Characterizing Water Flow from Multi-Tool Oscillation for Algae Harvest

A thesis submitted in partial fulfillment of the requirement
for the degree of Bachelor of Science in Physics
from the College of William and Mary in Virginia,

by

Collin P. Pampalone

Advisor: William Cooke

Senior Research Coordinator: Irina Novikova

Williamsburg, Virginia

April 2018

Contents

List of Figures

List of Tables

Abstract

In order to design a better harvesting tool for for an in-water algae growth system, I had to create a method of characterizing the water flow induced by the current tool in three dimensions. Using a GoPro Hero 3, I found equations that allow the tracking points and created a tank allows for an economical Particle Imaging Velocimetry system to be built. This system uses a tank of water with particles of rosin that are illuminated by a laser. I characterized the lens of the camera and created equations to convert displacement in the image to physical displacement. further, I created accurate derivatives of the equation that can be used to find the distance between points. I also produced a tank with a mirror mount to allow for a 3 dimensional view of the water flow.

Future work will focus on processing the video to track the particles, a vector map of the flow can be produced. This map can be further analyzed to differentiate laminar and turbulent flow. Finding which qualities of the blade on the cutting tool correlate to turbulent flow will help in optimizing the tool to better harvest algae.

Chapter 1

Introduction

1.1 Background

Pollution in the Chesapeake Bay, especially in the form of nutrient runoff, has been a problem for decades. The excess nutrients cause unstable algae blooms, which can block out sunlight for sea grasses, deplete oxygen for fish, and cause dead zones. These problems negatively impact the ecology of the Bay and its ability to provide natural resourses to the surrounding economies.

Virginia has developed a Watershed Improvement Plan (WIP) to reduce the amount of nutrients entering the Chesapeake Bay in order to conform with EPA total maximum daily load (TMDL) of nutrient emission. Many plans and methods to improve the water quality are costly relative to the percent of emissions they eliminate. [2]

In 2014 and 2015, a team from William and Mary and VIMS developed a method of cultivating and harvesting algae in Chesapeake Bay tributaries. Algae grows quickly and absorbs nutrients from the water. By creating a system of growing and harvesting the algae, nutrients can be effectively removed from the water system. Additionally, algae can be used as a source of bio-fuel, organic compost, or for a number of other applications. Because of the flexibility of algae as a commercial product, its sale could

offset the cost of this method of emission reduction

1.2 Current Issues

The current method of cultivation uses screens suspended underwater on which algae grows. The in-water algae growth system (IWAGS) consists of screens (substrate) floating underwater with algae growing on either side. The team found that, because algae grows exponentially, significantly more algae can be grown in a harvest cycle by leaving a layer of seed algae instead of removing all of it from the screen. Additionally, they found that scraping damaged the screen and made repairs more frequent. When the screen is removed from water, the algae lays flat and mats. When it remains underwater, it floats out from the screen making it possible to cut. A number of under water cutting methods were tested, and the best method they found used an off-the-shelf multi-tool saw. The saw uses an oscillating blade, and can be electrically powered or pneumatic. This versatility is useful for the saline environment in which the algae grows.

However, the multi-tool harvesting still has issues. The team found that different blades and vibrations speeds perform better or worse than others. Different shaped blades and frequencies cause different combinations of turbulent and laminar flow. The characteristic flow would move the algae in different ways, making the tool's ability to harvest unpredictable.

1.3 Goals

In order to characterize the flow of water in three dimensions, I have created a setup capable of viewing three dimension in a water tank. I characterized the lens of a GoPro, which will be used to record the movement of particles. This is a Particle Imaging Velocimetry (PIV) setup. This method involves using rosin, or small

particles with neutral buoyancy, that will be illuminated by laser light in a tank of water. [3]

Since the GoPro is not designed to be used for PIV off the shelf, I designed the setup myself. I first characterized the lens. The camera features a fisheye style lens, so equations must be made to turn pixel coordinates from images into coordinates in real distances (meters). Additionally, I found the derivative of the equation so that I could calculate the distance between particles. I produced a tank capable of viewing three dimension from a single camera. I used 2 orthogonal views of the tank provided by a mirror to create a view that is split between the overhead (giving x and y position) and side (giving z) views.

Chapter 2

Laminar versus Turbulent Flow

The oscillating blade of the multi-tool induces two type of flow in the the water around it, turbulent and laminar flow. Laminar flow is the smooth sliding of parallel layers of water over each other. Laminar flow could result in waves that propagate out from the blade and increase the coherence length, or the distance a coherent wave propagates. These wave would move the flexible algae, pushing it away from the the moving blade, and preventing the blade from cutting it.

Turbulent flow, on the other hand, causes chaotic, unpredictable motion in the water. Turbulent flow can be predicted by a high Reynolds number, typically over 1000, a dimensionless quantity that is the ratio of internal forces to viscous forces in the fluid:

$$Re = \frac{\rho u L}{\mu} \tag{2.1}$$

Where ρ is the density, u is the velocity of the fluid, L is the characteristic length, and μ is the dynamic viscosity. The characteristic length is a one-dimensional length value that is based on the size and shape of the object moving through the fluid.

While the Reynolds number at which flow becomes turbulent is dependent on a number of factors, a higher number correlates with more turbulence. In order to increase the Reynolds number, the velocity of the fluid (or of the object moving through the fluid) or the characteristic length must increase. The density and dynamic

viscosity are constant for a given fluid.

In the case of the multi-tool in water, the the density and dynamic viscosity are that of water. The velocity would be the velocity of the blade (around 1m/s) and the characteristic length would be the length of the blade tooth (around 1mm). The current estimate for the Reynold's number based on these values is around 10000. When the blade is tested, it can be verified if it is inducing turbulent flow.

Chapter 3

Characterizing the GoPro Lens

3.1 Theory

The GoPro Hero 3 features an aspherical fisheye-style lens. Therefore, unlike a standard lens, the image produces is distorted further from the center in order to capture a wider view. In order to characterize the lens, I created an equation that relates the pixel coordinates to real coordinates. I assumed that the real radius length is a function of the radius length in pixels R_p, and z, the distance from the lens to the graph paper plane. Note that I also assumed that the lens azimuthally symmetric.

$$R_l = f(R_p, z) \tag{3.1}$$

Where R_p is simply the distance from the center of the image to a given point in pixel units, i.e. $R_p = \sqrt{x_p^2 + y_p^2}$.

