THE COMPACT OBJECT AND STARFIELD SIMULATOR (COSS)

An Undergraduate Research Scholars Thesis
by
PETER MACK GRUBB
JOHN MAYO

Submitted to Honors and Undergraduate Research
Texas A&M University
In partial fulfillment of the requirements for the designation as

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Research Advisor: Dr. Tom Pollock

May 2014

Major: Computer Engineering – EE Track

May 2014

Major: Mechanical Engineering

TABLE OF CONTENTS

ABSTRACT

The Compact Object and Starfield Simulators. (May 2013)

Peter Mack Grubb
Department of Electrical and Computer Engineering
Texas A&M University

John Mayo
Department of Mechanical Engineering
Texas A&M University

As objects continue to be placed into space, Earth orbiting debris are becoming a prominent issue. To counteract this, NASA's Marshall Space Flight Center's (MSFC) Engineering Directorate has funded a project to create a test bed for systems designed to track and identify said objects, known as a starfield simulator. This system generates 2D images and displays them for viewing by potential optics being considered for Earth orbiting debris detection systems. One method studied previously for identifying objects is the Fine Scale Optical Range (FiScOR)[1], a method for physically modeling space objects at a distance using 3D printing and then recording light curves; however, this system did not consider physical objects in the context of an actual starfield. Thus the goal of this proposed research is to develop a system which allows for the gathering of FiScOR type physical object light curves in the context of a starfield simulator such as the one funded by NASA's MSFC.

ACKNOWLEDGEMENTS

Thank you to NASA's Marshall Space Flight Center's (MSFC) Engineering Directorate for the funding necessary for a large portion of this project.

Thank you to Dr. Christian Bruccoleri for many hours of code development and mentoring in support of this project.

Thank you to Dr. Chip Hill and the staff at the Space Engineering Research Center for support and logistics keeping this project going.

NOMENCLATURE

CAD	Computer Aided Design
CCD	Charge Coupled Device
CMOS	Complementary metal-oxide-semiconductor
CNC Mill	Computer Numerical Control Mill
COSS	Compact Object and Starfield Simulator
FiScOR	Fine Scale Optical Range
IR	Infrared
Light Curve	A plot of light intensity vs. time
MSFC	Marshall Space Flight Center
NASA	National Aeronautics and Space Administration
Starfield Simulator	System that displays stars that would be seen by a system at a specific location based on orbital and camera parameters

CHAPTER I

INTRODUCTION

The Compact Object and Starfield Simulator has been designed and assembled to test and evaluate satellite grade optics for use in identifying extended space objects. Physically, this facility consists of a computer generated 2D starfield simulator with one to two centimeter models of simple or complex shapes placed in front of the projection screen. This allows for the simulation of extended space objects, and their relative visibility against the backdrop of stars. The objects were designed in 3D CAD and produced in plastic by 3D printing and CNC milling. Details, such as imperfections in the surfacing of the objects, were achieved to dimensions as fine as 200 micrometers. Mechanisms were provided to rotate the object. Additionally, an illumination source approximating the solar spectrum was used and light curves were recorded using CMOS monochrome and color cameras.

The concept of modeling miniature objects in limited lab space has been proven conceptually accurate through the Fine Scale Optical Range[1]; however, objects are not alone in space and usually have a backdrop of stars. Thus, our proposal was to make a way to view these objects rotating in front of a starfield. This required a way to view the objects and starfield at the same or similar focal distance, thus necessitating a lens to make all objects at a certain distance in focus. A telescope was used with a collimator lens to view the objects and starfield in focus. Data was then recorded and analyzed to determine the effects of the star backdrop on the accuracy of the light curve data gathered.

Background

Currently there is great concern in the Aerospace and Defense communities about the proliferation of orbital debris[2]. A very small piece of debris in orbit can completely destroy a multi-million dollar satellite just by being in the wrong place at the wrong time[3]. When this was confined to large objects that were known to be in orbit, it was of limited concern, as it was very simple to map and calculate the orbits for these objects; however, with the growth of anti-satellite weaponry[4][5], and the general disrepair of satellites that have been in orbit for many years, a phenomenon known as "debris fields" has begun to be a problem. While the general orbit of the field as a whole can be modeled, there are two challenges that make these fields of great concern[6]. Firstly, over time they expand from the event that originally caused the debris field to form. This will allow objects to drop into higher or lower orbits that have large differences in location very quickly. Secondly, there is no established method for tracking these objects. To establish an accurate orbit for an object, observation across multiple orbits is necessary; however, accurately identifying a single dot in a changing field of dots is highly problematic. To this end, a method for identifying objects in some consistent way is desired.

The Space Engineering Research Center has extensive experience in dealing with this problem. One approach has been to work with star trackers, cameras with wide angle lenses and relatively small chips[7]. Another approach used at SERC is to create modeling environments that allow for rapid testing. Examples of this include the Fine Scale Optical Range (FiScOR)[1] and the NASA MSFC funded star simulator used in this project.

One method for identifying objects is by determining their physical properties. Physical properties such as material composition and object shape rarely change from one orbit to the next. Thus, if an object can be quickly identified by its physical properties from one orbit to the next, it becomes distinguishable. However, before methods of quick physical property determination can be developed, accurate light curves for known objects must first be collected.

A “light curve” is a graph of how the intensity of light changes over time. Usually the units used are “ADU’s” or Analog to Digital Units. This is the technical term for the pixel value in an image of a given light source at a given gain on the sensor. These ADU’s are plotted against either time or degrees depending on the specific application of interest. The resulting plot is known as a “light curve” and can be analyzed using typical signals analysis techniques such as the Fourier transform.

Light curves of known objects have multiple sources. One such source is ground based astronomy of known objects. This provides very accurate data, but is limited by the nature of objects in orbit. Additionally, it is somewhat impractical as quickly imaging a wide variety of objects needed for algorithm development is highly expensive. FiSCoR[1] created a method of rapidly imaging objects in order to support accurate theoretical models. However, it did not take into account other empirical factors such as the starfield behind the object. Thus, a method of accurately imaging known objects in an orbit like scenario is needed in order to create accurate light curves for supporting modeling efforts.

