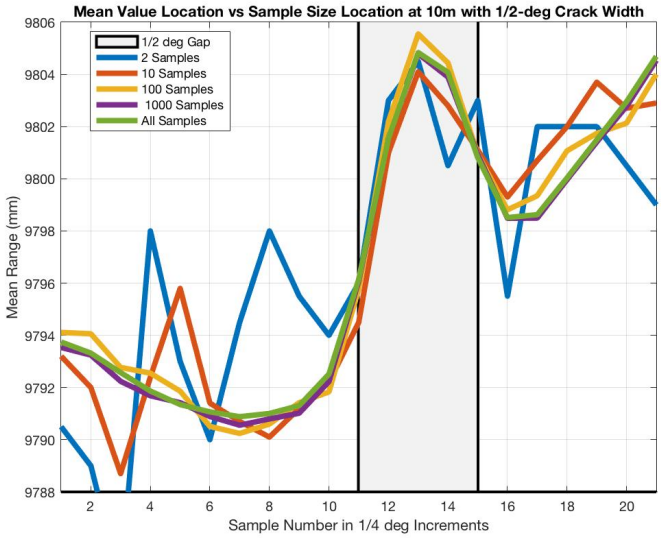


Figure 103. Mean Range Estimate for 1/2°of target separation

at the 2-sample mean in blue, it's possible to see the occurrence of mixed pixels identified by [43]. The range measurements by the sensor do not return an accurate range due to the depth of the gap in between the targets, but a large indication of a gap in a potential surface does exist. Figure 103 shows even more indication, with a stronger correlation between each sample size. With this in mind, the results seem to indicate that an object with less than 1/4° of angular width relative to the sensor may not be indicated in the returned LiDAR range measurement but may in fact get lost in the noise due to "mixed pixels".

The standard deviation of the measured ranges for the case with zero target separation in Figure 104, and the case with 85mm of target separation in Figure 105. When comparing the two extreme case represented by these figures, it is noticeable that there does not seem to be an effect of the standard deviation of the results, even at the anticipated location of mixed pixels mentioned previously. It therefore seems unlikely to use the standard deviation for any kind of edge detection in the measured range results in of itself.

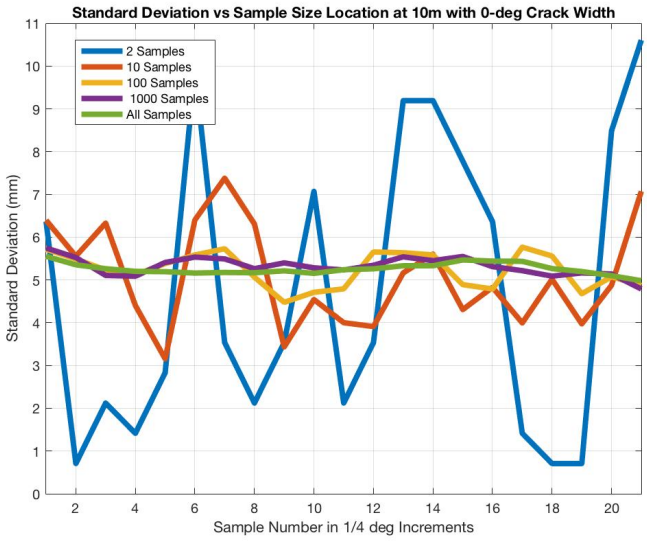


Figure 104. Standard Deviation of Results with 3/8°of target separation

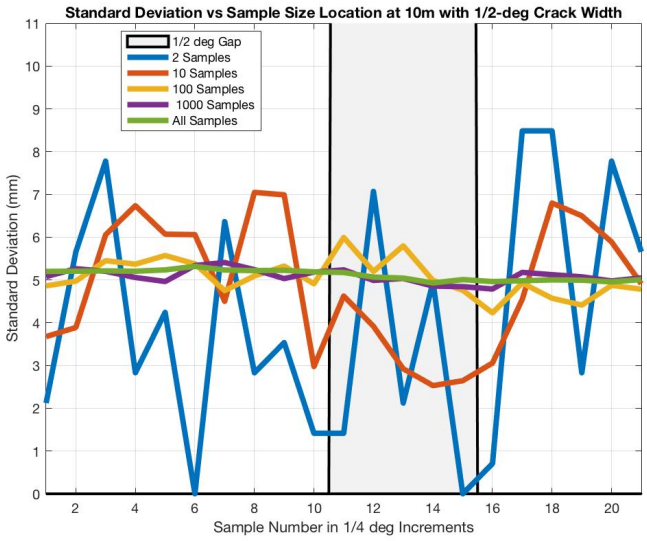


Figure 105. Standard Deviation of Results with 1/2°of target separation

4.5 Intensity-Range Profiles

As discussed in Section 2.2, it's fairly common for laser rangefinders to use a threshold detector based on the returned intensity from a target, and as such it would be incomplete to not understand how the returned intensity is affected by the target distance or by a mirrored surface in the sensor propagation path. Figures 106 and 107 illustrate the test scenario to take the data for the intensity vs. range profile.

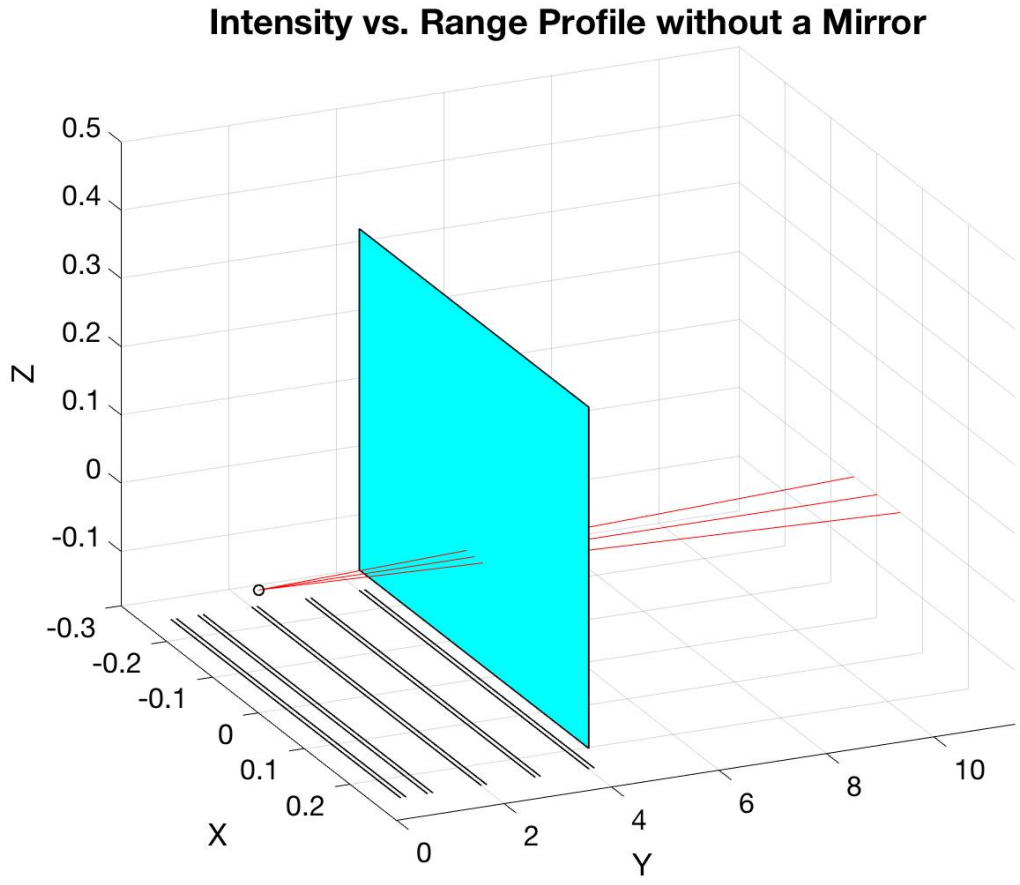


Figure 106. Intensity Profile Test Scene

Figure 106 displays the variety of ranges that were tested against in the initial test design. From an overhead perspective the target boards, in blue in Figure 106, appear to be black as it was attempted to place the target board perpendicular to the beam sensor beam

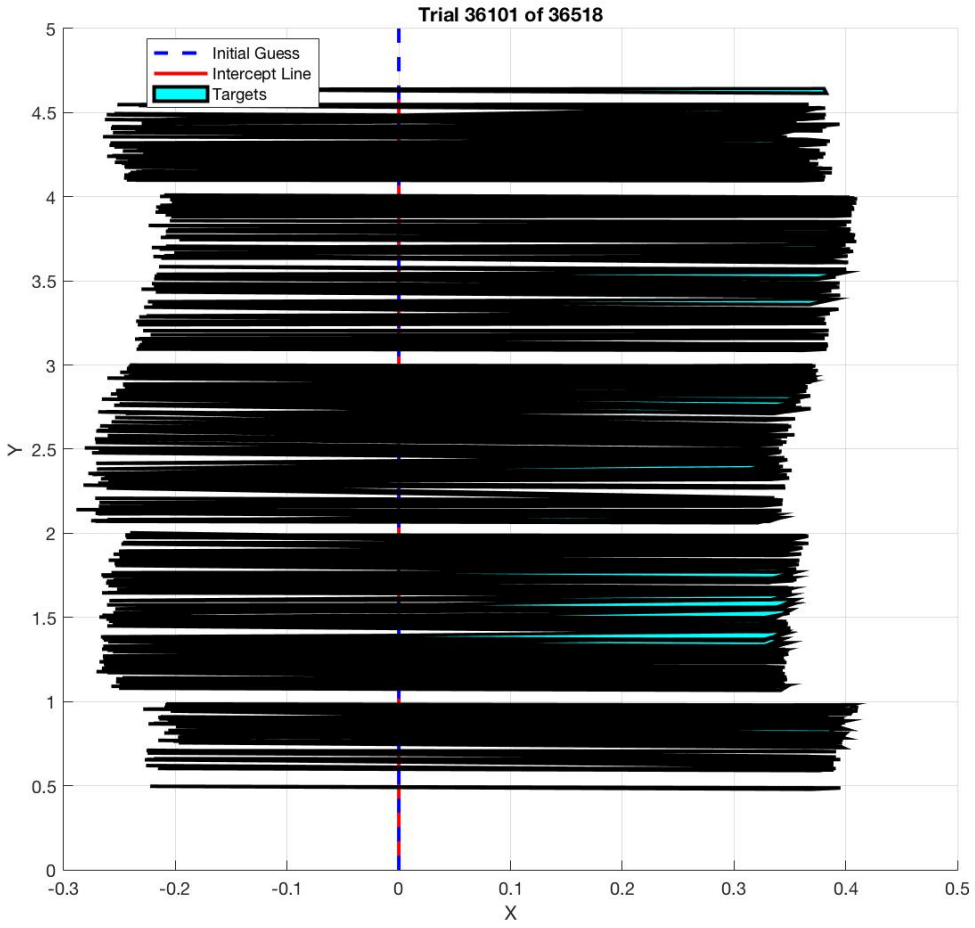


Figure 107. Illustration of Target Positions for Intensity Profile Test Relative to Laser Source

path. The target board was positioned along the y-axis up to a range just shy of 6m. At a distance of 0.5m, 1.0m, 2.0m, 3.0m, and 4.0m the target was removed for approximately 10cm to induce gaps in the data. These gaps allow for a ”sanity check” to ensure the proper ranges are returned, or not returned at the location of the gaps. Figure 107 shows a representation of the target position from an overhead perspective and there were too many positions to fully display and only 1 out of 100 are displayed. Even with 1% of the target positions displayed it's possible to see the 10cm gaps at 0.5m, 1.0m, 2.0m, 3.0m, and 4.0m.

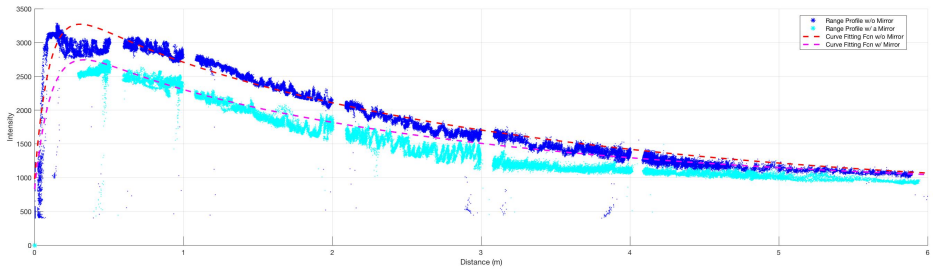


Figure 108. Intensity vs. Sensor Range Return Profiles

The full intensity profile can be seen in Figure 108 in dark blue, with the secondary intensity profile shown in light blue displaying the data from the subsequent test with a mirror in the propagation path. The dashed red lines represent the curve fit solutions in Equations 61 and 62. It was expected that the intensity profile to be proportional to 1/(range^3), but one can see that the plot portrayed does not entirely replicate that assumption. As it turns out, the hardware in the Hokuyo sensor limits the intensity return when reaching a specific threshold. This threshold is not public knowledge but may be near an intensity value of 3300 according to the data. Past this threshold (at a shorter distance) the intensity values drop quickly and in this region the range values may be less accurate according to a Hokuyo technician[96].