I then converted the polar function to Cartesian:

$$x_l = \cos(\phi) \cdot R_l = \frac{x_p}{R_p} \cdot f(R_p, z) \tag{3.2}$$

$$y_l = \sin(\phi) \cdot R_l = \frac{y_p}{R_p} \cdot f(R_p, z) \tag{3.3}$$

Where I assumed that the real angle is equal to the angle shown in the picture, i.e. $\cos(\phi) = \frac{x_l}{R_l} = \frac{x_p}{R_p}$ and similarly for the $\sin(\phi)$. See figure 3.1. Using the data I collected, I can determine the form of $f(R_p, z)$.

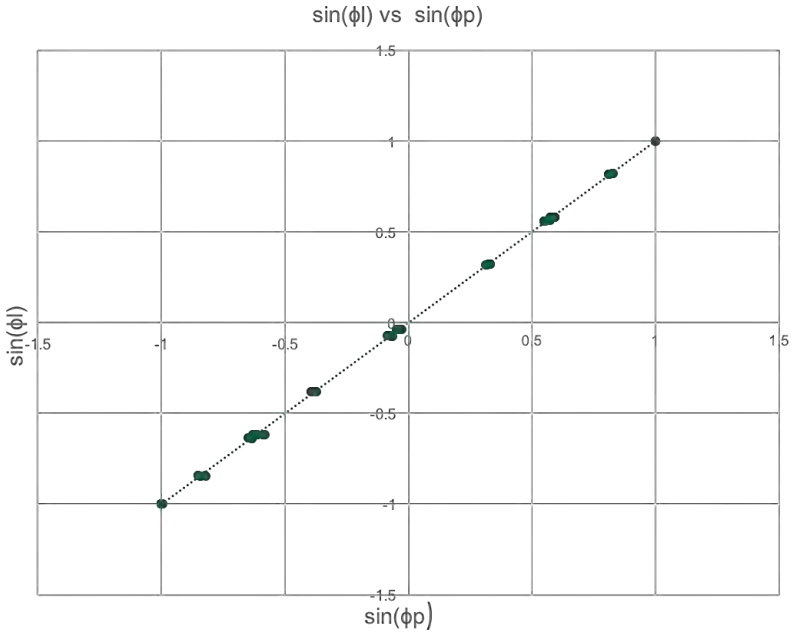


Figure: 3.1: A graph of $\sin\phi$ measured on the graph paper (x_l/R_l) vs $\sin\phi$ measured in the image. Note that the linear fit has a slope of 1, so the value for the angles from the center are the same

3.2 Calibration

In order to create the mapping equations for x_l and y_l, I first had to collect data. I created a setup with the camera mounted so that the lens was aligned with the center of a piece of graph paper about 10cm away (see figure 3.2). I took a still image of the graph paper. Using Matlab, I loaded the image and selected 45 points on the image at regular intervals across the graph paper, recording the real coordinates given (via the graph paper) and the pixel coordinates (i.e. the distance in pixels relative to the center of the image). See figure A.1 in the appendix for an example of one of the photos.

Once I had points from the photo, I graphed the points in various ways. Using a graph of R_l (the real length measured on the graph paper in centimeters from the center) versus R_p (the pixel length measured in pixels from the center of the image),

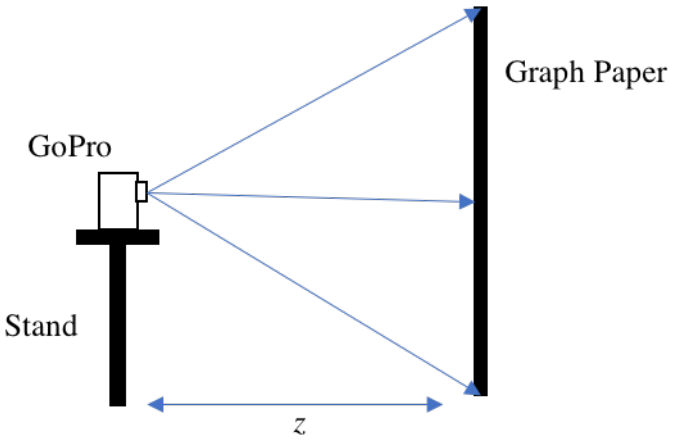


Figure: 3.2: A side view of the calibration setup. The graph paper is attached to a mount that slides back and forth in the z direction. The camera lens is aligned with the center of the graph paper

I found that there is a slight but significant quadratic trend especially towards the center and edges of the image (see figure 3.3. For the data shown in figure, the quadratic term fixes an error of approximately 0.06cm (6%) at the far edges of the image.

Therefore, the equations for R_l, x_l, and y_l are:

$$R_l = a \cdot R_p^2 + b \cdot R_p + c \tag{3.4}$$

$$x_l = \cos\phi \cdot (a \cdot R_p^2 + b \cdot R_p + c) \tag{3.5}$$

$$y_l = \sin\phi \cdot (a \cdot R_p^2 + b \cdot R_p + c) \tag{3.6}$$

Where $\cos\phi = \frac{x_p}{R_p}$ $\sin\phi \frac{y_p}{R_p}$.

Next, I wanted to find how z changed the equation. I hypothesized that the coefficients a, b, and c, were dependent on z. The variable z is the distance from the lens to the center of the graph paper (see figure 3.2). In order to test this, I took five more photos of the graph paper while varying the z value. the graph paper was mounted to a translational stage controlled by micrometers, so I simply adjusted the dial to get several different values for z.

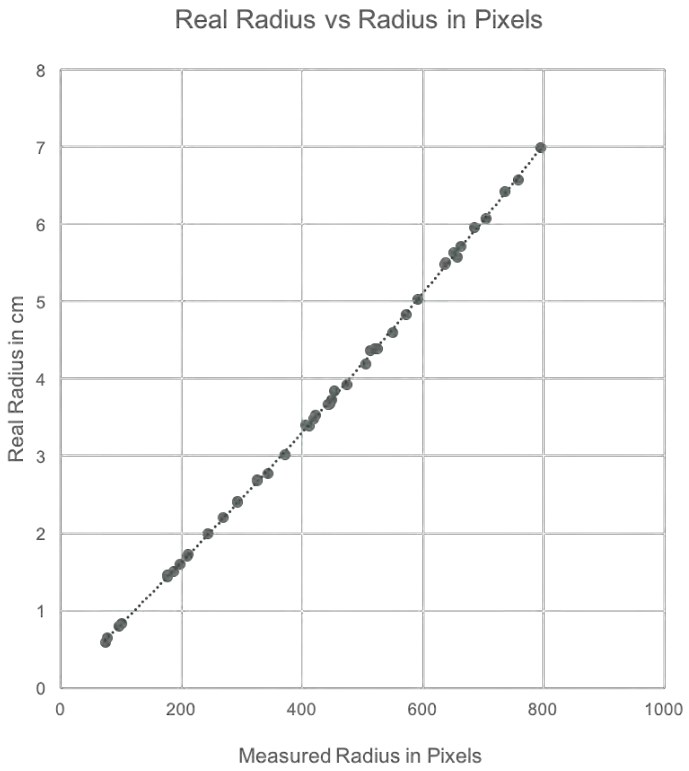


Figure: 3.3: A plot showing the distance of points on the graph paper from the center in centimeters vs their distance from the center of the image in pixels. Also shown is the quadratic fit. While the quadratic fit resembles a linear fit, the quadratic curve fixes an error of 0.06cm at the edge of the image.