To this end, the research team has created the Compact Object and Starfield Simulator (COSS). Using small, physical models in front of a simulated starfield, observed with a telescope through a collimator lens, we have created a method of accurately imaging these objects at sizes analogous to what would be seen in an orbital scenario. This allows for the testing of satellite hardware in actual flight scenarios, and the building of empirically accurate lighting models for objects in orbit. Through COSS, satellite hardware can be verified as being accurate for given object types prior to launch, saving time and money in the event of insufficient optics for a given task.

Preliminary Data

As part of the preliminary efforts to build COSS, data was gathered to compare FiScOR[1] data with the Phong model for a simple white, diffuse object. Once the data was gathered from both FiSCoR and the Phong models, it was analyzed using Matlab. This analysis took two forms: normalized cross correlation constants, and matched graphs. For the cross correlation constants, the Matlab function xcorr() was used; however, for cross correlation to have any meaning, the time step in the data must be equal. Thus, prior to running xcorr, the data was up-sampled using the Matlab function interpolate() until both the Phong model data and the FiScOR data had the same number of samples. Additionally, the data sets were normalized by dividing each element by the integral of the data set, giving an area under the curve of 1 in the resulting set. The results from this are recorded in Tables 1 through 4.

Table 1. 10 deg Horizontal, 16.5 deg Altitude

	Red	Green	Blue	White
Cube 1	0.999976	0.999912	0.999891	0.99995
Cube 2	0.999979	0.999931	0.999907	0.999979
Cube 3	0.999969	0.999828	0.999787	0.999893
Cube 4	0.999979	0.999954	0.999927	0.999963
Octagon	0.999942	0.999906	0.999859	0.999908

Table 2. 45 deg Horizontal, 16.5 deg Altitude

	Red	Green	Blue	White
Cube 1	0.999981	0.999948	0.999947	0.999972
Cube 2	0.999982	0.999938	0.99993	0.999979
Cube 3	0.999979	0.999937	0.999935	0.999971
Cube 4	0.999971	0.999883	0.999848	0.999913
Octagon	0.999967	0.999881	0.999826	0.999904

Table 3. 90 deg Horizontal, 16.5 deg Altitude

	Red	Green	Blue	White
Cube 1	0.999992	0.999989	0.999949	0.999991
Cube 2	0.999989	0.999975	0.999974	0.99999
Cube 3	0.999991	0.999975	0.99996	0.999983
Cube 4	0.999982	0.999981	0.999974	0.999986
Octagon	0.999991	0.999964	0.999971	0.999982

Table 4. 135 deg Horizontal, 16.5 deg Altitude

	Red	Green	Blue	White
Cube 1	0.999993	0.99999	0.999987	0.999991
Cube 2	0.999984	0.999976	0.999968	0.99998
Cube 3	0.999991	0.999971	0.999959	0.999977
Cube 4	0.999976	0.999974	0.99997	0.999976
Octagon	0.999996	0.999996	0.999993	0.999997

The other method used for comparing the data was visual. A Matlab script was created to graph each color against its corresponding Phong model. First, the script took the interpolated data that was input to xcorr() and found the difference in their relative magnitudes. Based on this factor, the Phong model was scaled to match the real data. Then, the means of the two distributions were aligned. This allows the data to be graphically compared, giving a more detailed picture as to what the data means. To show the periodic nature of the data, two full revolutions are pictured in the graphs below. The graphs for a cube rotating about an axis through one edge of the cube are shown in Figures A1-A4 in Appendix A.

Experimental Plan

Between January and August, a system was developed for simulating objects in the far field. The results from this project were published at the Advanced Maui Optical and Space Surveillance Conference in August 2012[1] and the SpaceVision conference in November 2012. This system

accurately modeled objects in the far field, but did so with optics that were unlikely to be used in a space application due to their short focal length. Additionally, the system did nothing to account for the effects of background stars and planets. From this, a motivation grew for a more accurate space object simulation environment. The aims of this system were as follows:

Aim 1: Build a starfield simulation system suitable for use with long focal length optics.

The first step for creating an accurate representation of a space scenario was to create a simulation environment that could quickly display appropriate imagery at a distance within the chosen optic's focal range. Most space applications use telescopes or other long focal length optics which also have minimum focal distances that are 10+ meters in length. This makes creating indoor testing environments difficult.

Aim 2: Add FiScOR style models and light sources to the simulation environment.

Most starfield simulation systems use purely digital models for object surfaces, which, as shown with FiScOR, are not accurate representations of the objects in question. Thus, we wanted to add accurate object simulation to our starfield.

Aim 3: Show that object light curves are equally accurate in the presence of background stars and significant camera noise.

For FiScOR data to have practical space applications, the light curve data needs to be accurate regardless of the background in question and in the presence of large degrees of camera noise. Thus, we wanted to show that object light curves in the presence of starfields are equally accurate to those taken with no background.

The result of successfully completing these three aims was a system which could be used for completely simulating what a given camera would see in orbit. It was more accurate than a traditional starfield simulator as it did not rely on unreliable virtual lighting models for illuminating objects in the scenario.

The most challenging aspect of aim 1, keeping the length of the apparatus short enough to fit indoors while testing long focal length telescopes, was accomplished with a custom made collimator lens. This lens takes in light at different angles and collimates, or aligns it in a parallel direction in order to appear at an infinite focal length to the telescope. This reduces the overall length of the setup to just 10 meters for any telescope, thus not too long to fit in a room. The option of a mirror between the collimator and starfield screen is also viable to shorten the length of the path.

The simulated starfield was projected on a 12 bit screen while extended space objects were modeled using real objects made of satellite materials in order to collect the most accurate data as seen with FiScOR. These objects were rotated by a small electric motor hooked to a National Instruments control system and controlled by a Labview VI. A light source was positioned at a near 90 degree angle to prevent glare on the starfield screen. This way, a real object was completely viewable in front of different starfield backgrounds.

For validating the system, the first step was to perform a distortion analysis using a grid of Gaussian centroids. By analyzing how these centroids are rendered in the camera, the distortion

caused by the collimating lens and other parts can be quantified. This figure needs to be as low as possible in order for the system to represent an accurate depiction of the scenario in question.