The test scene for the intensity profile data collect with a mirror can be seen in Figure 109. Due to the test environment, positional data could not be reliably obtained after 5m

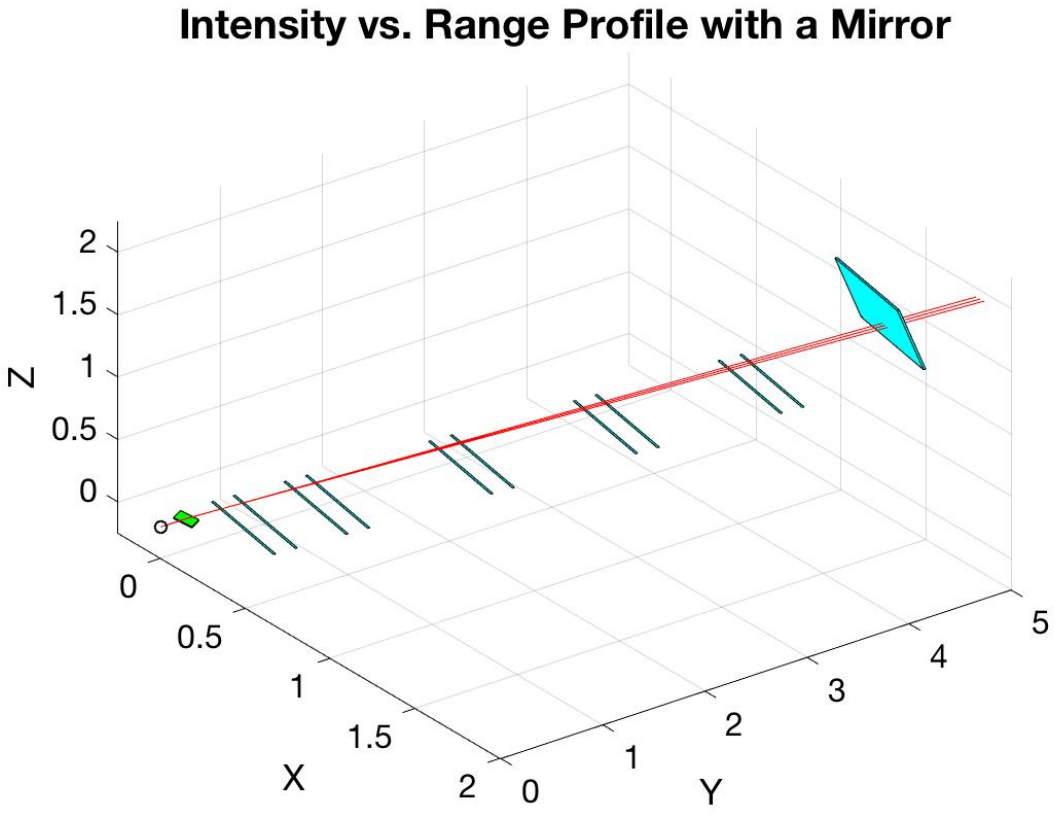


Figure 109. Intensity Profile Test Scene w/ a Mirror

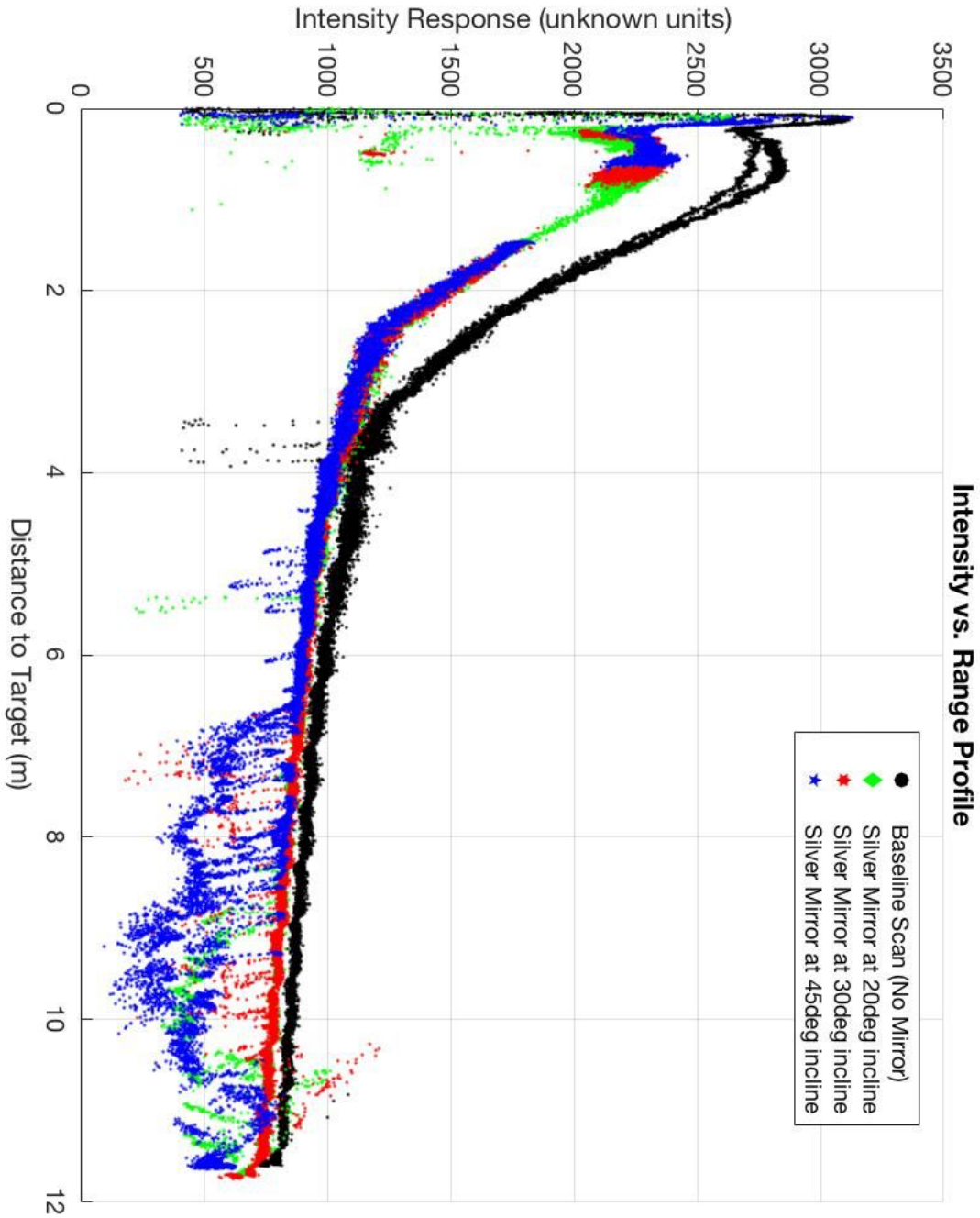


Figure 110. Continuous Intensity vs. Sensor Range Return Profile

but another test was conducted without any data gaps to help illustrate the overall trend in the intensity profiles. Figure 110 displays this data collect and a total distance of 11.5m was obtained. This trend appears to exponentially decay towards an intensity value of 500 with a fairly linear region between 1.5m and 2.5m. The initial data collect is in black and has no obstructions to the sensor beam. The test was also conducted with a silver first surface mirror at three different deflection angles.

Angles of 20°, 30°, and 45°were evaluated in Figure 110 and correspondingly displayed in green, red, and blue. If one looks carefully, it's possible to notice that the scatter plots for 30°and 45° appear to have a section missing from them near a target range of 1m-1.5m. There is no actual data missing in these plots, but this is an example of what happens when the entire field of view of the sensor is not covered by the reflecting surface and edge effects occur. The data collects at those "missing" points actually return a range that is somewhere in between the true range and 0m. This is noticeable as a cluster of point near 0.5m and and intensity of 1250. These points are not real measurements but are a good example of the anomalous returns discussed in previous sections.

$$I_{nm} = 3417e^{-0.233r} - 2935e^{-10r} + 3000log\left(\frac{r}{5}+1\right)e^{-r} - 1250e^{\frac{r}{r}} + 1000e^{-2r} + 750 \tag{61}$$

$$I_{m} = 3717e^{-0.233r} - 2535e^{-8.9r} + 3000log\left(\frac{r}{5}+1\right)e^{-r} - 1250e^{\frac{r}{r}} + 1000e^{-2r} + 900 \tag{62}$$

where I_{nm} and I_m are the expected intensity returns for a given range, r, with and without a mirror in the path.

4.6 Range Function Correction Factor

Once the sensor has been characterized by the range test, beam divergence test, angle test, gray level test, and the intensity profiles an overall correction factor can be calculated, and applied to the range function. Modifying Equation 33 from Section 3.3 to include the calculated offsets in Equations 54, 58, and 59 a corrected range equation can be expressed as

$$r(\hat{x}) = r(\tilde{x}) - E_{r_{nm}} - E_w - E_a \tag{63}$$

where $r(\tilde{x})$ is the expected range with a given target location, $E_{r_{nm}}$ is the error of the range function from the measured range, E_w is the error of the range function due to a flat matte pure white painted board, and E_a is the error due to the angle of the normal vector of the target board with respect to the sensor beam location.

The improvement in the accuracy to estimate the sensor range return by the range function can be seen in the second rows of Figures 56, 59, and comparing Figure 89 with the corrected ranges in Figure 90 will also show the improvement in the sensor range estimation to an overall range error less than 2cm at a target distance of 4.0m.

The error model was completed using a single collection of data for each component of the error model. Although the error model, as applied to the range function estimate, is shown to improve the range function estimates when compared to the actual ranges, this test was not repeated. Repeating this test many times would be beneficial as a way to eliminate measurement error bias due to the specific scenario and environmental factors.

4.7 Recursive Least Squares Function

After finalizing the Range Function error correction adjustment in the previous section, the next step is to run the trials to collect data to verify the RLS algorithm. First a test scene was needed to verify that the algorithm is able to converge. Figures 111 and 112

show the initial test using an unknown mirror angle of deflection and an arbitrary distance in the mirror location and the target location. The targets are again in light blue, with the initial estimated RLS solution in red while the calculated RLS solution is in dark blue. The estimated solution is assuming that the sensor beam increment of interest is landing on the center of mass for the group of targets. This gives a ”close enough” initial guess for the the RLS algorithm to be able able to converge. This initial test used eight target board orientations near a distance of 2-2.5m from the sensor origin with a position STD displayed in Table 21 and the calculated azimuth, elevation, and mirror distance at 20.3°, 25.1°, and 0.1996m respectively. The sensor beam at -0.25° did not land on the target board and was intentionally conducted as such to verify the algorithm could handle different number of targets.

Table 21. Standard Deviation of Target Boards Positions for Initial Test Trials

	Target 1	Target 2	Target 3	Target 4	Target 5	Target 6	Target 7	Target 8	Target 9
-0.25°	1.49mm	0.46mm	1.25mm	0.55mm	1.43mm	0.42mm	0.26mm	4.60mm	
0.00°	1.49mm	0.46mm	1.25mm	0.55mm	1.43mm	9.49mm	1.04mm	0.26mm	5.38mm
+0.25°	1.49mm	0.46mm	1.25mm	0.55mm	1.43mm	6.72mm	1.02mm	0.26mm	5.05mm

Once convergence is tested on a single trial with the basic recursive least square algorithm, the repeatably test is conducted using ten trials across five variants of the RLS algorithm. The first two are an RLS algorithm that only estimates azimuth, and elevation, and an RLS algorithm that estimates the mirror distance along with azimuth and elevation. The next three variants are Extended Recursive Least Squares (ERLS), Weighted Recursive Least Squares (WRLS), and Extended/Weighted Recursive Least Squares (EWRLS). The results can be seen in Table 23 and Figures 113 through 115.

The bold entries in Table 22 represent the targets that were artificially manipulated to increase the standard deviation of the target to induce extra noise in hopes to elicit responses between the different algorithms to show which one can handle noise more effectively than the others. For this data collect, the noise may not have been large enough to elicit the

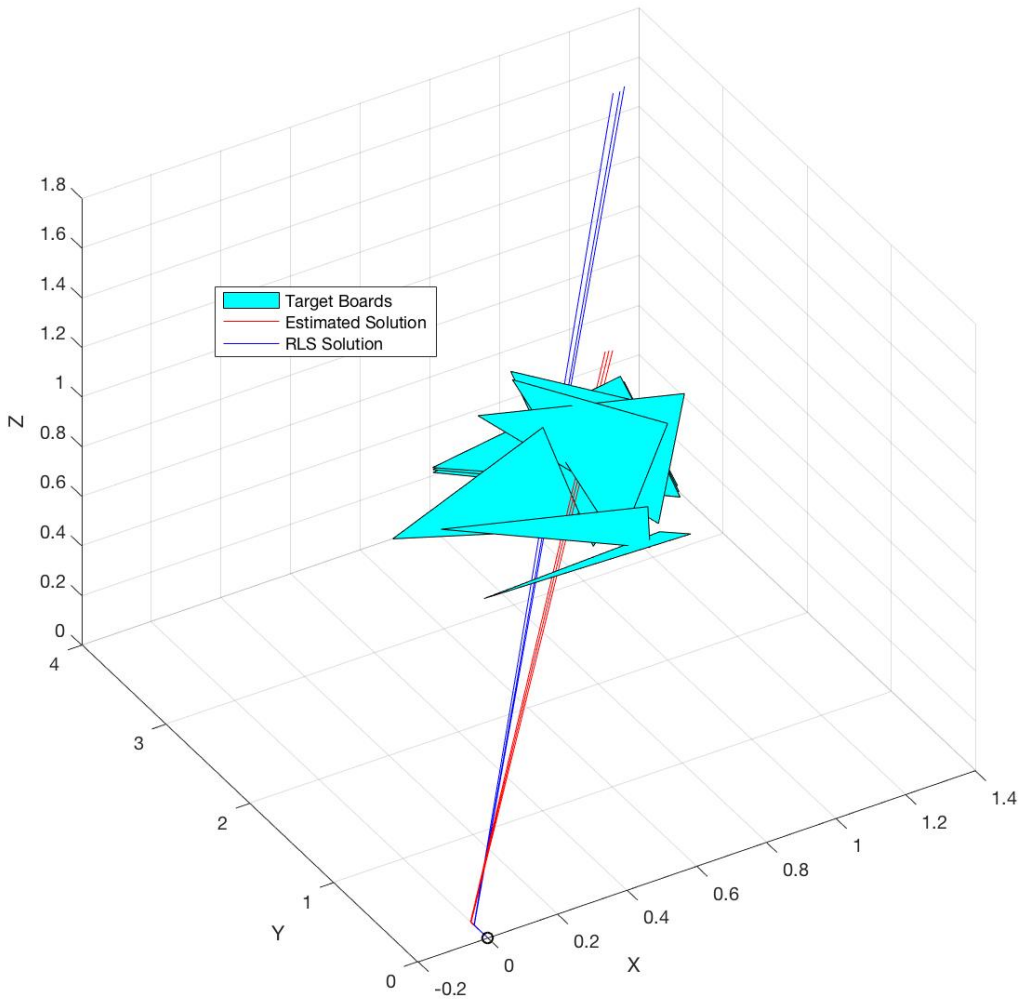


Figure 111. Illustration of Target Positions for RLS Test Relative to Laser Source