For each photo, I took a total of 30 points spaced out evenly across the axes of the graph paper, recording both the real and pixel distances. For each set of points, I made a quadratic fit of R_l from R_p using equation 3.4. To fit, I used a least squares fit varying the value of the coefficients to minimize error. With the coefficients determined for each photo (each value of z), I plotted the coefficients vs z. See figure B.1 in the appendix for a table of the z values for each photo, the respective fit coefficients, and the RMS of the error for the points. The error of the points was the difference between the actual distance measured on the graph paper to the point and the predicted distance using the equation. See figures A.2, A.3, and A.4 in the appendix for graphs of the coefficients vs z.

Using linear fits of the coefficient values vs z, I was able to make equations that give the value of the coefficient for different values of z (i.e. at different distances from the lens).

$$a = 2.98 \cdot 10^{-7} z - 7.24 \cdot 10^{-7} \tag{3.7}$$

$$b = 4.36 \cdot 10^{-4} z - 1.59 \cdot 10^{-3} \tag{3.8}$$

$$c = 0.0141 z + 0.4377 \tag{3.9}$$

These equations can be used to find the correct coefficients for equation 3.4 at a given value of z.

3.3 Derivative of the Characterization Equation

When conducting PIV, the distance between particles (or rather, the relative position at different times) affects the accuracy rather than the the absolute distance. Therefore, the errors that will matter most are the errors on the derivative of the characterization equation, i.e. the errors on calculating the distance between two

points. The derivative with respect to R_p is:

$$\frac{\partial R_l}{\partial R_p} = \frac{\partial f(R_p, z)}{\partial R_p} \tag{3.10}$$

I can then calculate the distance between points using:

$$\Delta R_l = \frac{\partial f(R_p, z)}{\partial R_p} \cdot \Delta R_p \tag{3.11}$$

In Cartesian coordinates, the set of equations is:

$$\Delta x_l = \frac{\partial f(R_p, z)}{\partial R_p} \cdot \Delta x_p \tag{3.12}$$

$$\Delta y_l = \frac{\partial f(R_p, z)}{\partial R_p} \cdot \Delta y_p \tag{3.13}$$

3.4 Derivative Calibration

Since the form of R_l vs R_p is quadratic, the derivative is a linear equation of the form:

$$\frac{\partial R_l}{\partial R_p} = 2 \cdot a \cdot R_p + b \tag{3.14}$$

In Cartesian coordinates the equation for change in location becomes:

$$\Delta x_l = (2 \cdot a \cdot R_p + b) \cdot \Delta x_p \tag{3.15}$$

$$\Delta y_l = (2 \cdot a \cdot R_p + b) \cdot \Delta y_p \tag{3.16}$$

I verified the equations using a new set of images. I used a platform with capable of moving in 3-dimensions (see figure 3.4) with a needle attached. Since the camera overlooks a mirror at a 45 degree angle, the needle can be seen from both overhead and from a front view provided by the mirror (see figure A.5 in the appendix). These view are orthogonal, so 3 dimensional motion can be captured. A breakdown of how different movements are shown in the image is provided in table 3.1. It is important to note that a change in y_l may not necessarily be shown as a change in y_p. In the

Motion Relative to Camera Lens	Motion Relative to Figure 3.4	Motion Shown in Overhead Image	Motion Shown in Mirror Image
Left/Right (x_l)	In/Out of Page	+/- x	+/- x
Up/Down (y_l)	Right/Left on Page	+/- y	Not Visible
To/From (z_l)	Up/Down on Page	Not Visible	-/+ y

Table 3.1: A summary of how movement of the needle in the lab appears as movements in the image. The axis given in the left column can be taken as the axis relative to the lab.

mirror view, a change in $+y_p$ indicates a change in $-z_l$. A change in x_p corresponds to a change in x_l for both views. It is also important to note that x_p is measured left (-) to right (+) in the image in pixels, while y_p is measure bottom (-) to top (+) of the image in pixels.

I then took photos of a needle attached to the platform while moving it in small increments in several directions. I began by taking a photo of the needle in an initial position which I used as my reference position. I then took 9 more photos each corresponding to a movement in the x, y, or z direction (relative to lab) and recorded the the real change in position in centimeters. I then loaded the images in MatLab and took pixel coordinates for the overhead and mirror views in the image.

I measured the change in position in pixels for each movement, giving a Δx_p, a Δy_p, and a Δh_p (with Δh_p as the length of the line in pixels connecting the two

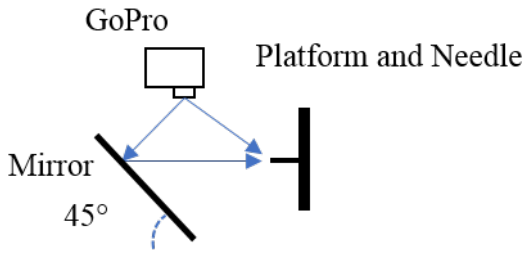


Figure: 3.4: A side view of the 3D calibration setup. The graph paper is attached to a mount that moves in x, y, and z directions. The camera is aligned in between the needle and mirror to provide a view of both. The mirror allows a view showing up and down movement

Coefficient	z (cm)	a (cm/pixel^2)	b (cm/pixel)	c (cm)
Mirror View	15.8	3.98E-06	5.29E-03	6.60E-01
Overhead View	11.1	2.58E-06	3.24E-03	5.94E-01

Table 3.2: The coefficients for the mapping equation (equation 3.4) for the overhead and mirror views. Coefficients are calculated using equations 3.7, 3.8, and 3.9.