With the added background, the data has been compared to the FiScOR model sans background to study whether it alters the light data from the objects themselves. There was considerable concern that the backlighting from the screen would provide significant back illumination which would render the light curves meaningless. By comparing the light curves gathered both with and without a background, it can be shown that the effects of the background are insignificant from a light curve perspective.

Additionally, both monochrome and color cameras were tested on the apparatus, and the gathered light curves compared. As the background is projected on a 12-bit monochrome display, there was some concern that color noise artifacts would become more significant. By comparing the color and monochrome data, it will be shown that the aggregate light curve is consistent independent of the imaging device.

CHAPTER II

METHODS

COSS works off of one of the same fundamental assumptions that makes the FiScOR[1] possible: namely that compared to a wavelength of light, both the small one to two centimeter models and a full size object are extremely large. Thus, from the perspective of the light gathered, the size of the actual object is somewhat unimportant. All that matters is the scaling and composition of the light.

The second major assumption that COSS works from is based off of the resulting backlighting from the screen. Namely, it is assumed that any backlighting from the screen is much smaller than the intensity of the illuminated object. Thus, light curve data collected of the object in front of the screen should be shown to be similar in shape to the same object without the screen behind it. This will be shown later in the paper, as it is not a trivial assumption.

The model is illuminated by a quartz halogen light source with a fiber optic feed designed to mimic solar lighting. As a part of setup, the light source was measured using a USB Stellar-Net® spectrophotometer. The spectrum of the light source is shown in Figure 1.

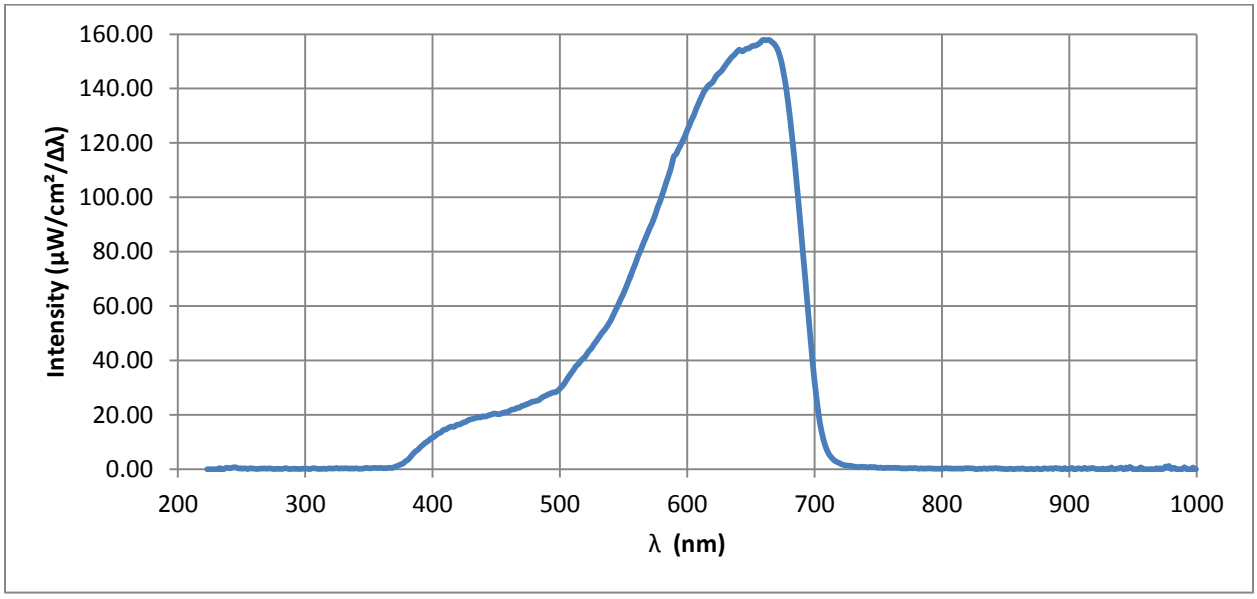

Figure 1. Spectrum of Light Source. This source emits more light in the near infrared than is found in sunlight, even with an IR blocking filter in place.

This light source has a baffle and IR filter to limit unintentional reflections off other lab apparatus and excess photon count respectively. It was placed parallel to the starfield to minimize reflections on the surface of the screen. This placement only lit one half of the objects, similar to the object lighting done with FiScOR. The model itself was mounted on a rotating platform at the height of the starfield midpoint.

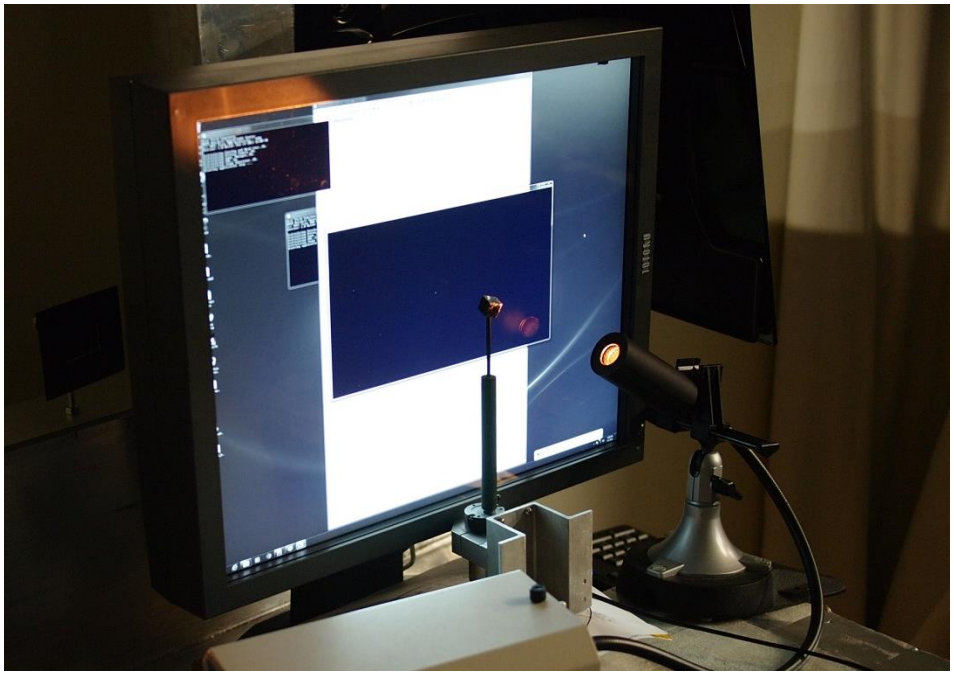

Figure 2. Object with starfield. Note the back half of the object is not illuminated.