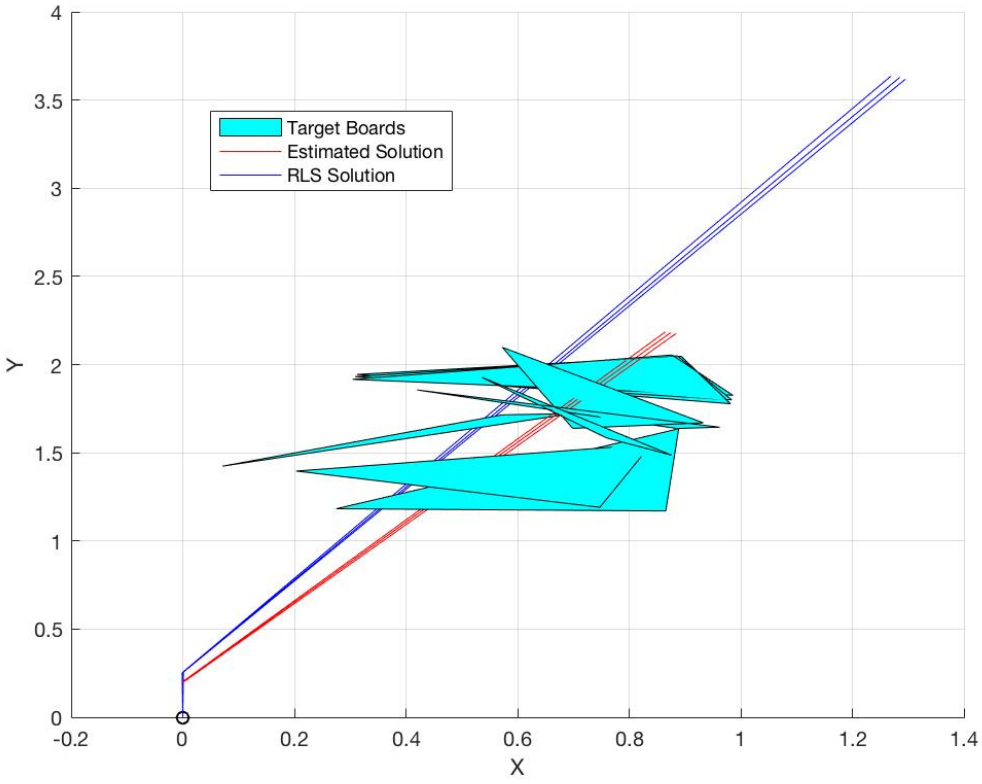


Figure 112. Overhead Illustration of Target Positions for RLS Test Relative to Laser Source

Table 22. Standard Deviation (in mm) of Sensor Returns for the 10 RLS Test Trials

	Target 1	Target 2	Target 3	Target 4	Target 5	Target 6	Target 7
Trial 1	6.12	5.432	5.45	6.74	6.46	8.69	
Trial 2	5.59	5.83	6.18	6.62	6.02	7.55	5.95
Trial 3	6.10	5.20	5.08	4.09	5.37	5.34	8.27
Trial 4	6.29	8.68	5.63	5.09	5.45	***15.65***	
Trial 5	5.11	5.83	6.62	4.29	4.94		
Trial 6	6.11	5.47	5.61	4.77	4.31		
Trial 7	4.68	4.97	6.08	6.84	4.88	***14.66***	
Trial 8	9.69	5.04	6.83	***12.55***	5.25	4.78	***11.13***
Trial 9	4.79	8.87	5.46	9.81	***15.15***		
Trial 10	7.18	***14.40***	8.70	5.74	5.68	***14.67***	

intended effects. The standard deviation in Table 22 is also portrayed differently than in the previous section. Previously the standard deviation of the target location is displayed to show the stability of the target itself but here it is desirable that the variability of the sensor return is published. These STD values are proportional to the weights used in WRLS as W in Equation 41 which uses 1/variance as the individual weights. The STD of the sensor return represents the STD of the target board, any noise caused by the environment such as a mirror, and any noise caused by the electronics that may not be easily identifiable.

Table 23. Mean and Standard Deviation of Azimuth, Elevation, and Mirror Distance Across 10 Trials with 5 RLS Methods

	RLS d-constant		RLS		ERLS		WRLS		EWRLS	
	Mean	STD	Mean	STD	Mean	STD	Mean	STD	Mean	STD
Azimuth (UC)	41.7158	0.4132	40.6631	0.3217	40.6631	0.3217	40.6541	0.3556	40.6537	0.3559
Azimuth (C)	41.4217	0.1853	41.1913	0.3356	41.1913	0.3356	41.1780	0.3664	41.1777	0.3665
Elevation (UC)	45.2649	0.3726	44.8403	0.0927	44.8403	0.0927	44.8631	0.1203	44.8631	0.1203
Elevation (C)	45.3590	0.1822	45.2458	0.1511	45.2458	0.1511	45.2702	0.1716	45.2701	0.1716
Distance (UC)	7.3000	0	4.8633	0.4291	4.8633	0.4291	4.8674	0.4529	4.8674	0.4529
Distance (C)	7.3000	0	6.8255	0.5075	6.8255	0.5075	6.8300	0.5753	6.8298	0.5754

Table 23 shows the results of the 60 total tests conducted to test the repeatability of the RLS algorithms. The cumulative center of mass for the target positions in each trial was used as the initial conditions for the RLS algorithms at each of the ten trials. This means that the initial condition was not constant between trials but was constant within the trial.

The results of Table 23 are visualized in Figures 113, 114, and 115. Here the converged values of azimuth using the corrected range function is in thick blue, the elevation using the corrected range function is in thick magenta, and the values using the uncorrected range function are in the corresponding thin blue and magenta lines for reference.

Another important component in the comparison of RLS algorithms is the speed of convergence to an estimated value, which is displayed in Table 24 for each trial. This convergence rate may depend entirely on target board orientations and the residual errors due to the range function itself, but does not relate to the accuracy of the estimate to the true values. In each trail shown in Table 24 the RLS algorithm with two degrees of freedom (RLS-2D) converges within the iteration limit of 500 iterations, even under trial six in which none of the RLS other algorithms with three degrees of freedom (DOF) converge. The RLS-2D algorithm seems to be reliable, possible due to limiting the algorithm to only two degrees of freedom forces it to iterate closer to the best guess whereas a third parameter allows the other RLS algorithms to adjust the almost unobservable mirror distance, d, to get lower residuals. In trial 9, where all the RLS algorithm seem to converge strongly, the WRLS and EWRLS versions of RLS converge slightly faster. This implies that the W in WRLS and EWRLS allows for faster convergence with targets with a little bit of noise or that the targets were in almost an ideal orientation collectively. But this conclusion can not be made based on one data set. This information causes more questions to be asked such as ”What level of noise on the target is required in order to show a significant difference in RLS algorithm performance?” or ”What is the ideal target orientation to ensure the most accurate convergence?” Questions such as these are valid but are not within the scope of this research and will not be investigated further.

Table 25 is presented in addition to the previous information and displays the Mean sensor range at each target for each trial, the residual from the EWRLS algorithm at each iteration, and the Theta, the angle between the normal vector of the target and the beam

Table 24. Estimated Azimuth, Elevation, and Mirror Distance for 10 Trial Test Run

	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5	Trial 6	Trial 7	Trial 8	Trial 9	Trial 10
Azimuth (est)	38.062°	39.557°	40.131°	39.677°	37.578°	38.48°	36.935°	39.185°	42.056°	38.135°
Elevation (est)	46.697°	46.492°	46.898°	47.885°	49.653°	49.576°	49.648°	50.317°	50.204°	48.321°
Mirror Distance (est)	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m	0.18519m
RLS-2D Iterations	275	270	242	256	260	344	287	275	195	263
RLS-3D Iterations	326	387	291	273	215	500+	298	337	80	420
ERLS Iterations	326	387	291	273	215	500+	298	337	80	420
WRLS Iterations	324	396	283	290	214	500+	273	373	77	488
EWRLS Iterations	324	396	283	290	214	500+	273	373	77	488
Convergence value	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$	$< 2x10^{-4}$

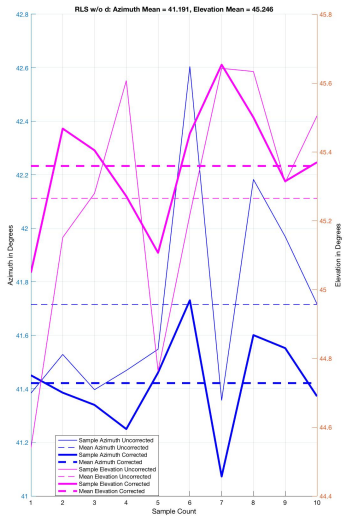


Figure 113. RLS-2D Algorithm Results

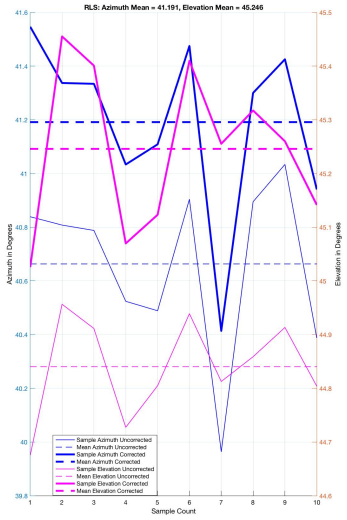


Figure 114. RLS-3D & ERLS Algorithm Results

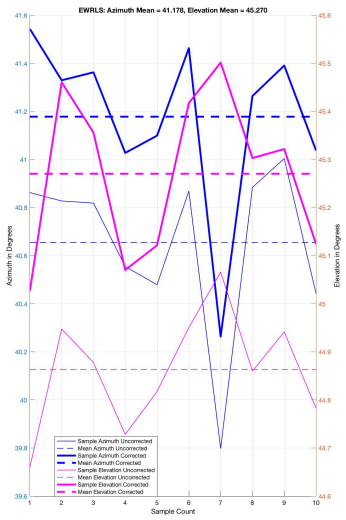


Figure 115. EWRLS & WRLS Algorithm Results

path. Again, the entries in bold correspond to the iterations with high standard deviation values as in Table 22. This was collected to help identify, if any, a correlation between the residuals, angle to the normal vector, and the sensor range return. It was anticipated that the residuals may be accounted for by the error in the range function due to the application of the range function correction factor. A correlation has not been identified at this time.

For the sake of completeness, a constant initial estimate for azimuth, elevation, and mirror distance as seen in Table 24 is used for all trials to test the RLS convergence and resulted with an almost exact outcome as the previously mentioned results in Table 22 down to four decimal places. No noticeable difference appears on the graphs or data and for this reason it is not displayed. This shows that the instantaneous center of mass of the

Table 25. Mean Sensor Range Returns and Residuals of Center Element for the 10 RLS Test Trials

	Target 1	**Target 2**	**Target 3**	**Target 4**	**Target 5**	**Target 6**	**Target 7**
Trial 1	3.1220m	2.7055m	2.7059m	2.6376m	2.6223m	2.4581m	
Residual (1)	1.46mm	5.29mm	5.97mm	-12.43mm	3.58mm	-8.70mm	
Theta to Normal	22.415°	21.206°	21.233°	50.294°	56.38°	47.825°	
Trial 2	2.2933m	2.4867m	2.7848m	2.8013m	2.9682m	2.8114m	2.8606m
Residual (2)	-2.42mm	1.50mm	1.26mm	-1.40mm	-0.249mm	1.58mm	0.523mm
Theta to Normal	54.438°	58.965°	37.14°	42.415°	36.479°	45.552°	57.251°
Trial 3	2.8012m	2.7509m	2.955m	2.4436m	2.8209m	2.8653m	2.5237m
Residual (3)	-2.57mm	0.379mm	-1.26mm	5.51mm	0.571mm	0.216mm	-5.77mm
Theta to Normal	52.219°	60.131°	39.466°	60.989°	50.599°	50.723°	60.989°
Trial 4	2.6552m	2.8602m	2.7314m	2.3787m	2.237m	**2.2297m**	
Residual (4)	-1.88mm	5.72mm	-1.51mm	-0.659mm	0.339mm	**0.970mm**	
Theta to Normal	53.644°	45.391°	53.765°	40.726°	43.039°	**54.879°**	
Trial 5	2.2377m	2.2754m	2.3732m	2.1738m	2.2647m		
Residual (5)	-0.503mm	0.721mm	-0.163mm	-0.231mm	0.228mm		
Theta to Normal	54.607°	49.603°	52.059°	46.669°	57.254°		
Trial 6	2.1011m	2.186m	2.2809m	2.3182m	2.2176m		
Residual (6)	2.63mm	-4.08mm	1.65mm	0.362mm	-0.470mm		
Theta to Normal	57.902°	48.367°	49.335°	28.171°	16.451°		
Trial 7	2.0681m	2.1605m	2.3277m	2.3675m	2.2642m	**2.0759m**	
Residual (7)	-19.78mm	14.54mm	-0.350mm	-5.39mm	3.34mm	**19.28mm**	
Theta to Normal	40.387°	35.717°	41.125°	46.387°	43.175°	**55.675°**	
Trial 8	2.0878m	2.2688m	2.3535m	**2.4874m**	2.5026m	2.3841m	**2.0905m**
Residual (8)	-4.06mm	0.703mm	2.46mm	**-3.03mm**	0.455mm	-0.373mm	**2.99mm**
Theta to Normal	55.646°	46.748°	47.587°	**53.97°**	56.658°	31.296°	**42.454°**
Trial 9	2.0839m	2.218m	2.3579m	2.5178m	**2.552m**		
Residual (9)	-2.16mm	3.73mm	0.442mm	-2.67mm	**3.02mm**		
Theta to Normal	43.423°	26.757°	28.152°	43.895°	**57.397°**		
Trial 10	2.4048m	**2.4946m**	2.4576m	2.520m	2.4065m	**2.2429m**	
Residual (10)	0.547mm	**-3.97mm**	1.82mm	-3.53mm	6.87mm	**-7.63mm**	
Theta to Normal	67.483°	**55.434°**	55.68°	35.919°	25.737°	**41.914°**	

board is close enough to the true values to allow the RLS algorithm to converge, and with practicality in mind, the calculated estimate will be used instead of the hard coded constant values for the rest of the experiments.