points, i.e. $\sqrt{\Delta x_p^2 + \Delta y_p^2}$).In order to verify my equation for the derivative, I divided the distance the point moved on the platform and the total number of pixels it changed. In other words, for each point I calculated $\frac{\Delta h_l}{\Delta h_p}$ (with $\Delta h_l = \sqrt{\Delta x_l^2 + \Delta y_l^2}$. These points result in what is essentially the size of the distance each pixel covers in cm/pixel. These values are plotted in figure 3.5 vs the value of R_p where they were measured. Note that the overhead view and the mirror view have different coefficients leading to different slopes and intercepts. This is because the views have different effective distances from the lens. While the overhead view is 11.1cm away, the mirror view is 15.8cm away. In order to find the z distance for the mirror, I simply added the value of z for the overhead calculation to the horizontal distance from the mirror to the point. See figure A.6 for a diagram of the geometry. Technically, the value of z_mirror is $z_overhead + \tan\theta \cdot d$ with θ being the angle of the mirror to the horizontal (y) axis and d is the horizontal distance from the point to the mirror. In the overhead view, x_p and y_p correspond to movements in the x and y directions while in the mirror view, x_p and y_p correspond to movements in the x and z directions.

Since I wanted to test the validity of my equation for the derivative (as well as the validity of my equations for the coefficients), I compared the measured data to the prediction from my equation. I calculated the value of the coefficients using their respective values of z and equations 3.7, 3.8, and 3.9 (see table 3.2). I included plots of equation 3.14 for each value of z in figure 3.5. Note that the predicted line closely fits the data.

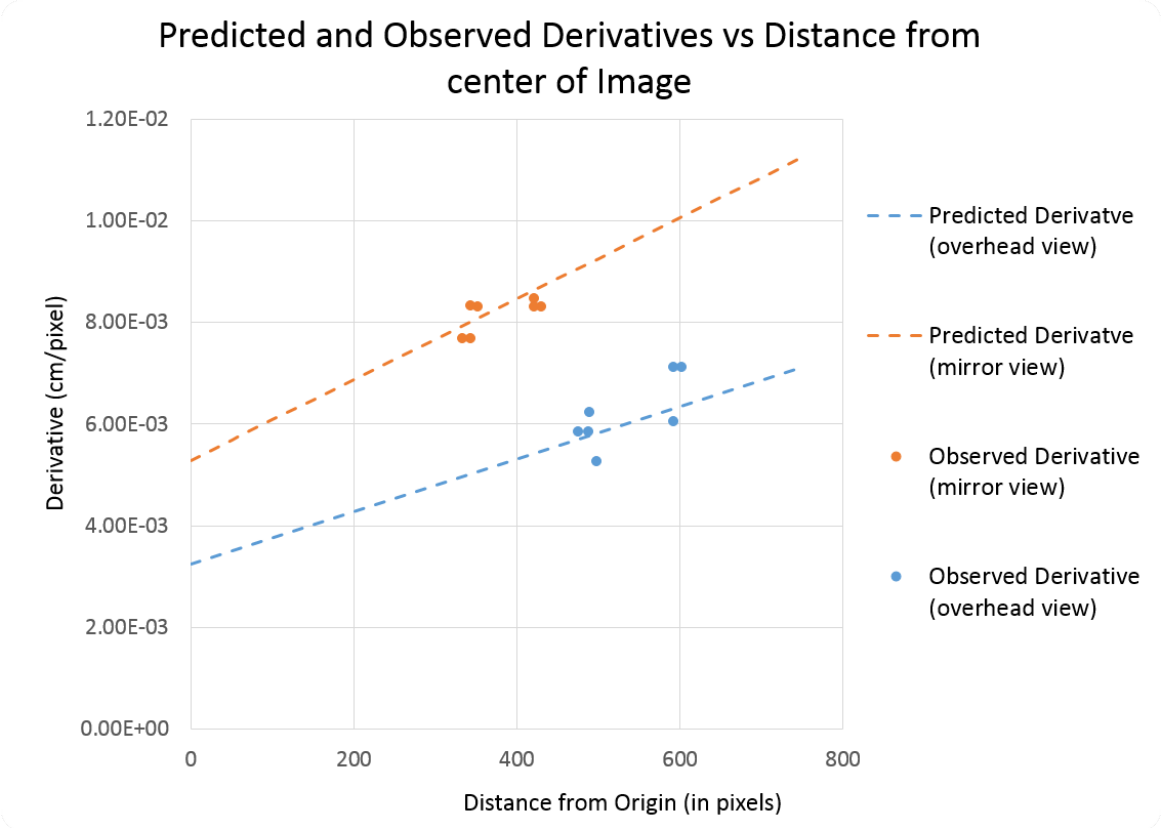


Figure: 3.5: he plot of the derivative for both the top view and mirror view in centimeters per pixel. The points show the observed derivative, i.e. the fraction of the real movement in centimeters divided by the observed movement in pixels in the image. The line shows the predicted derivative using equation 3.14.

Photo	x (pixels)	y (pixels)	Δhl (cm)	Δhp (pixels)	Observed Derivative (cm/pixel)	Predicted Derivative (cm/pixel)	Error (cm/pixel)	% Error
Mirror View								
1	210	-274						
2	210	-274	0.0	0.00	n/a	8.03E-03	n/a	n/a
3	209	-273	0.0	1.41	n/a	8.02E-03	n/a	n/a
4	221	-274	0.1	12.04	8.30E-03	8.09E-03	2.19E-04	3%
5	208	-274	0.1	13.00	7.69E-03	8.02E-03	-3.30E-04	-4%
6	208	-261	0.1	13.00	7.69E-03	7.94E-03	-2.48E-04	-3%
7	208	-273	0.1	12.00	8.33E-03	8.02E-03	3.18E-04	4%
8	326	-268	1.0	118.11	8.47E-03	8.64E-03	-1.75E-04	-2%
9	338	-267	0.1	12.04	8.30E-03	8.71E-03	-4.07E-04	-5%
10	326	-268	0.1	12.04	8.30E-03	8.64E-03	-3.38E-04	-4%
Overhead View								
1	257	418						
2	255	401	0.1	17.12	5.84E-03	5.69E-03	1.48E-04	3%
3	253	418	0.1	17.12	5.84E-03	5.76E-03	7.86E-05	1%
4	272	418	0.1	19.00	5.26E-03	5.82E-03	-5.52E-04	-9%
5	256	417	0.1	16.03	6.24E-03	5.77E-03	4.71E-04	8%
6	254	413	0.0	4.47	n/a	5.74E-03	n/a	n/a
7	254	415	0.0	2.00	n/a	5.75E-03	n/a	n/a
8	419	419	1.0	165.05	6.06E-03	6.30E-03	-2.41E-04	-4%
9	433	418	0.1	14.04	7.12E-03	6.35E-03	7.77E-04	12%
10	419	419	0.1	14.04	7.12E-03	6.30E-03	8.25E-04	13%

Table 3.3: Table 3.3 shows the observed position of the needle tip in both views, the observed change in cm, the observed movement in pixels, the calculated derivative (cm/pixel) via equation 3.14, the measured derivative calculated using $\frac{\Delta h_l}{\Delta h_p}$ (i.e. the fraction of the real movement in centimeters divided by the observed movement in pixels in the image), the error and percent error of the derivative.