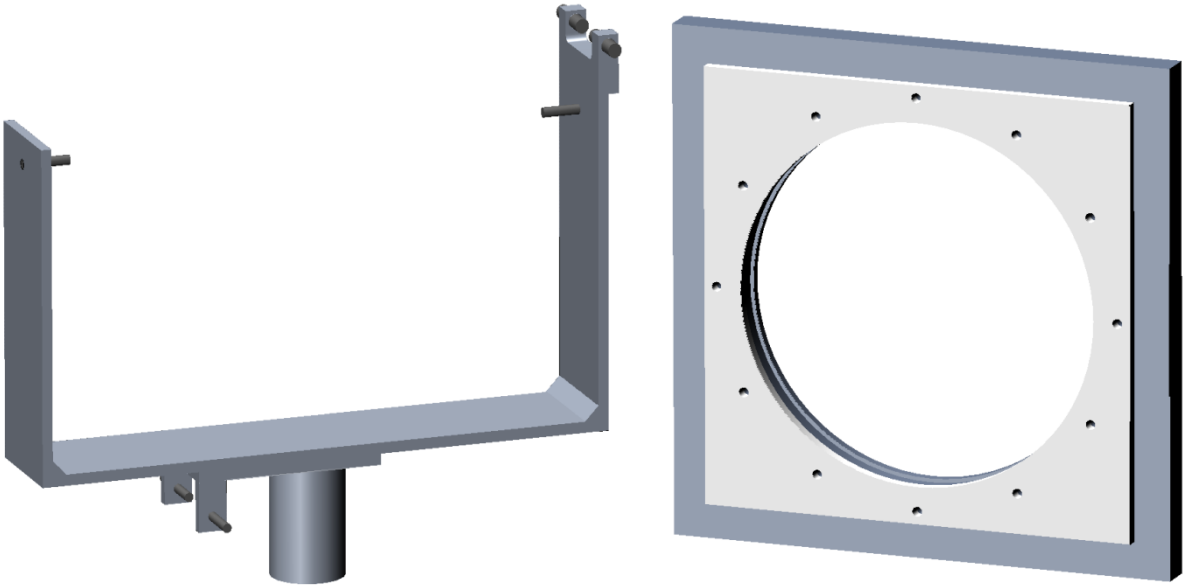

Figure 3. Collimator lens mount. Note the screws for adjusting.

The collimator holder features a carbon-fiber graphite epoxy material to reduce weight and thermal expansion. This ensured that the collimator did not move once set in position, even if the

thermal conditions around it changed, ensuring accuracy in collecting data. Also featured on the mount are threaded inserts to fine-tune the adjustments with a push-pull mechanism.

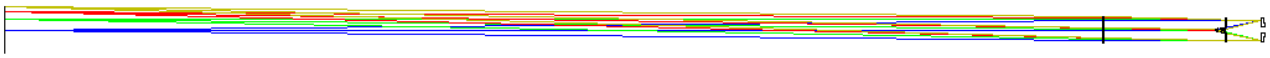

Figure 4. Diagram of entire light system from camera to object via the collimator lens.

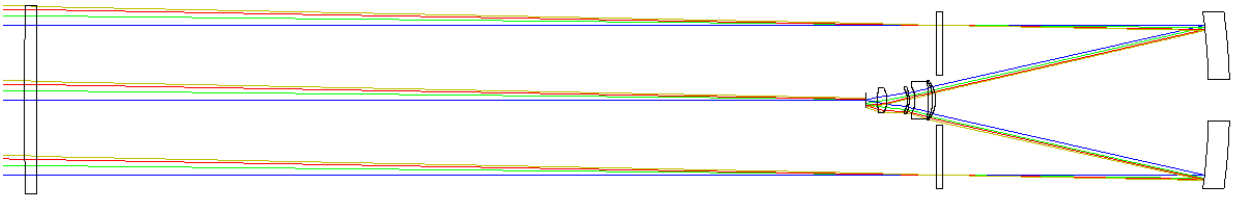

Figure 5. Detail of collimator lens (far left) and telescope with camera system.

The collimator lens takes the incident light into the telescope and refocuses it so that the minimum focal length is much shorter than it would be without. Figure 4 shows the complete system with the refocused light hitting the incident object. Figure 5 shows the detail of the collimator into the telescope and its camera.

The telescope mount features a slider to further fine tune the position along with the focus knob. Furthermore, in the configuration used, the camera mounts on the front of the telescope, thus limiting the physical size of cameras that were used. For this experiment, the screen that displays the starfield along with the spinning model was on a rolling cart. All of these adjustments were necessary in order to achieve the desired focal length of the telescope.