4.8 Calibration Procedure

The first step in evaluating the whole calibration procedure, a model needed to be built with the intent to transform the 2D LiDAR sensor into a 3D sensor. In order to develop an appropriate external superstructure to be mounted over top of the 2D scanning LiDAR

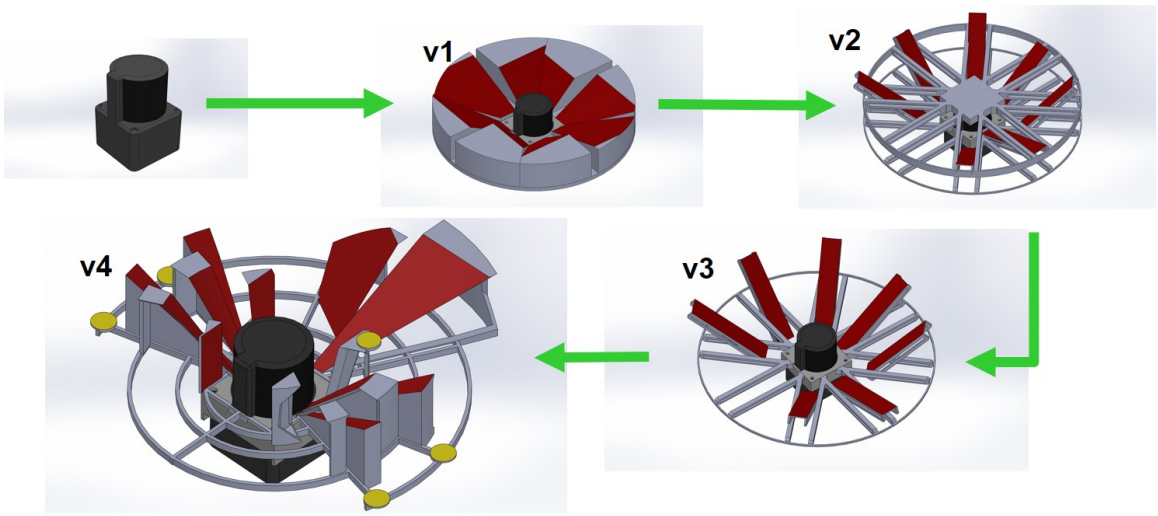


Figure 116. Illustration of Target Range for Intensity-Range Profile Relative to Laser Source

sensor, an iterative design process was conducted, and the summary of this can be seen in Figure 116. The initial prototype, v1, was intended as a rough baseline, with a few representative mirror surfaces shown in red at deflection angles close to 45°. This design is deceptively small in the figure, but the overall diameter was 10in, and was too large and heavy to place on a quad-rotor UAV so another design was investigated. The next design, v2, weighed significantly less, and movable mirrors were designed to facilitate a wider range of testing scenarios. This design also proved to be cumbersome and generally unusable. The top layer caused a lot of multi-path noise, in addition to significant edge effects due to the amount of sharp corners within the sensor field of view. The top layer was removed and the bottom layer was strengthened to produce the next prototype, v3. The prototype, v3, was used extensively in initial testing. The structure was relatively lightweight, and the mirrors were reconfigurable in elevation. For the the final test of the calibration procedure a more sophisticated design was desired. This is shown as v4.

The last prototype took a heuristic approach to mirror layout. A larger view can be

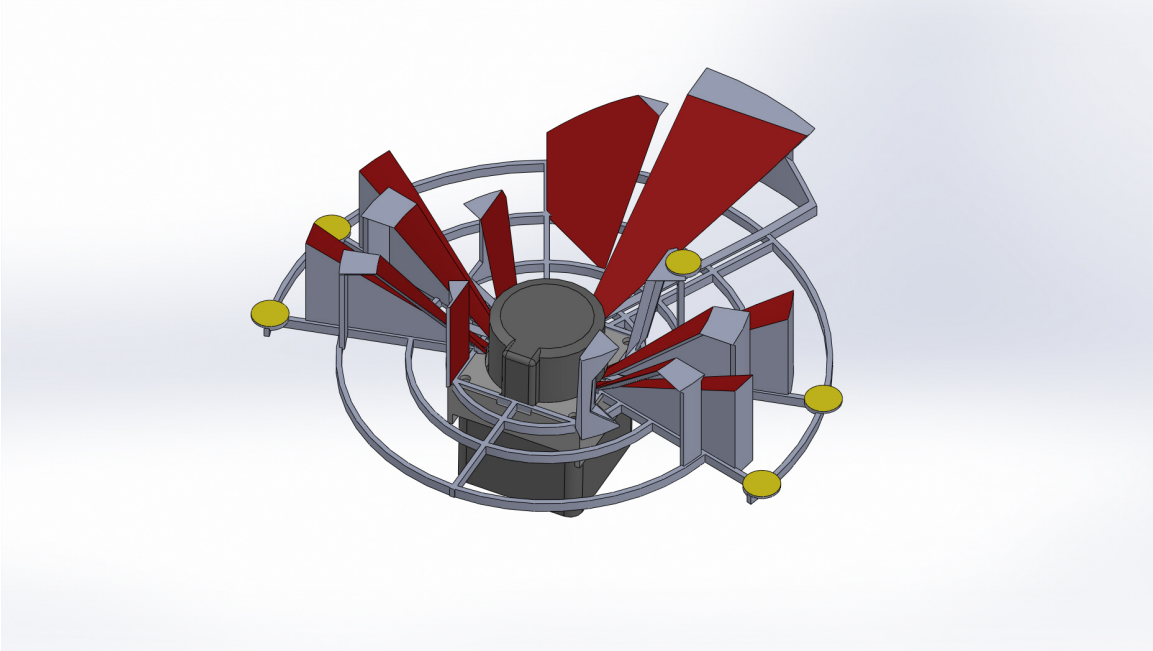


Figure 117. Prototype Mirror Superstructure Model

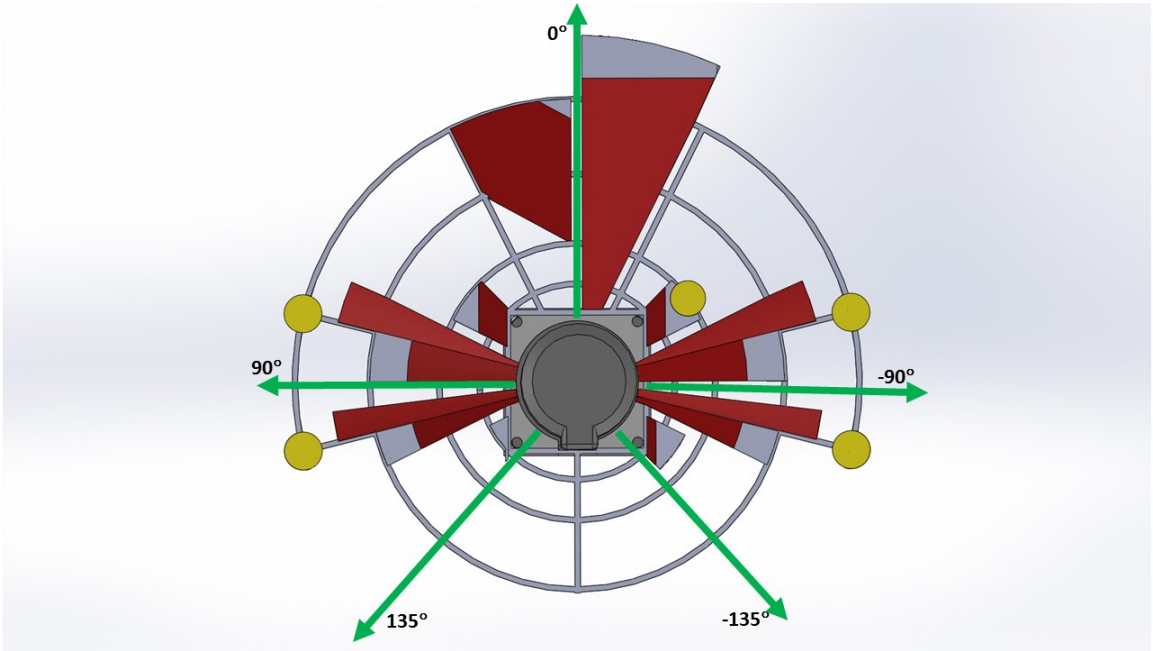


Figure 118. Overhead View of Prototype With Center Angle Identified

seen in Figure 117. Again, the red surfaces represent the locations of mounted mirrors and the locations in yellow are areas to place VICON markers, which do not affect the results of the calibration procedure. An additional view is presented in Figure 118 to show the sensor beam element orientation, and it also shows the entire field of view of the sensor, $\pm 135°$. With a final view of the sensor and structure in Figure 119. This view portrays the intended orientation of the sensor upon a UAV with respect to the direction of travel, where the direction of travel is along the X-axis.

The goal of this last prototype, v4, was to design something that could fit on a quad-rotor UAV and collect usable data for flight. There were openings in the structure to allow for unobstructed range measurements at every 45°. This would allow for a sense of the immediate vicinity. The two vertical mirrors on the top of the structure, as viewed in Figure 119, deflect the sensor beam into the blind-spot above the sensor (along the Z-axis). Four mirrors offer a 45° deflection for a 90° look angle towards the X-direction, the forward direction of travel. Four additional mirrors offer 45° look angle near the $\pm$ Y-axis. The bottom two vertical mirrors were a designed to range against the space +45° in both the Z-axis and the Y-axis on both sides of the X-axis. The last two mirrors on the bottom allow for beam deflection angle to face in the direction of flight but still angled downward toward the ground. This last prototype is still just a prototype used for the purpose of showing a proof-of-concept design. Further design optimization, in terms of both mirror construction and structure fabrication, is desired for a truly workable system.

Based off of initial results a representative set of samples of 13 possible locations from the original 270° FOV were selected and those areas of angle space are identified in Figure 120 in light gray. These 13 section are comprised of 370 beam elements, which is about 1/3rd of the original 1081 beam elements that are seen by the sensor. The v4 model was scanned with the Hokuyo laser rangefinder prior to mirror placement on the structure to determine a close estimate of the mirror distances which can be seen in green within the

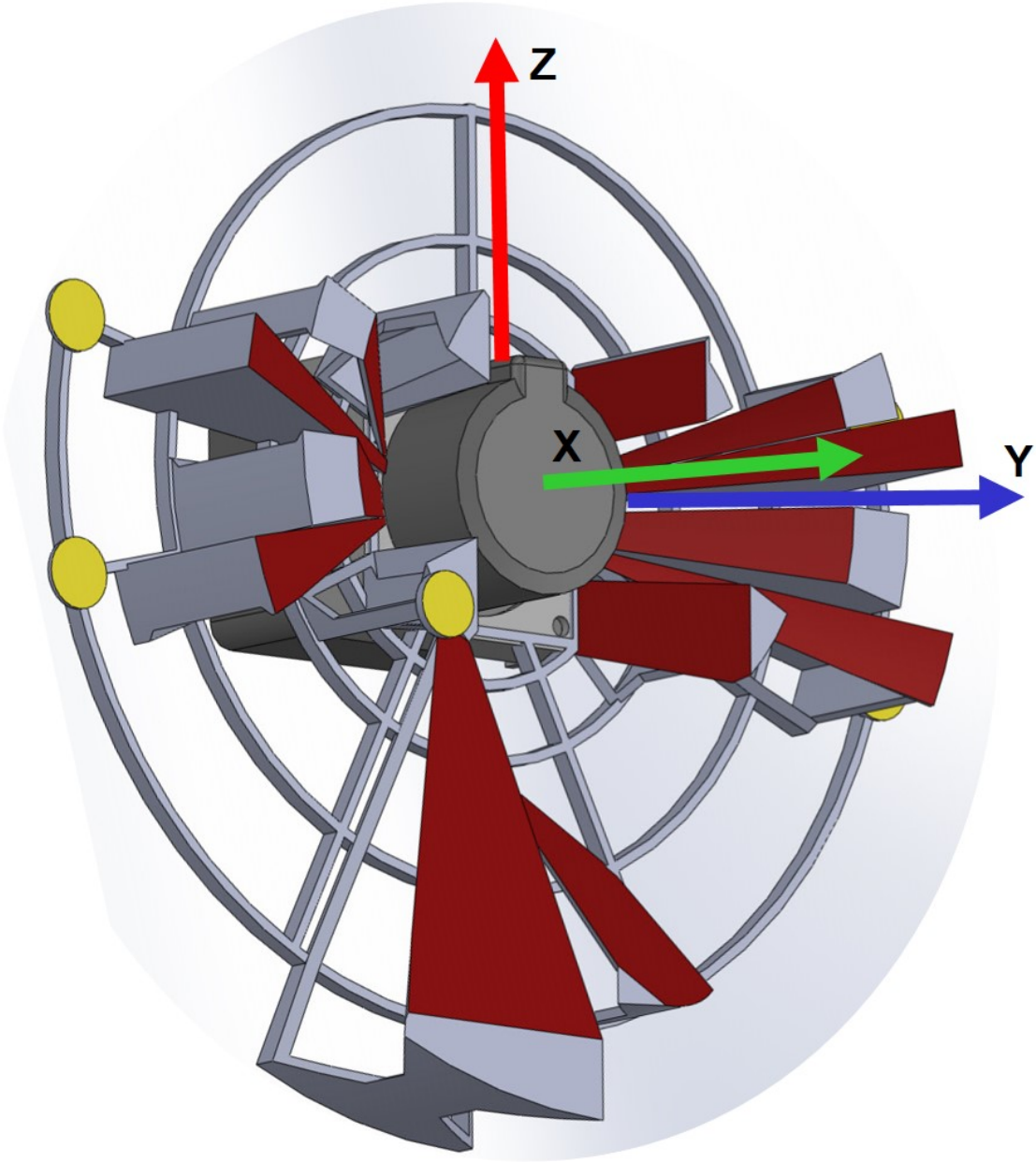


Figure 119. Diagram Depicting the Final Prototype with the Intended Orientation for Flight

light gray sections in 120.