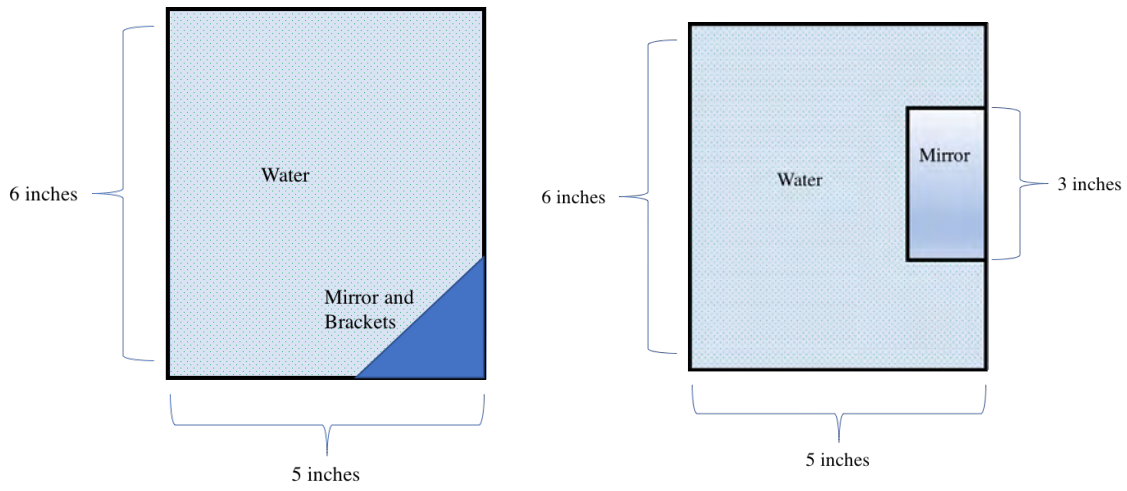
3.5 Errors

I calculated the errors on each of the movements. Table 3.3 shows the observed position of the needle tip in both views, the observed change in cm, the observed movement in pixels, the calculated derivative (cm/pixel) via equation 3.14, the measured derivative calculated using $\frac{\Delta h_l}{\Delta h_p}$ (cm/pixel), the error and percent error of the derivative. Note that some values are shown as n/a. This occurred when calculated the derivative for a movement that is not visible in that view, i.e. a change in z in the overhead view or a change in y in the mirror view.

Chapter 4

Tank Setup

Next, I had the workshop fabricate a water tank. The tank was made with 1/4 inch (0.635cm) clear acrylic sheets bonded with acrylic cement. Clear material was necessary so that particles can be seen from the side (see figure 4.1). I then added triangular brackets to the bottom of the tank that allow me to mount a mirror at a 45 degree angle (see figure 4.2). The mirror creates a side view of the tank is visible from overhead (see figure 4.3). The GoPro is mounted at the top of the tank facing down, slightly submerged in the water. Because of the mirror, the view will be split between overhead (allowing calculation of the x and y coordinates) and a side view (allowing calculation of the z coordinate).



(a) Side view of the tank

(b) Overhead view of tank

Figure: 4.1: Overhead and side views of the tank with dimensions. The mirror is mounted in the bottom of the tank at a 45 degree angle.

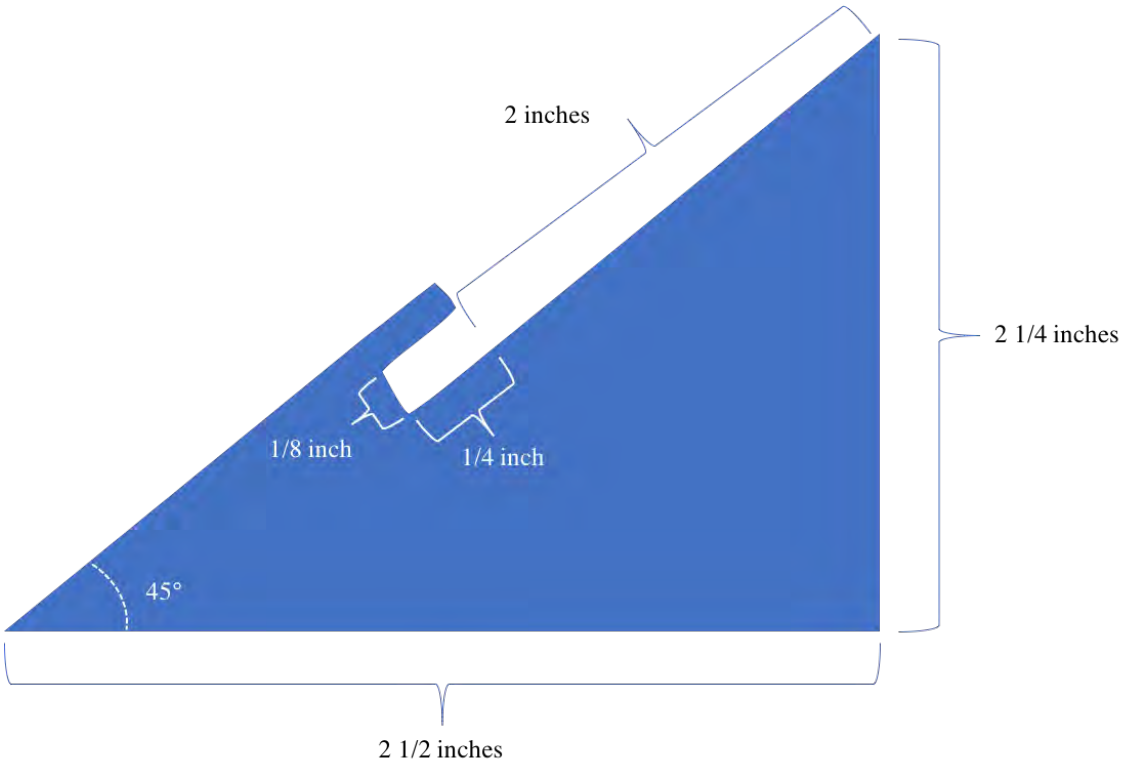


Figure: 4.2: A side view of one of the two brackets used to hold the mirror. Each bracket is made of 1/4 inch acrylic. the 1/8 inch thick mirror slides into the slot and is held tightly