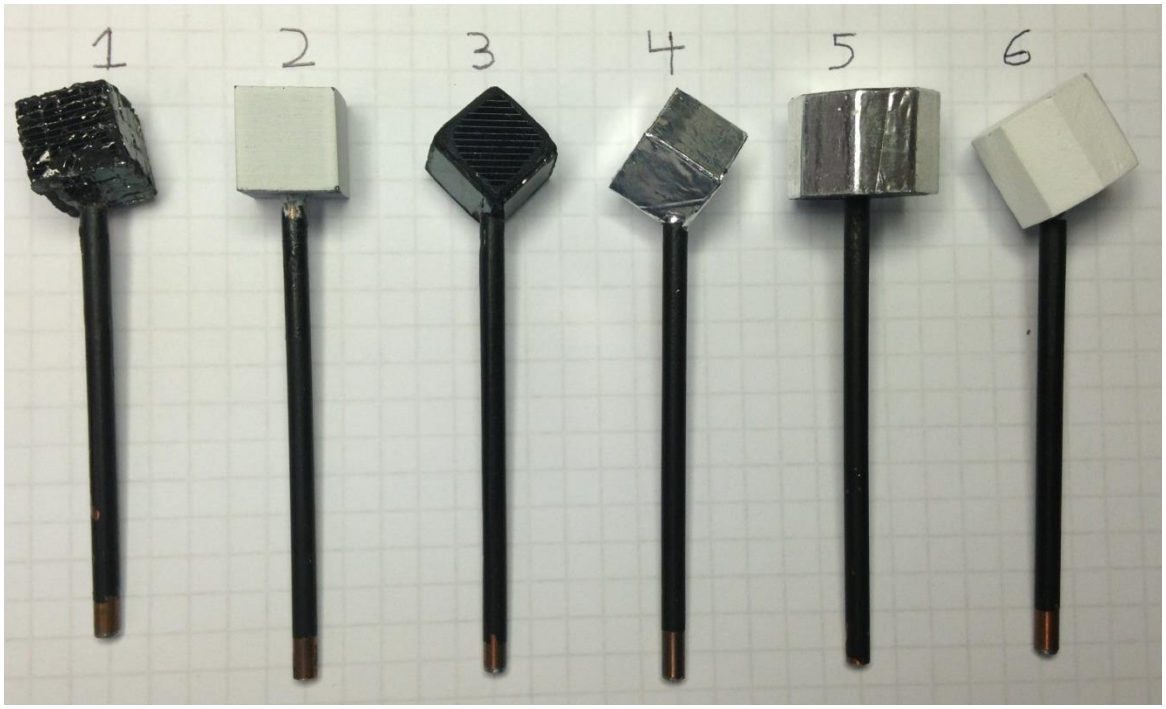

Figure 6. Model materials from left to right: cracked solar panels, white diffuse surface, undamaged solar panels, aluminum tape, aluminum tape, and a white diffuse surface. All the stalks are coated in black spray paint.

The model is kept at a ten centimeter height on a rotating platform that can move with the light. The model itself can be mounted at any axis of rotation, and should is approximately one to two centimeters in diameter.

The models are built using both 3D printers [8][9] and CNC machines. For this experiment they are cubes and octagonal prisms. These models are covered in various real materials, so that the table can be used to measure and compare light curve data from not just shape, but also physical properties. The models can also be spray painted for various effects. Furthermore, a baffle was placed in front of the lower half of light source to prevent stray refractions from the stalk on which the object rested.

This system was setup linearly between two rooms with a window in between. The original idea was to hold the entire apparatus in one room with a mirror in the middle; however, the mirror acquired created too much distortion and could not be utilized. Thus, the system was laid out linearly, pointing through a window between two lab spaces. Tests were done to ensure that the window did not introduce excess distortion. The distance between the collimator lens and object is exactly 32 feet and 3.79 inches, while the distance between the telescope and lens was not as essential. Furthermore, the exact distance between the collimator lens and object varied some due to the lens not being made perfectly to spec, thus utilizing the adjustments on the lens for precise focusing.

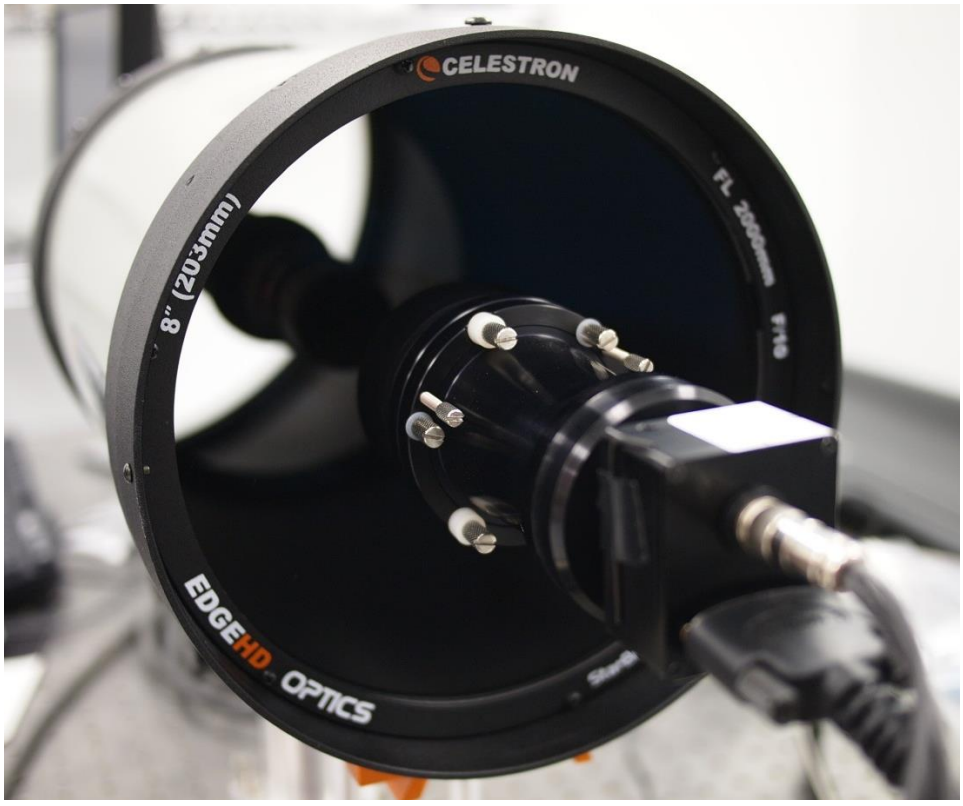

Figure 7. The camera mounted on the front of the telescope

The final component of COSS is the camera itself. Virtually any camera may be used, but accurate light curve determination requires a large number of images per revolution. If a high speed camera is unavailable, this can be achieved via electric motors capable of slow turn rates. In the case of FiScOR, the Silicone Imaging® SI640-HFRGB, pictured in Figure 7, and a Silicone Imaging® SI-1920HDM were used. The first was selected because it is a high frame rate color camera which uses a Bayer mask[11] and a high speed Camera Link interface. The second camera was selected due to its ability to record 12bits of monochrome data per pixel, matching the bit depth of the display, and its use of the Camera Link interface. Additionally, the monochrome camera has a higher effective frame rate, as the color camera is not as sensitive and thus must use a longer integration time.