After the mirrors were fabricated and secured to the structure, another test run was collected in open space with a maximum allowable beam propagation of about 11m. The representative set as seen in Figure 120 is verified as appropriate choices of angle space by evaluating the results. The results with a clear beam propagation path while simultaneously producing the low amount of variation during an initial data check as seen logarithmically in Figure 121 reasonably match the representative set of angle space chosen. It can been seen in Figure 121 that the locations with large variations are not used for the calibration test. The large variations are caused by edge effects, or are areas where the beam literally lands on an edge of a mirror. In either case the range cannot be accurately predicted at this time and those sensor beam locations are marked as unusable.

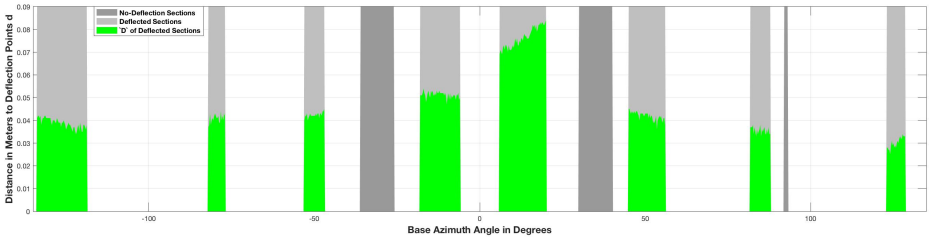


Figure 120. Usable Element Locations Using Prototype Mirror Structure

Table 26. Usable Element Locations Using Prototype Mirror Structure

Section	1	2	3	4	5	6	7	8	9	10	11
Angle Max	-118.75°	-77.0°	-47.0°	-26.0°	-6.0°	20.0°	40.0°	56.0°	87.75°	93.0°	128.0°
Angle Min	-133.75°	-82.0°	-53.0°	-36.0°	-18.0°	6.0°	30.0°	45.0°	81.75°	92.0°	123.0°
Est Azimuth	-139.84°	45.64°	135.03°	0.0°	72.36°	129.53°	42.70°	0.0	-15.03°	0.0°	-40.74°
Est Elevation	21.88°	84.76°	26.93°	0.0°	52.87°	61.40°	38.92°	0.0	82.95°	0.0°	15.15°

Three additional sections are identified in dark gray and represent locations without reflective surfaces. The data presented in green are the mirror distances, d, and each measured location after some data filtering eliminating faulty range returns from the test. The eight locations identified in light gray in Figure 120 are tested and the results can be seen in Figures 122 through 129. The angle space of all 11 sections can be seen in Table 26.

The estimated azimuth and elevation are also presented for reference. The estimates for azimuth are calculated with respect to the y-axis, therefore a 0° will correspond to a path on top of the y-axis.

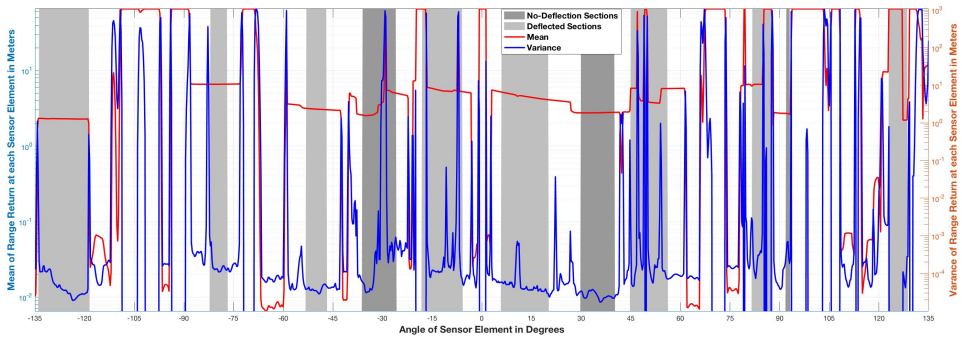


Figure 121. Variance and Mean of Usable Element Locations Using Prototype Mirror Structure

Figures 122 through 129 show the target board locations as triangles in light blue, the estimated azimuth and elevation angle of the beam deflection in red and the results of the RLS algorithm in dark blue. The sensor point of origin is indicated in Figure 122 with a blue circle and the point of beam deflection which is the same as the mirror distance, d, is indicated by the bend in the propagation path. In each of the eight range sections identified in Figure 120 the beam is deflected across a single flat mirror and the propagated beam will propagate as a straight line until a target is reached. An indicator on the validity of the results is if the converged results do propagate the beam segment as a line and not as an incoherent cluster or scattering of points.

In each of the eight cases illustrated in Figures 122 through 129 the converged results seem to propagate the beam as a line as is expected. The center of mass of each group of targets is indicated as a red line. The converged results also result in angles similar to those indicated in red which is also used as an additional 'soft' check. If the results converged to a different location or were not propagated as a line at the individual section, then it would be known that the RLS algorithm did not converge to reliable results at that location. This occurred during initial testing of the Range function, and the RLS algorithm, and those

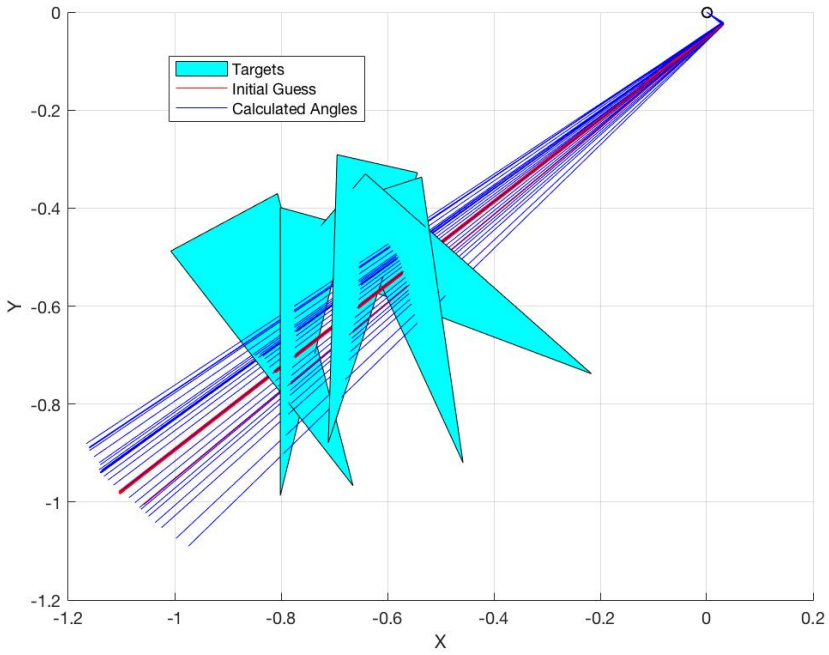


Figure 122. RLS Solution in -133.75°to -118.75°

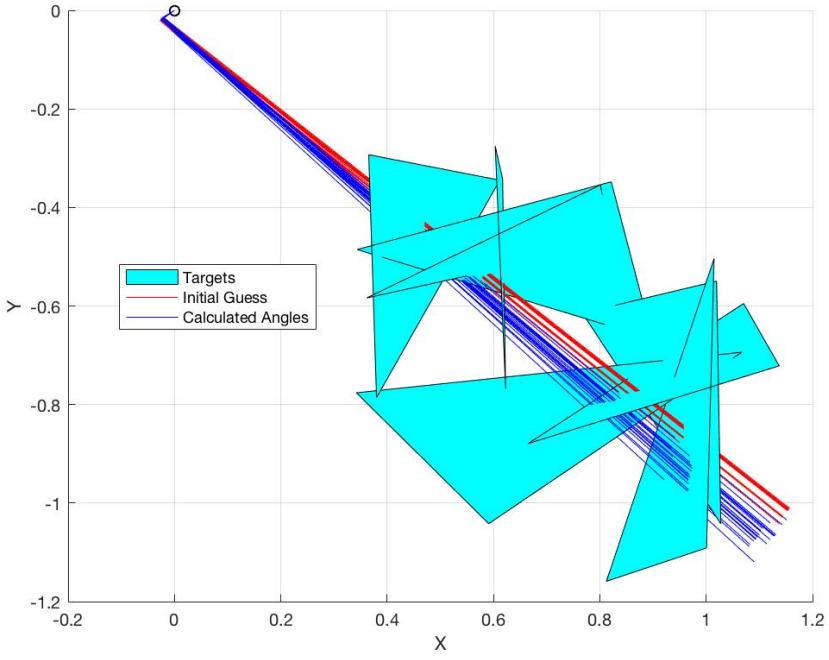


Figure 123. RLS Solution in 123.0°to 128.0°

results were used as part of the troubleshooting procedure which are not included here.

The design of the mirror structure is intended to give a representative example of possible mirror deflection angles. Figures 122 and 129 reflect the propagated beam at a slight elevation angle but at an azimuth greater than 90°. This set of angle space was chosen to range against a 'blind' spot for the sensor in the 2D planar view. Originally the Hokuyo sensor has a 270° field of view but by showing this angle of deflection is possible then the whole 360° could be attainable with the right mirror design.

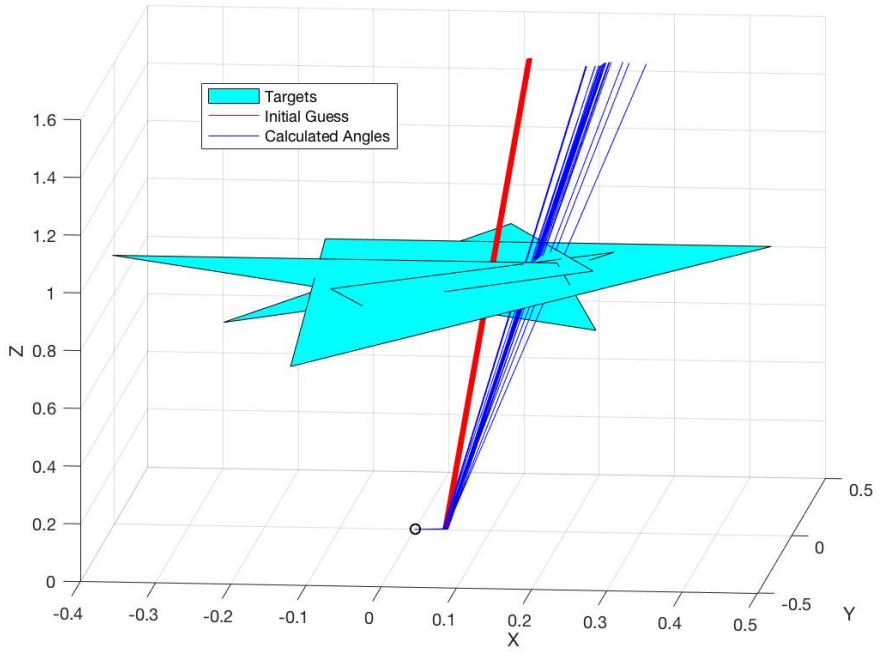


Figure 124. RLS Solution in -82.0°to -77.0°

Figures 124 and 125 represents an angle space with a deflection near 90° out of the original 2D plane of the sensor. Figure 122 shows a set of results where the estimated angle of deflection in red does not fall within the set of calculated angles of deflection. This shows that convergence in the RLS algorithm can happen even though the center of mass of the target group is not directly in line with the beam path. Results in this region show the possibility to range against target perpendicular to the sensor's original field of

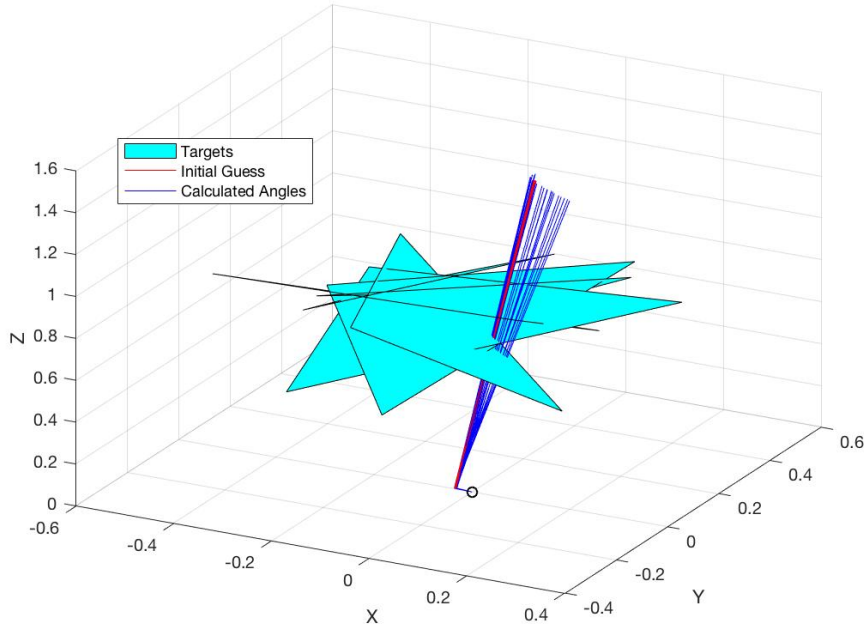


Figure 125. RLS Solution in 81.75.0°to 87.75°

view such as clearance above or below the sensor. Depending on the placement of the sensor on a UAV, this set of azimuth and elevation angles could be used to concentrate a large set of range points to a specific direction. If the sensor was positioned such that the top is facing the forward direction of the UAV, for example, the sensor would then show a much denser point cloud in the UAVs forward direction of motion.

The next set of possible azimuth and elevation angle represented by Figures 126 through 129 show a combination of angles between the aforementioned examples. Positive convergence results show that almost any spherical angle is achievable with properly designed reflection surfaces.