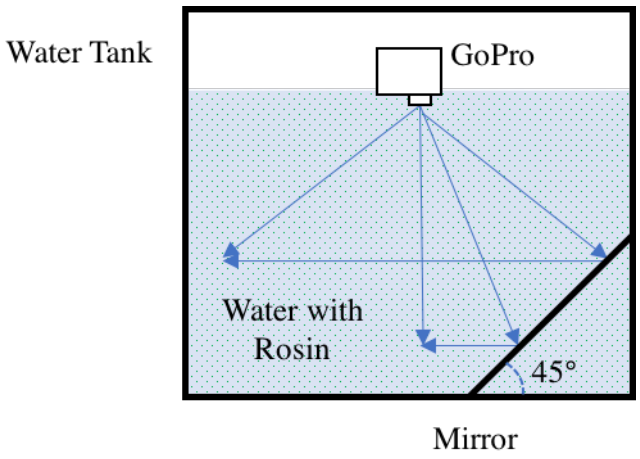


Figure: 4.3: A side view of the water tank with rosin (represented as green dots) the GoPro, and the mirror. Ray lines from the camera show how 3D coordinates can be measured using a single camera and mirror

Chapter 5

Conclusion

The errors on the points in the are reasonable. For the mirror view, all are below 5%. For the overhead view, they are not as good, however,they are all below 14%. One of the main setbacks in reducing the error is from the wide angle fish eye lens of the GoPro. As objects come closer to the lens, they become increasingly blurry and out of focus, making it difficult to identify a single point and increasing uncertainty. Additionally, as object move further away, the coverage of a single pixel increases which also increases the uncertainty. Through trial and error, I found that distance of 11cm is a sweet-spot: it is the closest an object can be while still in focus. However, a higher distance may increase accuracy.

Continued work on this project will focus on the coding of a video processing program and testing different blade shapes.

Appendices

Appendix A

Figures

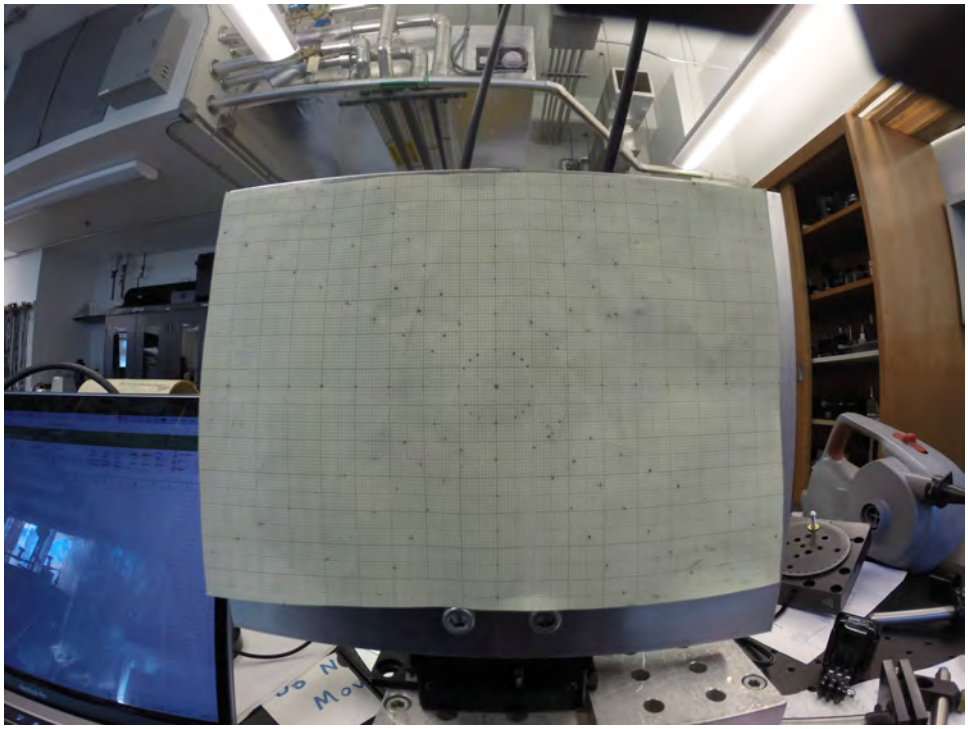


Figure: A.1: An example of one of the photos used to characterize the lens. Small lines are spaced 0.1cm apart, making it easy to determine real distances from the photo.

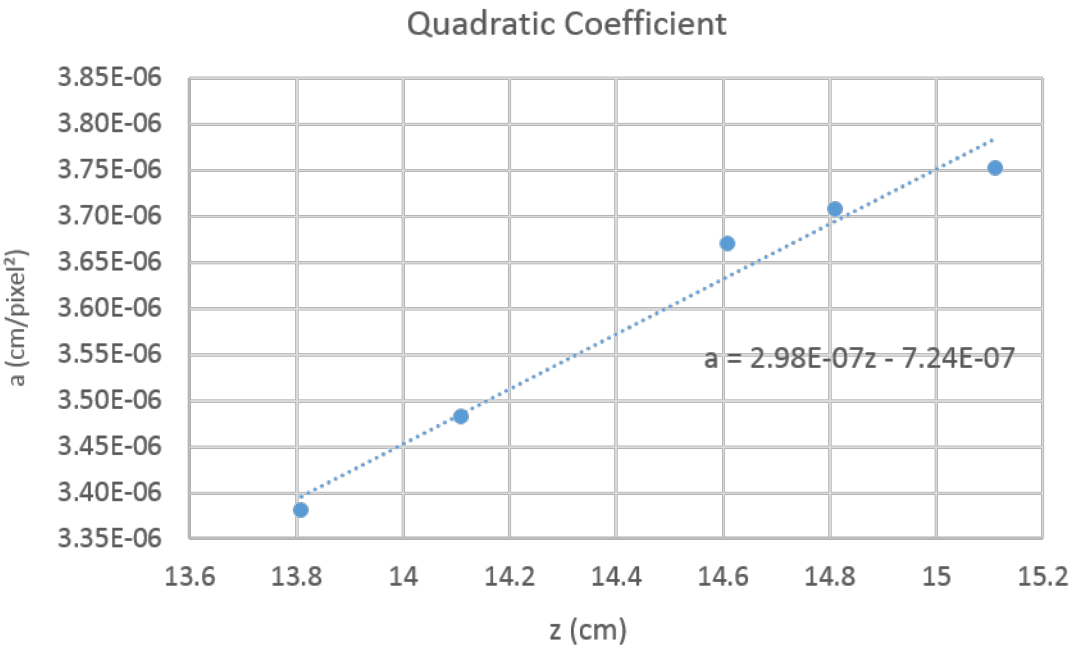


Figure: A.2: This plot shows the value of several a coefficients (the quadratic term) at different values of z. Also plotted is a linear fit of the points and its equation.