An EPIX PIXCI® EB1 interface card and the EPIX XCAP® software were used for the purposes of frame grabbing. This allowed for captures on the order of 350 frames per revolution of the models in monochrome, and captures of approximately 100 frames per revolution in color. The color camera had fewer images as it required a much larger integration time to be able to see both the model and the monitor in the background. This is due to the fact that the monochrome camera is much more sensitive than the color camera, allowing for faster integration times. The specific telescope used for this project was a Celestron EdgeHD 2000mm lens with a 203mm aperture.

The reason for the color camera is based out of the success of FiScOR[1], where it was found that some materials such as Solar Cells or the Gold Multi-Layer Insulation (MLI) used in satellites have very distinctive color signatures. A monochrome camera will often lose details

such as this, rendering objects indistinguishable. By using a color camera, objects of the same shape with different real material coverings can be imaged, and their R, G and B data compared separately; however, there was some concern that due to the lower sensitivity of the color camera, essential light curve data will be lost when considered in aggregate against the monochrome camera. Thus, both cameras were tested on the system and compared.

CHAPTER III

RESULTS

The results can be split into two broad categories: distortion analysis and comparative light curve data. The first category is distortion analysis; this was performed by taking a picture of a matrix of Gaussian centroids, and then finding how many pixels each centroid differed from its ideal position. Corrections were performed for consistent axial tilt, as the 12-bit display's stand had a small issue that was not initially corrected when the distortion analysis images were taken.

Once this correction was completed, the centroids in the resulting corrected image were calculated, and compared with the theoretical centroids. The results from this are represented graphically in Figure 8, and shown in tabular form in Appendix B.

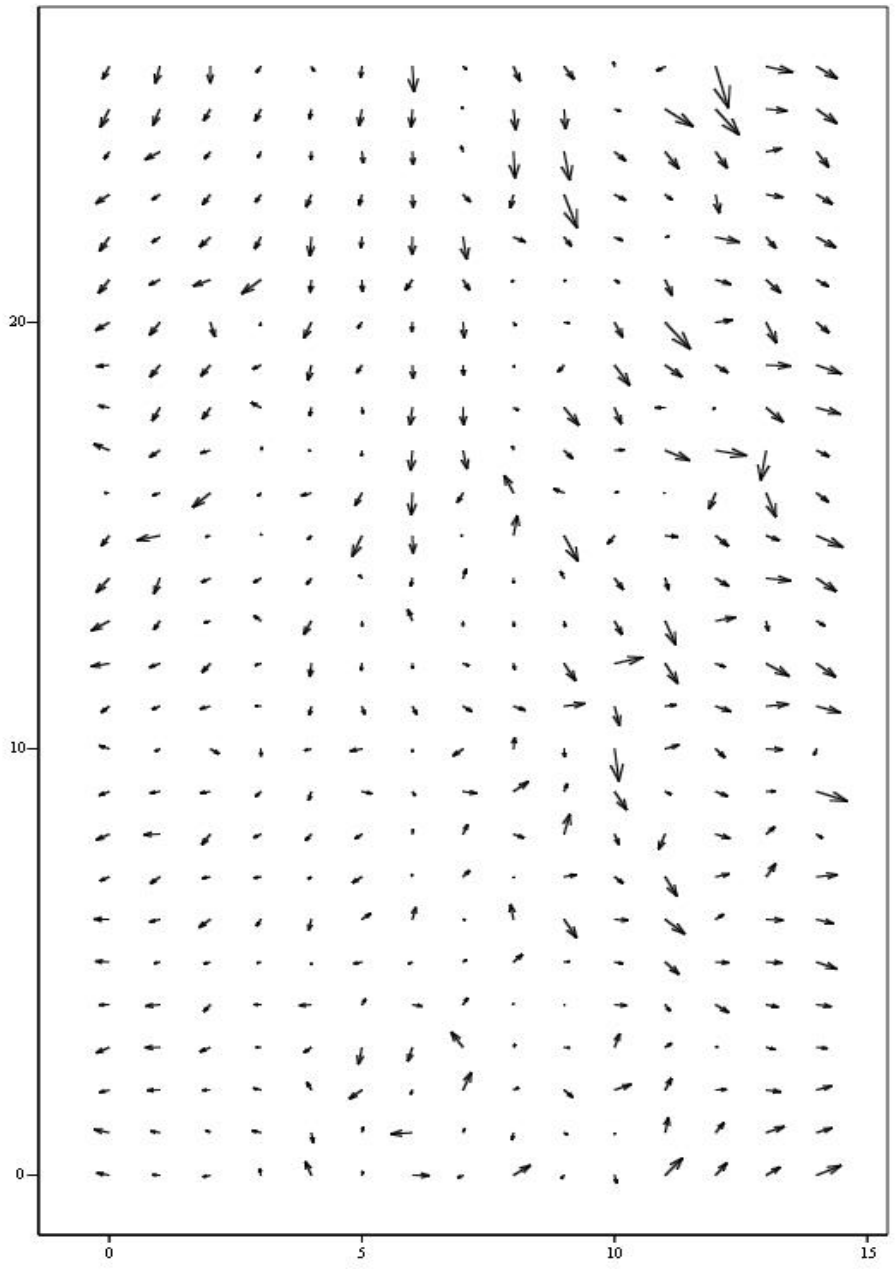

Figure 9. Distortion map for recorded data. The heads of the arrows represent the actual position of the centroid, while the tails are the actual position. Note that the difference is greatly exaggerated in this plot in order to make it more readable. Actual values are less than 0.6 pixels at maximum. See Appendix B for actual values.

The second category of data is comparative light curves. This data was gathered by taking large image sets of the cubes over a full revolution of the object. This data was taken at various frame rates depending on the capabilities of the cameras, and the integration time with the best signal to

noise ratio was used for these graphs. Once this data was gathered, several Matlab scripts were written for reducing the data to 2D plots.

Each of these scripts summed the pixel values in the image, giving a single data point for the light intensity in each image. Once the data reduction process was completed, two different types of graphs were created. The first was a simple three color light curve for models with and without background. These plots are shown below, with Figures 9, 11, and 13 having no star background, and Figures 10, 12 and 14 with one.