Next, the calibration results are compared between the EWRLS algorithm with two DOF (azimuth and elevation), and the RLS algorithm with three DOF (azimuth, elevation, and mirror distance). Figure 130 shows a dual y-axis presentation of the two DOF EWRLS results which plots azimuth and elevation vs. the angle. The eight different sections of

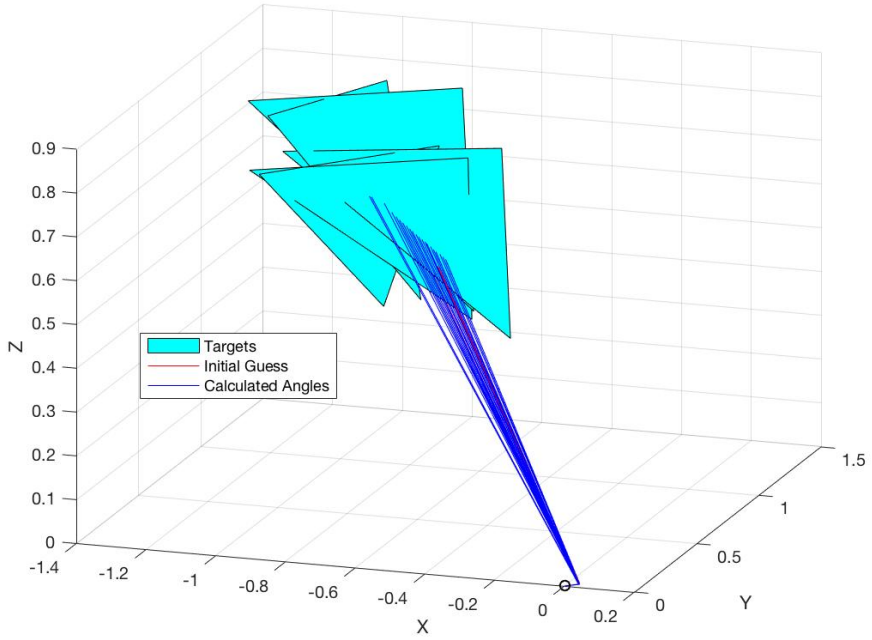


Figure 126. RLS Solution in -53.0°to -47.0°

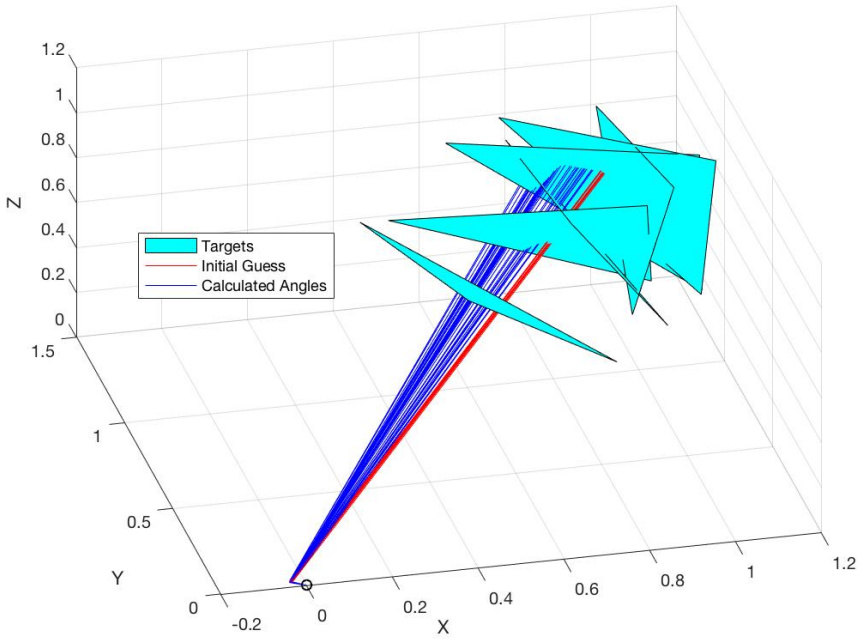


Figure 127. RLS Solution in 45.0°to 56.0°

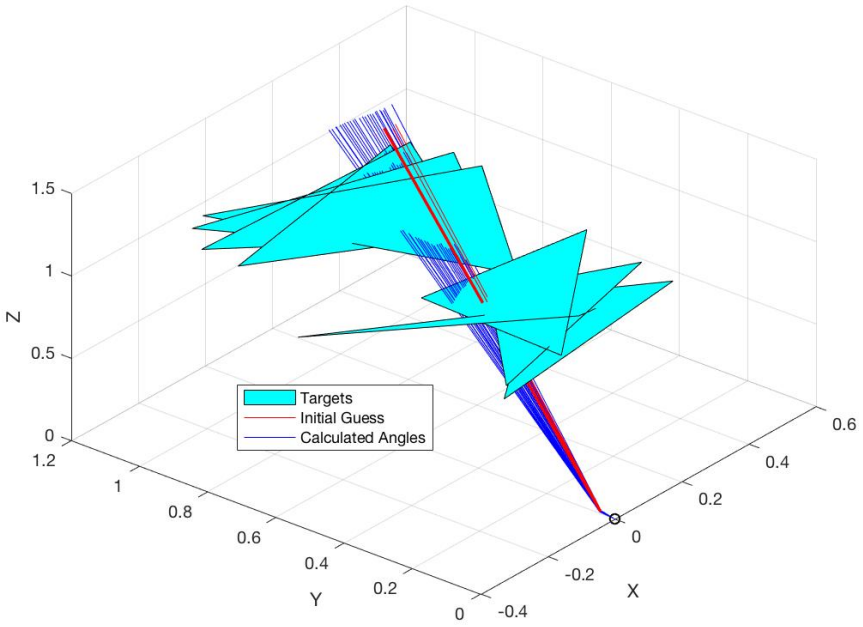


Figure 128. RLS Solution in -18.0°to -6.0°

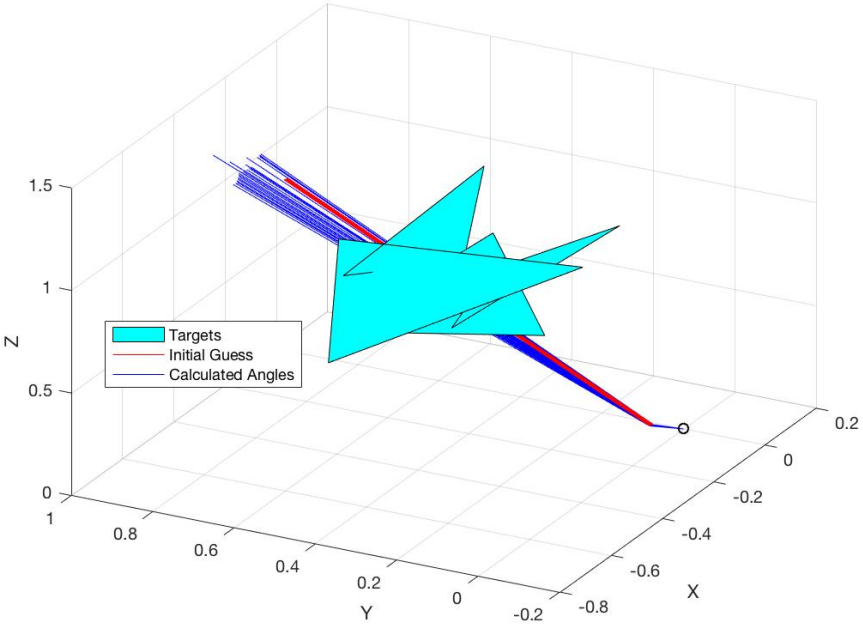


Figure 129. RLS Solution in 6.0°to 20.0°

angles referred to previously can be seen here. For each section the estimated azimuth is shown in red, the estimated elevation is shown in magenta, while the calculated values for azimuth and elevation are shown in dark blue and light blue respectively. The initial approximate values for azimuth and elevation are linear in each section whereas the calculated values in dark and light blue show more variation. When difference of scale is taken into account between azimuth and elevation, the variation appears to be within the same order of magnitude. One item of note is the variation within a section. When comparing the results between the third and fourth grouping from the left there is a larger difference between the elevation estimates and the calculated elevation values in the forth section than in the third. This likely due to the differences in target orientations between the third and forth set. The orientations of the targets in the forth set most likely did not allow for as much observability in the elevation parameter as the orientations of the targets in the third set. This highlights the importance of choosing an appropriate set of targets in this algorithm.

The next figure, Figure 131, presents the results of the three DOF EWRLS algorithm. Again, in each section the initial approximate azimuth is shown in red, the initial approximate elevation is shown in magenta, while the calculated values for azimuth and elevation are shown in dark blue and light blue respectively. In this case the mirror distance is used as an initial estimate and then the algorithm is allowed to converge to a new mirror distance along with the elevation and azimuth. When looking at the plot presented in Figure 131 and comparing it with Figure 130 it can be seen that the sections that show larger variation in the the two DOF solution are even greater in the three DOF solution, whereas the sections that converged nicely in the two DOF case converge just as well in the three DOF case. This is likely caused by the observability of the mirror distance parameter, d, with respect to the range function. The mirror distance requires relatively large changes to induce large changes in the range function result. This can cause some of the estimated mirror distances to converge to unrealistic results in order to minimize the RLS algorithm residuals

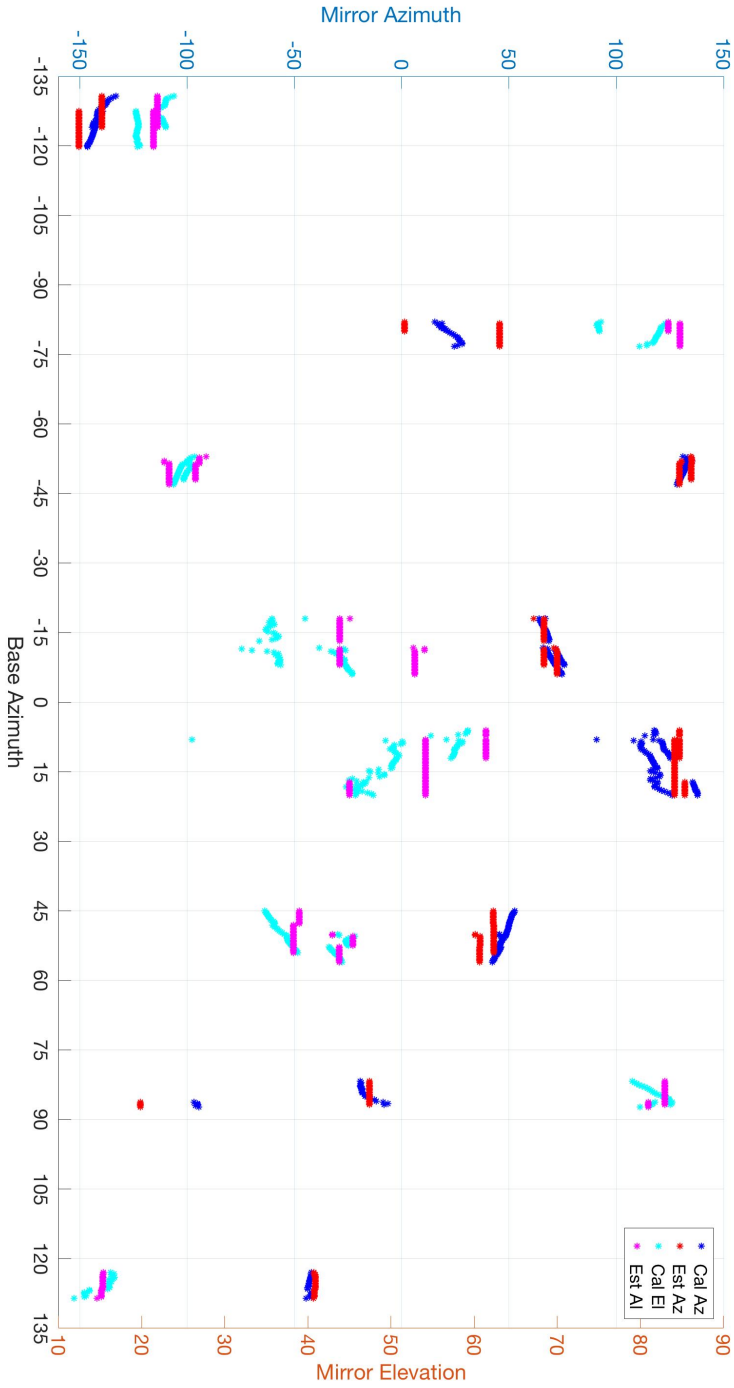


Figure 130. Illustration of EWRLS Algorithm Results Using 2 DOF

in Equation 38 from Section 3.4.

Continuing to evaluate the calibration algorithm, we can also look at the effect of using the the error correction term derived in Section 4.6. An arbitrary subset of residuals in the RLS algorithm from the full calibration using the corrected ranges, and the uncorrected ranges are displayed in Table 27. This is only a soft indicator that the error correction step is necessary because the precise locations of the azimuth and elevation not identifiable. The decrease in residuals from between using the error corrected range function and the uncorrected range function varies from sub-millimeter to a 7mm improvement in residuals for this subset.

Table 27. Comparison of RLS Algorithm Residuals with and without Error Correction

Sensor Element	No Error Correction	With Error Correction	Difference In Residuals	Sensor Element	No Error Correction	With Error Correction	Difference In Residuals	Sensor Element	No Error Correction	With Error Correction	Difference In Residuals
-130.75°	0.912mm	0.836mm	0.076mm	-51.25°	0.515mm	0.435mm	0.081mm	8.25°	1.557mm	1.384mm	0.172mm
-130.25°	0.755mm	0.684mm	0.071mm	-51°	0.505mm	0.425mm	0.081mm	8.5°	1.431mm	1.263mm	0.167mm
-129.5°	0.651mm	0.582mm	0.068mm	-50.75°	0.406mm	0.307mm	0.098mm	8.75°	8.635mm	1.423mm	7.21mm
-129.25°	0.619mm	0.552mm	0.067mm	-50.5°	0.504mm	0.423mm	0.081mm	9°	1.437mm	1.269mm	0.167mm
-128.75°	0.601mm	0.536mm	0.065mm	-50.25°	0.524mm	0.441mm	0.083mm	9.25°	1.457mm	1.288mm	0.169mm

The last test, as described in Section 3.4, evaluated the accuracy of the calibration algorithm, which was comprised of the results of the sensor characterization, the estimated range function, error modeling, and the RLS algorithm implementation. Using the same test setup as the test scene described in the RLS repeatability, but with a VICON marker used to mark the location of the sensor beam as it lands on the target, the true angles can be calculated and an error metric is produced. An arbitrary location within the usable angle space of the v4 model, shown previously in Figure 120, is selected for these trials. In this instance, the beam element at 11.5° is used, and 20 trials were conducted. Figure 132 illustrates the actual azimuth of the deflection angle in a solid blue line, and the actual elevation in solid red. The estimated values are displayed in the corresponding dotted lines. It can be seen that the Root Means Squared Error (RMSE) in azimuth is 0.36°, and that the RMSE in elevation is 0.24°.