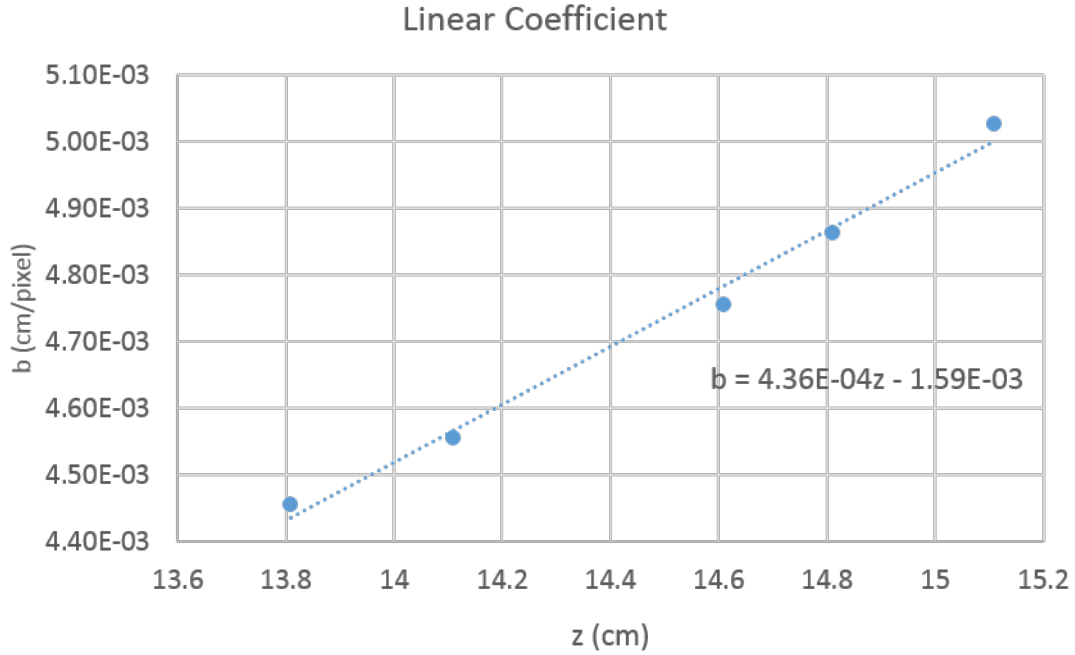


Figure: A.3: This plot shows the value of several b coefficients (the linear term) at different values of z. Also plotted is a linear fit of the points and its equation.

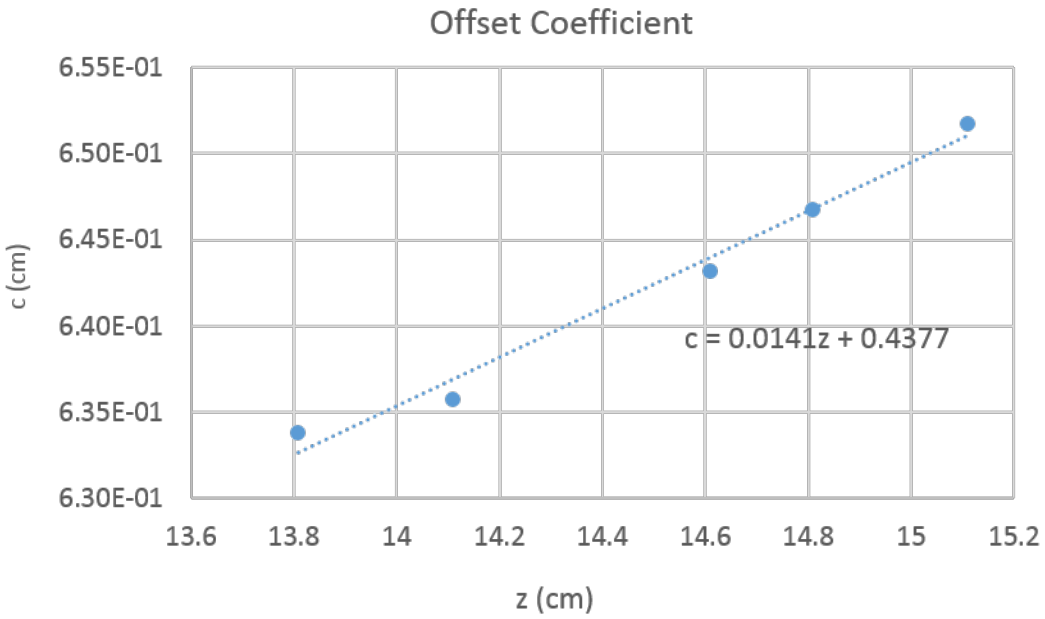


Figure: A.4: This plot shows the value of several c coefficients (the offset term) at different values of z. Also plotted is a linear fit of the points and its equation.