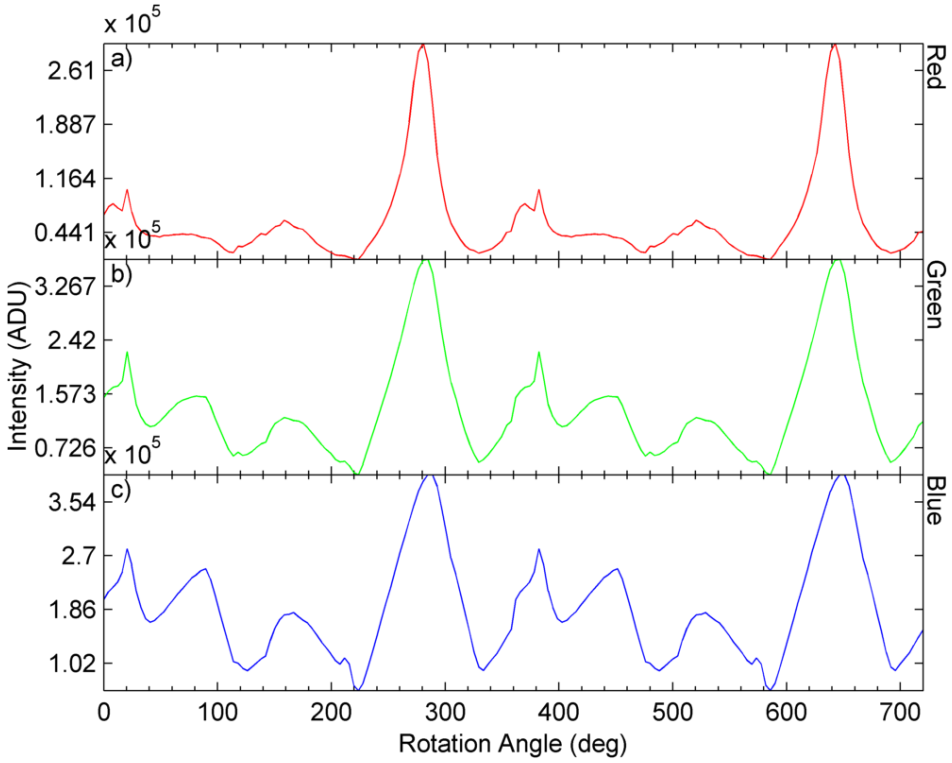

Figure 10. Model 3, a cube covered in solar cell panels with an axis of rotation through one edge of the cube. No star background.

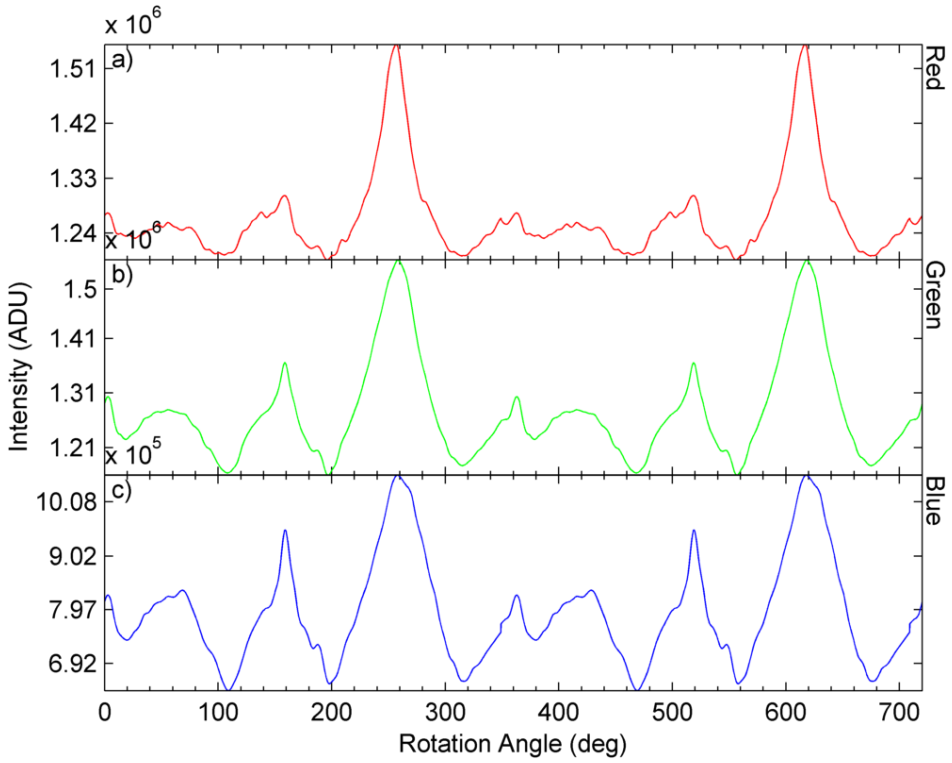

Figure 11. Model 3 again, this time with a star background. Notice that the values are all one order of magnitude higher, and there is more noise with the background.