Figure 131. Illustration of EWRLS Algorithm Results Using 3 DOF

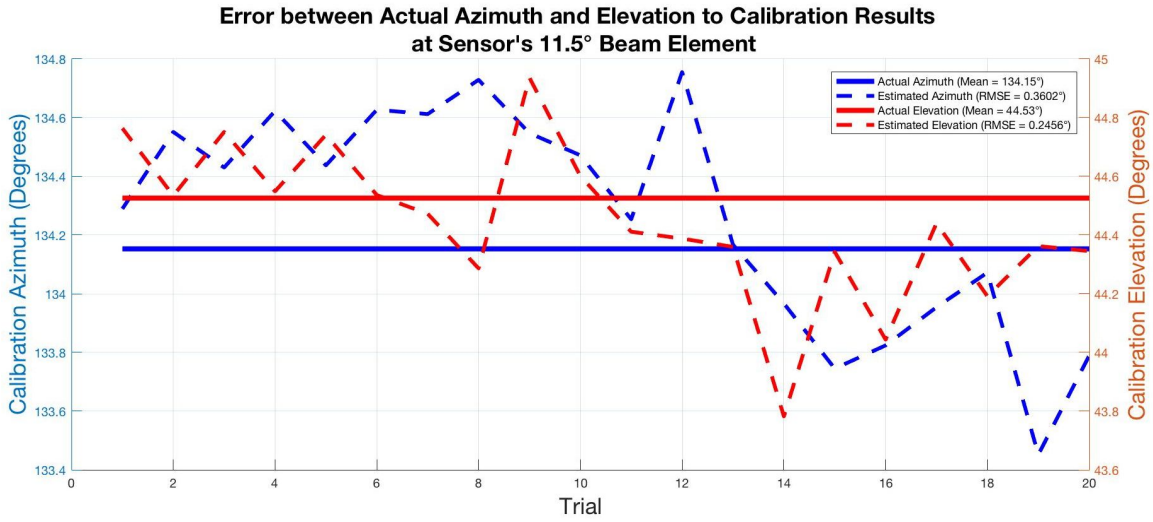


Figure 132. Error Between Actual and Estimated Angles w/o Using Error Correction

4.9 Summary

In this chapter, a series of experiments were conducted to investigate and analyze the results of the proposed calibration algorithm. It is shown that a thorough characterization of the LiDAR sensor is important in order to develop an accurate error model. This error model is then fed into a range function that is able to estimate the expected range return value as a function of target characteristics such as target shape (in this case planar), target shade of gray, target orientation, and target distance. Additionally, the effect of including a mirror in the propagation path of the sensor beam is characterized and included in the error model. As part of this characterization a few other important optical effects were identified. These effects included edge effects, and distortion which was not included in the error model but was included in the reflective superstructure design to minimize its effects. Testing of the range function based on the characterization of the Hokuyo laser rangefinder resulted in less than 1cm of range error at a target range of 4 meters.

Once the range function was developed, then a set of targets were designed to allow for RLS algorithm convergence. Azimuth, elevation, and deflection point information for each individual range element of the sensor was then forwarded to a transformation algorithm.

It was shown that recursive least square approximation of the parameters is achievable, and repeatable with a properly defined range function. Additionally, it was found that there is no significant improvement between RLS, ERLS, WRLS, and EWLS. A possible improvement between RLS algorithms is the speed at which convergence occurs. In a post-processing environment, such as in this research, there is no significant reason to use anything other than the original RLS in order to keep the implementation more simple.

V. Conclusion

This research proposed equipping an existing 2D LiDAR sensor with a reflective superstructure to deflect the projected beam pattern to other areas of interest, thereby converting it into a 3D system and reducing the need for additional sensors in autonomous UAV applications which usually require a full suite of sensors. This thesis described the prototype design and testing with an emphasis on calibration algorithms. Once calibrated, the modified sensor allows for a smaller physical and computational footprint to perform navigation and mapping techniques with a single sensor in GPS-denied environments. By modifying the existing scanning LiDAR sensor, the data presented by the sensor can be exploited beyond the initial layer of the results.

The issue with adding sensors is the increase in energy consumption and weight, with a corresponding decrease in flight time, for each additional sensor. Choosing to use an externally mounted reflective structure also eliminates the need for additional moving parts, such as actuators for sensor re-orientation, that have a much higher failure rate than rigid parts. Additionally, by minimizing the number of sensors, and by limiting the new orientation of the beam to a static set of parameters, the transformation between the 2D output can be pre-calculated before flight during the proposed calibration algorithm. Once calculated, the transformation can be implemented in real-time during flight by simple matrix multiplication in order to sense a 3D environment.

5.1 Summary of Results

In Chapter 3, a process to characterize a sensor with a fabricated target was described. The tests designed characterized the statistical return of the sensor when coupled with a specific type of target, which also accounted for the the color, shape, and orientation of the target. The target was designed such that the ability to mathematically represent the

target in the algorithm was simplified. By characterizing the sensor with the target, an error model was able to be derived to account for most range discrepancies in the sensor range return with this predetermined target.

Once the error model was finalized, an expected range function was developed and used in a recursive least square algorithm. The range function is developed to account for a reflected sensor beam by a static azimuth, θ, and a static elevation, ψ, with a point of reflection at a distance of 'd' from the sensor point of origin. Next a series of tests were presented to allow for the RLS algorithm to converge to an estimate of the angles of azimuth and elevation.

Chapter 4 then discussed the results of the proposed experiments in Chapter 3. The first five tests were used to characterize the output of the sensor as a function of target distance, target orientation, and target color. The tests verified that the range return increasingly followed a Gaussian distribution as the the target range increased, and also illustrated the effect of the target color and target orientation on the returned sensor ranges. This information was then used to develop an error model which was able to lower the expected range function error to less than 2cm.

The next set of tests showed that the RLS function can converge reliably across of series of ten test runs with a standard deviation of less than 1/2° for both azimuth and elevation, regardless of the RLS algorithm variant used, to include Weighted-RLS, Extended-RLS, and Weighted/Extended-RLS. It was also shown that no variant of RLS algorithm showed any relevant improvement over the others, but that if an unobservable, or only partially observable, parameter is used to define the convergence criteria, then unrealistic results may occur. Once the RLS function was shown to be consistent in convergence results, the final test of the whole calibration algorithm was conducted on a prototype of a mirrored structure. A beam location was randomly chosen within the usable angle space of the modified sensor, and a series of tests were conducted to evaluate the error between the

actual azimuth and elevation of deflection caused by the mirror at that location. The RMSE at that location was shown to have a combined error less than 1/2°, which equate to an approximate maximum angle error of 1.3cm at a range of 1.5m.

The results of the final test show that the algorithm converges near the estimated values, and that the resulting beam pattern in each of the areas tested mimic a beam pattern that is expected from a mirror with both an azimuth and elevation angle of deflection.

5.2 Future Work

The results presented in chapter 4 have pointed out several areas of improvement to this procedure, and the specific implementation of that procedure with the given prototype presented in this thesis with the Hokuyo laser rangefinder.

Sensor Characterization.

The range model developed to estimate the sensor range return for this research made conclusions that range return is a constant Gaussian shape with a consistent mean, and that the mean is the true range of the return if given more than 1000 samples of data to calculate that mean with. This assumption may be true for ranges greater than 4m but further investigation into the statistical characteristics of the range return is recommended at distances less than 4m. An additional area of improvement is through modeling the beam as a Gaussian-like rectangle. By propagating the beam as such, a more accurate error model can be realized. The only drawback is that each mirrored surface will required a independent characterization as the flatness, and other characteristics may not be constant between mirror after fabrication.

Superstructure Design.

A thorough investigation of the design of a reflective superstructure was not conducted due to time constraints. Multiple areas of improvement exists in this focus area. A critical component involves the field of view of the sensor with respect to each sensor range element location between $\pm 135°$. Mirror construction without edges in the sensor field of view, coupling this with a much closer deflection point on the reflective surfaces could allow for more possible reflection points within the angle space of the sensor, and a more thorough point cloud to represent the 3D environment.

Data Collection Procedure.

This research presented a data collection procedure based on collecting the range data being ranged off of a series of targets for each defined set of angle space. Each set of angle space within the final test included a minimum of five targets within a nominal orientation bound to help ensure algorithm convergence. This data collection procedure is time consuming and a reattempt of the procedure should be made to test all the angles of the sensor simultaneously.

Mapping and Navigation.

The next step in testing is to perform a series of tests to validate the results of the calibration procedure. A series of scans of a known environment as the sensor is rotated and translated should be conducted. Using the results of the calibration procedure, a valid representation of the environment should be completed to validate the results of the calibration algorithm. Once completed, a real-time implementation of the results can be used for position and velocity implementation.

5.3 Final Remarks

A new approach at converting a 2D scanning LiDAR sensor to be used as a 3D system was introduced. In this approach, a proof of concept design was developed and a the proposed calibration algorithm was conducted on the design. The results showed that less than 1/2° of RMS Error in both accuracy and repeatability is achievable.

The calibration algorithm presented in this research is based on a specific reflective superstructure attached to the specific scanning LiDAR sensor. Once calibrated, the results of the sensor-structure combination can be implemented without the need to re-calibrate if the structure is secured properly to the sensor. These results bring the possibility of simultaneous localization and mapping with a single sensor on a small UAV equipped with an inertial navigation unit closer to realization.

Bibliography

1. J. W. Goodman, *Statistical Optics*, 1st ed. Hoboken, NJ, USA: John Wiley and Sons, Inc., 2000.

2. Hokuyo ust-20lx scanning laser rangefinder. Accessed on 2017-02-01. [Online]. Available: https://acroname.com/products/HOKUYO-UST-20LX-LASER?sku=R360-UST-20LX

3. O. O. W. Book. Surfaces: Lambertian brdfs. Accessed on 2017-02-01. [Online]. Available: http://www.oceanopticsbook.info/view/surfaces/lambertian_brdfs

4. Geometrics. What is snells law? Accessed on 2017-02-01. [Online]. Available: http://www.geometrics.com/what-is-snells-law/

5. J. S. K. I. S. K. Dong Geol Choi, Yunsu Bok, "Extrinsic Calibration of 2-D Lidars Using Two Orthogonal Planes," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 83–98, 2016.

6. W.-J. L. Chien-Chou Lin, Yan-Deng Liao, "Calibration Method for Extending Single-Layer LIDAR to Multi_layer LIDAR," *System Integration (SII), 2013 IEEE/SICE International Symposium on*, pp. 677–681, 2013.

7. A. Yaacobi, "Integrated optical phased arrays for LiDAR applications," Ph.D. dissertation, MIT, 2015.

8. Kinect infrared projection laser dots. Accessed on 2017-02-02. [Online]. Available: http://www.howtomakeonline.org/ixkugmoneyjmrdxd/kinect-infrared-projection-laser-dots.html

9. B. Bellekens, V. Spruyt, R. Berkvens, R. Penne, and M. Weyn, "A Benchmark Survey of Rigid 3D Point Cloud Registration Algorithms," vol. 8, no. 1, pp. 118–127, 2015.

10. 1st surface mirrors: Technical specifications. Accessed on 2017-02-01. [Online]. Available: http://firstsurfacemirror.com/technical-specifications/

11. iphone: This is 7. Accessed on 2017-02-01. [Online]. Available: www.apple.com/iphone

12. Velodyne lidar: Hdl-64e. Accessed on 2017-02-01. [Online]. Available: http://velodynelidar.com/hdl-64e.html

13. Waymo: Self-driving car. Accessed on 2017-02-01. [Online]. Available: https://waymo.com

14. S. T. Jonathan Brookshire, "Automatic Calibration of Multiple Coplanar Sensors," *Proceedings of Robotics: Science and Systems*, pp. 33–40, 2011.

15. J. W. Goodman, *Introduction to Fourier Optics*, 2nd ed. Hoboken, NJ, USA: McGRAW-HILL, 2000.

16. E. W. Max Born, *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*, 7th ed., ser. Springer Tracts in Advanced Robotics. Cambridge, UK: Cambridge University Press., 1999, p. 986. ISBN 9780521642224.

17. E. A. A. D. M. Tournear, M. A. Hoffbauer, "Gamma-ray waveguides," *Applied Physics Letters*, 2008, dOI: http://dx.doi.org/10.1063/1.2905280.

18. C. O. J. S. A. Jarre, C. Fuhse, "X-ray waveguides - a new tool for high-resolution microscopy," *Physical Review Letters*, vol. 94, 2005, dOI: http://dx.doi.org/10.1063/1.2905280.

19. M. P. Society. (2012, may) The new world of gamma-ray optics. [Online]. Available: http://phys.org/news/2012-05-world-gamma-ray-optics.html

20. M. Katz, *Introduction to Geometrical Optics*, 2nd ed. World Scientific Publishing Co Pte Ltd, dec 2002, iSBN: 978-981-238-202-3.

21. M. N. Polyanskiy, "Refractive index database," http://refractiveindex.info, accessed on 2017-02-01.

22. Geoslam: Zeb-revo. Accessed on 2017-02-02. [Online]. Available: http://geoslam.com/hardware-products/zeb-revo/

23. H. C. W. C. Hsu, P. Shih and J. K. Liu, "A study on factors affecting airborne lidar penetration," *Terr. Atmos. Ocean. Sci.*, vol. 26, no. Part 2, pp. 241–251, 7 2015.

24. P. W. Milonni and J. H. Eberly, *Laser Resonators and Gaussian Beams, in Laser Physics*, 1st ed. Hoboken, NJ, USA: John Wiley and Sons, Inc., 2010.

25. R. L. B. Tso Yee Fan, "Diode Laser-Pumped Solid-State Lasers," *IEEE Journal of Quatnum Electronics*, vol. 24, no. 6, pp. 895–912, 1988.

26. J. D. Schmidt, *Numerical Simulation of Optical Wave Propagation With examples in MATLAB*, 1st ed. SPIE P.O. Box 10 Bellingham, Washington 98227-0010 USA: SPIE, 2010.