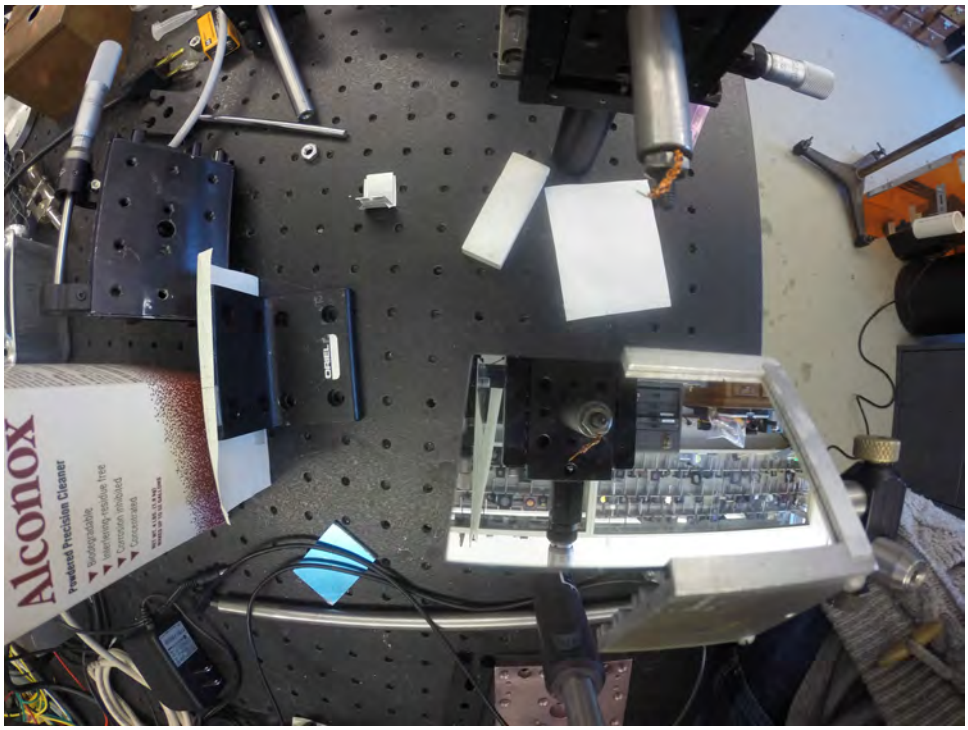


Figure: A.5: An example of one of the photos used to test small movements. Note that the brown needle can be seen both from overhead and in the mirror in the bottom of the image.

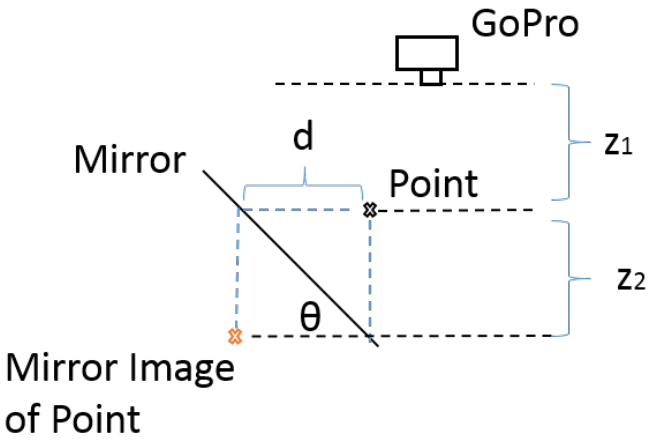


Figure: A.6: A diagram of the mirror geometry. The black x represents the point of interest while the orange x represents the effective location of the mirror image. If θ is 45 degrees, the d and z_2 are equal. If not, $z_2 = \tan\theta \cdot d$.

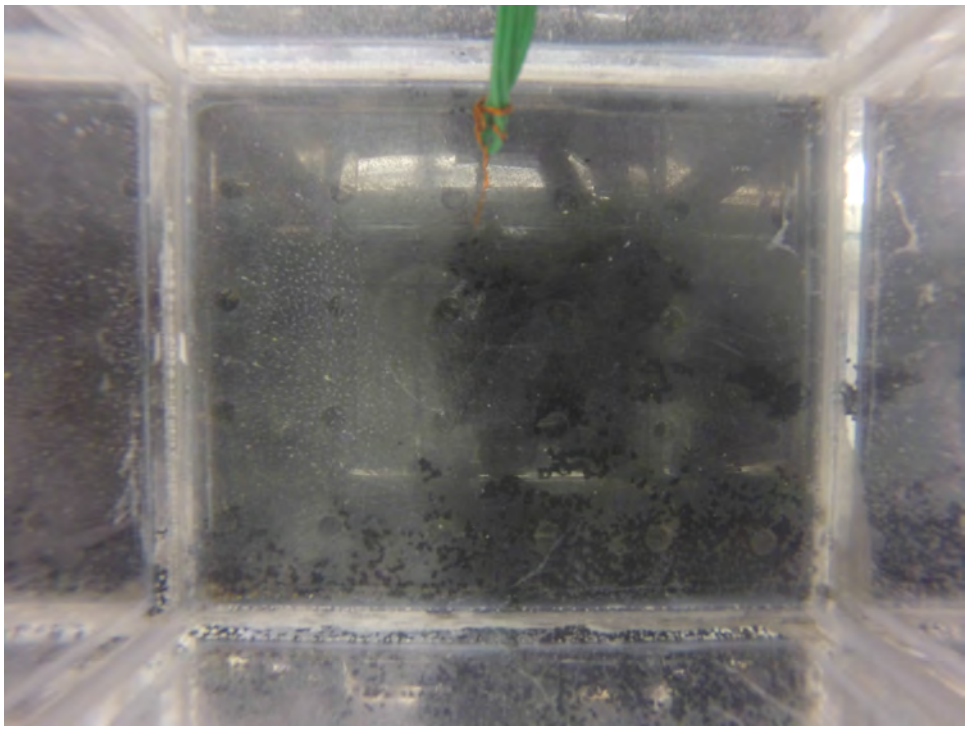


Figure: A.7: An example of the view inside the tank with water. The mirror has not been mounted in this photo, however a needle attached to a wire is shown.

Appendix B

Tables

z (cm)	*a (cm/pixel²)*	*b (cm/pixel)*	*c (cm)*	*RMS (cm/point)*
15.109	3.75E-06	5.03E-03	6.52E-01	1.41E-01
14.809	3.71E-06	4.86E-03	6.47E-01	1.44E-01
14.609	3.67E-06	4.76E-03	6.43E-01	1.41E-01
14.109	3.48E-06	4.56E-03	6.36E-01	1.39E-01
13.809	3.38E-06	4.46E-03	6.34E-01	1.43E-01

Table B.1: The table above shows the values of the coefficients for equation 3.4 at various values of z. It also shows the RMS of the error on each of the points used to find these values.

Bibliography

[1] Cooke, William E. and Small, Katherine D., Algal Aqualculture for Nutrient Assimilation and Removal, Center for Innovative Technology, 2015.

[2] Virginia. Department of Environmental Quality. *Virginia's Watershed Implementation Plans for the Chesapeake Bay*, 2010. http://www.deq.virginia.gov/Programs/Water/ChesapeakeBay/ChesapeakeBayTMDL/ChesapeakeBayWatershedImplementationPlans.aspx

[3] Prasad, Ajay."Particle image velocimetry." *Current Science*, 79, no. 1, 2000, pp. 51-60.