27. R. D. Richmond, *Direct-Detection LADAR Systems*. SPIE Press, 2010, vol. TT85.

28. F. E. Nicodemus, "Directional reflectance and emissivity of an opaque surface," *Appl. Opt.*, vol. 4, no. 7, pp. 767–775, Jul 1965. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-4-7-767

29. G. Waldman, *Introduction to Light: The Physics of Light, Vision, and Color*, ser. Dover books on physics. Dover Publications, 2002. [Online]. Available: https://books.google.com/books?id=PbsoAXWbnr4C

30. Ust-10lx/20lx product page. Accessed on 2017-02-01. [Online]. Available: https://www.hokuyo-aut.jp/02sensor/07scanner/ust_10lx_20lx.html

31. J. Greenfeld, "Terrestrial and mobile lidar mapping systems: Understanding the technology, applications and opportunities," 2015, http://c.ymcdn.com/sites/www.njspls.org/resource/resmgr/SurvCon2015Handouts/NJSPLSTLSandMLSr

32. (2010) Flash lidar technology overview. Advanced Scientific Concepts Inc Presentation Posted by NASA 26 March 2010. [Online]. Available: http://ssco.gsfc.nasa.gov/workshop_2010/day3/Roger_Stettner/Stettner_ASC_Workshop_Presentation.pdf

33. S. C. John A. Christian, "A survey of lidar technology and its use in spacecraft relative navigation," in *Survey of Current Technology*. American Institute of Aeronautics and Astronautics, 2014.

34. M. E. Romano, "Commercial geiger mode lidar," Harris Radio Corp: Public Presentaion, apr 2015. [Online]. Available: https://www.hokuyo-aut.jp/02sensor/index.html#scanner

35. "Scanning range finder," SOKUIKI Sensor Product Catalog, accessed on 2017-02-01. [Online]. Available: https://www.hokuyo-aut.jp/02sensor/index.html#scanner

36. "Hdl-64e," Velodyne Product Page, accessed on 2017-02-01. [Online]. Available: http://velodynelidar.com/hdl-64e.html

37. A. E. G. Bradly J. Cooke, "Field imaging radar/lidar through fourier transform heterodyne," Los Alamos National Laboratory for United States Naval Academy, Tech. Rep., jan 1999.

38. B. J. Cooke, A. E. Galbraith, B. E. Laubscher, and C. E. Strauss, "FIELD IMAGING RADAR/LIDAR THROUGH FOURIER TRANSFORM HETERODYNE," Los Alamos National Laboratory, Tech. Rep. January, 1999.

39. A. V. Jelalian, *Laser Radar Systems*. Artech House, 1992.

40. C. Mobley. Surfaces: Lambertian brdfs. Accessed on 2017-02-01. [Online]. Available: http://www.oceanopticsbook.info

41. C. G. Bachman, *Laser Radar Systems and Techniques*. Artech House, 1979.

42. S. M. Kay, *Fundamentals of Statistical Image Processing*, ser. Detection Theory. Prentice Hall, 2013, vol. 2.

43. J. B. Yoichi Okubo, Cang Ye, "Characterization of the Hokuyo URG-04LX Laser Rangefinder for Mobile Robot Obstacle Negotiation," vol. 7332, pp. 1–10, 2009.

44. M. D. Adams, "Lidar design, use, and calibration concepts for correct environmental detection," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 753–761, 2000.

45. N. Muhammad and S. Lacroix, "Calibration of a rotating multi-beam Lidar," pp. 5648–5653, 2010.

46. Sick usa, sensor intelligence. Accessed on 2017-02-02. [Online]. Available: https://www.sick.com/us/en/

47. A. F. Habib, A. P. Kersting, A. Shaker, and W. Y. Yan, "Geometric calibration and radiometric correction of LiDAR data and their impact on the quality of derived products," *Sensors*, vol. 11, no. 9, pp. 9069–9097, 2011.

48. W. X. Y. W. Wenyu Chen, Jia Du, "A Noise-Tolerant Algorithm for Robot-Sensor Calibration Using a Planar Disk of Arbitrary 3-D Orientation," pp. 1–14, 2016.

49. F. D. J. C. H. Z. Mengwen He, Huijing Zhao, "Pairwise LIDAR Calibration Using Multi-Type 3D Geometric Features in Natural Scene," pp. 1828–1835, 2013.

50. R. I. Hartley, “An investigation of the essential matrix,” accessed on 2017-02-02. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.64.7518&rep=rep1&type=pdf

51. J. Quenzel, N. Papenberg, and S. Behnke, “Robust Extrinsic Calibration of Multiple Stationary Laser Range Finders,” *12th IEEE International Conference on Automation Science and Engineering (CASE)*, no. August, pp. 1332–1339, 2016.

52. S. Debattisti, L. Mazzei, and M. Panciroli, “Automated extrinsic laser and camera inter-calibration using triangular targets,” *IEEE Intelligent Vehicles Symposium, Proceedings*, no. Iv, pp. 696–701, 2013.

53. G. Pandey, J. R. Mcbride, S. Savarese, and R. M. Eustice, “Automatic Targetless Extrinsic Calibration of a 3D Lidar and Camera by Maximizing Mutual Information,” *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pp. 2053–2059, 2012.

54. H.-J. C. Po-Sen Huang , Wen-Bin Hong and C.-Y. Chen, “EXTRINSIC CALIBRATION OF A MULTI-BEAM LIDAR SYSTEM WITH IMPROVED INTRINSIC LASER PARAMETERS USING V-SHAPED PLANES AND INFRARED IMAGES,” *IVMSP Workshop*, vol. 11, pp. 3–6, 2013.

55. C. X. Guo and S. I. Roumeliotis, “An analytical least-squares solution to the line scan LIDAR-camera extrinsic calibration problem,” *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2943–2948, 2013.

56. G. Atanacio-Jiménez, J. J. González-Barbosa, J. B. Hurtado-Ramos, F. J. Ornelas-Rodríguez, H. Jiménez-Hernández, T. García-Ramirez, and R. González-Barbosa, “LIDAR velodyne HDL-64E calibration using pattern planes,” *International Journal of Advanced Robotic Systems*, vol. 8, no. 5, pp. 70–82, 2011.

57. Rp-lidar: Low cost 5hz 360 deg lidar. Accessed on 2017-02-02. [Online]. Available: http://www.robopeak.com/blog/?p=278

58. Sweep: Scanning lidar for everyone. Accessed on 2017-02-02. [Online]. Available: https://www.kickstarter.com/projects/scanse/sweep-scanning-lidar

59. M. Lieber, C. Weimer, M. Stephens, and R. Demara, "Development of a validated end-to-end model for space-based lidar systems," *Proc. of SPIE*, vol. 6681, p. 66810F, 2007. [Online]. Available: http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.731277

60. P. Church, J. Matheson, and G. Roy, "Steerable LiDAR platform based on a 3D laser scanner using prisms," pp. 1–3, 2017. [Online]. Available: http://spie.org/newsroom/6819-steerable-lidar-platform-based-on-a-3d-laser-scanner-using-prisms

61. C. Weimer and T. Ramond, "An Electronically Steerable Flash Lidar (ESFL)," *Proc. Earth Sci. Technol. Forum*, 2011. [Online]. Available: http://esto.nasa.gov/2012test/conferences/estf2011/papers/Weimer{_}ESTF2011.pdf

62. K. Khoshelham and S. O. Elberink, "Accuracy and resolution of kinect depth data for indoor mapping applications," *Sensors*, vol. 12, no. 2, pp. 1437–1454, 2012.

63. Kinect sensor. Accessed on 2017-02-02. [Online]. Available: https://msdn.microsoft.com/en-us/library/hh438998.aspx

64. Y. Zhou, G. Jiang, G. Xu, X. Wu, and L. Krundel, "Kinect depth image based door detection for autonomous indoor navigation," *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication*, vol. 2014-Octob, no. October, pp. 147–152, 2014.

65. D. T. Venable, "Implementation of a 3D Imaging Sensor Aided Inertial Measurement Unit Navigation System," no. August, 2008.

66. Z. J.-C. S. M. E. C. Floreano, D., *Flying Insects and Robots*, 1st ed. Springer-Verlag Berlin Heidelberg, 2010, iSBN: 978-3-540-89392-9, DOI: 10.1007/978-3-540-89393-6.

67. M. a. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381 – 395, 1981.

68. K. G. Derpanis, "Overview of the RANSAC Algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2–3, 2010.

69. J. Elseberg, D. Borrmann, and N. Andreas, "Efficient Processing of Large 3D Point Clouds," in *Information, Communication and Automation Technologies (ICAT), 2011 XXIII International Symposium on*, 2011.

70. J. Han, F. Wang, Y. Guo, C. Zhang, and Y. He, "An improved RANSAC registration algorithm based on region covariance descriptor," *Proceedings - 2015 Chinese Automation Congress, CAC 2015*, pp. 746–751, 2016.

71. D. J. Raquet, "Vision-aided navigation," Consortium of Ohio Universities on Navigation and Timekeeping Presentation, may 2014.

72. P. Besl and N. McKay, "A Method for Registration of 3-D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

73. S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm," in *Third International Conference on 3D Digital Imaging and Modeling*. IEEE, 2001.

74. P. J. Huber, "Robust Estimation of a Location Parameter," *The Annals of Mathematical Statistics*, vol. 35, no. 1, pp. 73–101, 1964.

75. S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm," in *3-D Digital Imaging and Modeling, 2001. Proceedings. Third International Conference on*. IEEE, 2011.

76. J. F. F. Robert A Adams, *Sobolev Spaces*, 2nd ed. Academic Press, 2003, iSBN: 978-0-12-044143-3.

77. S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," *Computer Graphics Forum (Symposium on Geometry Processing)*, vol. 32, no. 5, pp. 1–11, 2013.

78. M. P. V. S Kirkpatrick, C D Gelatt Jr, "Optimization by simulated annealing," *Science*, vol. 20, no. 4598, p. 671680, 1983.

79. J. J. Leonard and H. F. Durrant-Whyte, "Simultaneous map building and localization for an autonomous mobile robot," *Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on*, no. 91, pp. 1442–1447 vol.3, 1991.

80. J. Mullane, B. N. Vo, M. D. Adams, and B. T. Vo, "A random-finite-set approach to Bayesian SLAM," *IEEE Transactions on Robotics*, vol. 27, no. 2, pp. 268–282, 2011.

81. I. Cheuk Yiu and S. K. Gupta, "Retrieving Matching CAD Models by Using Partial 3D Point Clouds," *Computer-Aided Design & Applications*, vol. 4, no. 5, pp. 629–638, 2007.

82. A. B. J. Jessup S. N. Givigi, "Robust and Efficient Multi-Robot 3D Mapping with Octree Based Occupancy Grids," in *IEEE International Conference on Systems, Man, and Cybernetics*, 2014, pp. 3996–4001.

83. A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap : An Efficient Probabilistic 3D Mapping Framework Based on Octrees," 2013.

84. B. P. G. J. F. T. F. J. L. R. W. Morgan Quigley, Ken Conley and A. Y. Ng, "Ros: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, 2009.

85. Ros/tutorials. Accessed on 2017-02-01. [Online]. Available: http://wiki.ros.org/ROS/Tutorials

86. Ros/turtlebot. Accessed on 2017-02-01. [Online]. Available: http://wiki.ros.org/Robots/TurtleBot

87. Macbook pro: A touch of genius. Accessed on 2017-02-01. [Online]. Available: http://www.apple.com/macbook-pro/

88. Trusty tahr/release notes - ubuntu wiki. Accessed on 2017-02-01. [Online]. Available: https://wiki.ubuntu.com/TrustyTahr/ReleaseNotes

89. Parallels desktop for mac. Accessed on 2017-02-01. [Online]. Available: http://www.parallels.com/products/desktop/

90. R. L. Y. Lorry. Vicons motion capture is embedded in the latest production of shakespeares the tempest. Accessed on 2017-02-01. [Online]. Available: https://www.vicon.com/press/vicons-motion-capture-is-embedded-in-the-latest-production-of-shakespeares-the-tempest

91. Ros.org - tf package summary. Accessed on 2017-02-01. [Online]. Available: http://wiki.ros.org/tf

92. Edmund optics optical mirrors. Accessed on 2017-02-01. [Online]. Available: http://www.edmundoptics.com/optics/optical-mirrors/

93. D. E. P. Henry C. Edwards, *Multivariable Calculus.*, 6th ed. Lebanon, IN: Prentice Hall, 2002.

94. J.-J. E. Slotine and W. Li, *Applied nonlinear control*. Englewood Cliffs (N.J.): Prentice Hall, 1991. [Online]. Available: http://opac.inria.fr/record=b1132812

95. P. Corke, *Robotics, Vision, and Control; Fundamental Algorithms in MATLAB*, 2nd ed., ser. Springer Tracts in Advanced Robotics. Springer, 2013, vol. 73.

96. P. Sup Premvuti, "Personal email correspondence," Kirinson Inc, U.S. Official Technical Representative of Hokuyo Automatic Co., Ltd., Nov 2016.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
23–03–2017	Master's Thesis	Sept 2015 — Mar 2017

4. TITLE AND SUBTITLE

Converting a 2D Scanning LiDAR to a 3D System for Use on Quad-Rotor UAVs in Autonomous Navigation

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Cooper, Matthew A., 1st Lt USAF

5d. PROJECT NUMBER

16G167

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENG-MS-17-M-019

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Research Laboratory,
Munitions Directorate
101 W. Eglin Blvd
Eglin AFB, FL 32542
COMM (850) 882-4600
Email: kevin.brink@us.af.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL/RWWI

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

A main concern with adding sensors is the increase in energy consumption and weight, with a corresponding decrease in flight time, for each additional sensor. This research proposes equipping an existing 2D LiDAR sensor with a reflective super-structure to deflect the projected beam pattern to other areas of interest, thereby converting it into a 3D system, and reducing the need for additional sensors in autonomous UAV applications. This research proposes an externally mounted lightweight reflective structure which eliminates the need for additional moving parts, such as actuators for sensor re-orientation, that have a much higher failure rate than a solid rigid parts.

15. SUBJECT TERMS

Autonomous, Indoor Navigation, LiDAR, 3D Sensor

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. J. F. Raquet, AFIT/ENG
U	U	U	U	200	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x4580; John.Raquet@afit.edu