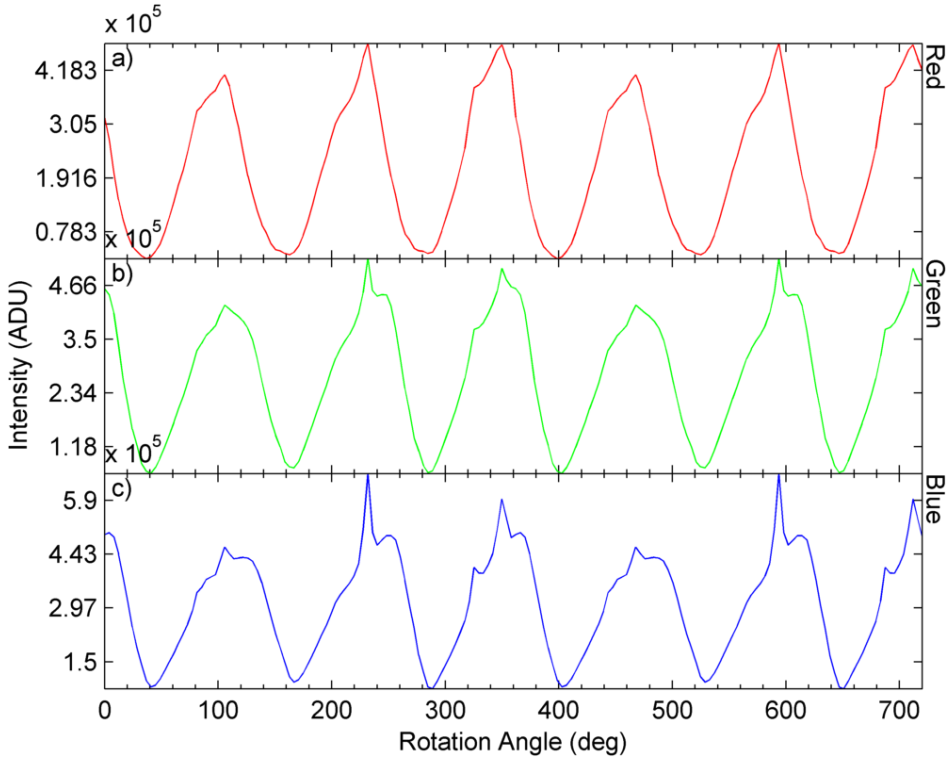

Figure 12. Model 4, a cube covered in aluminum tape with the axis of rotation through one corner of the cube. No star background.

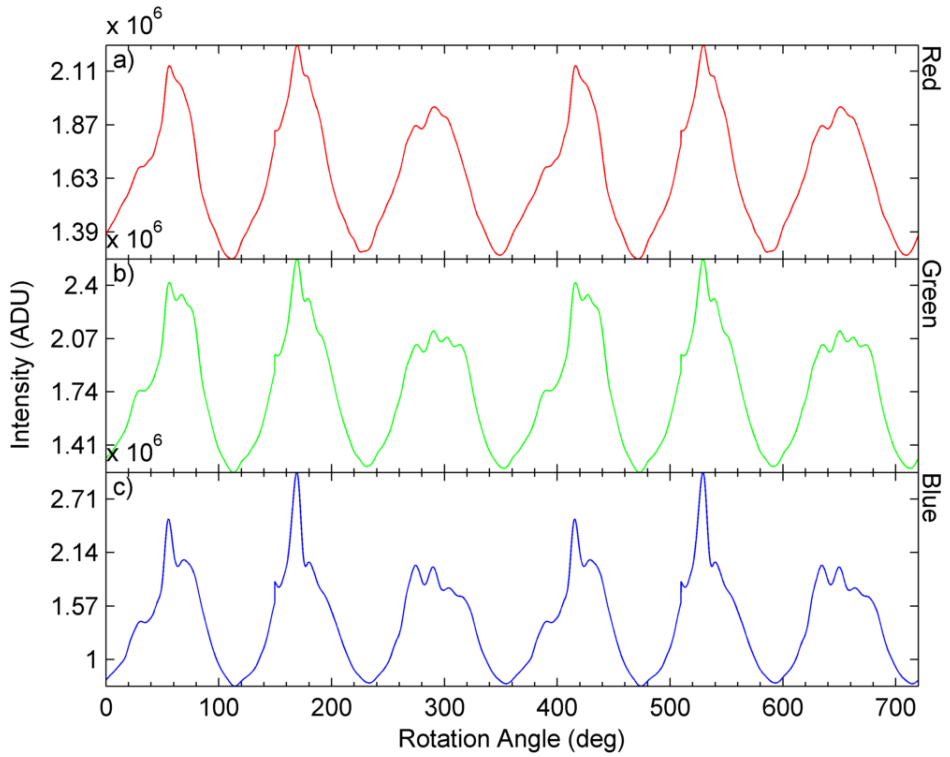

Figure 13. Model 4 again, this time with a star background. Notice that similar to the previous model, values are in general one order of magnitude larger.

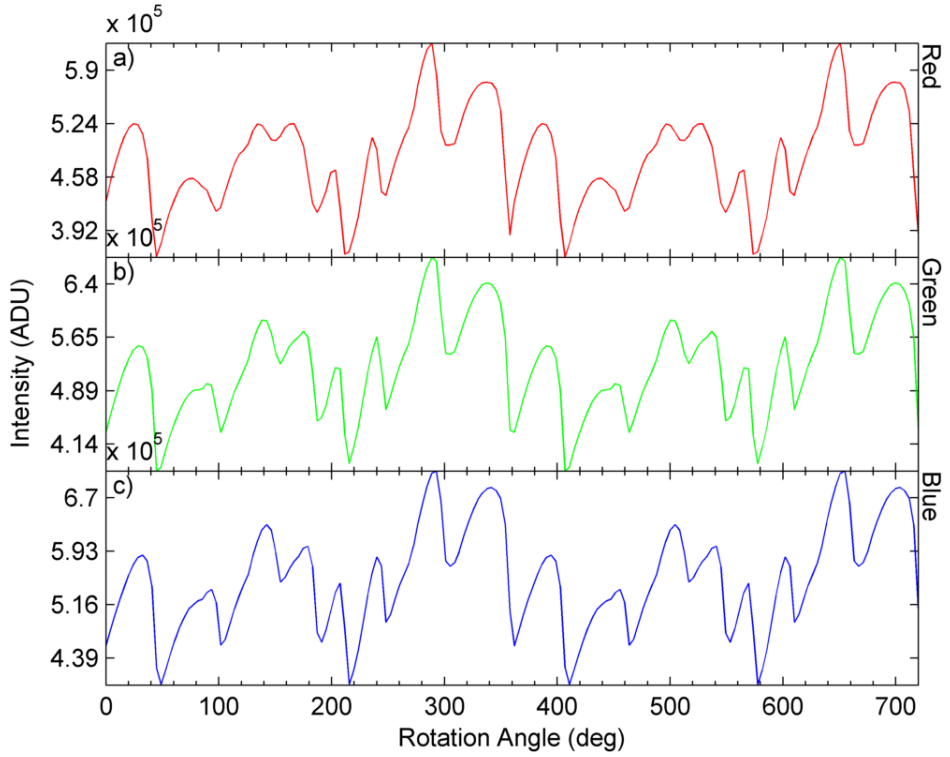

Figure 14. Model 6, an octagon with a white diffuse surface and an axis of rotation off center and at an angle. No star background

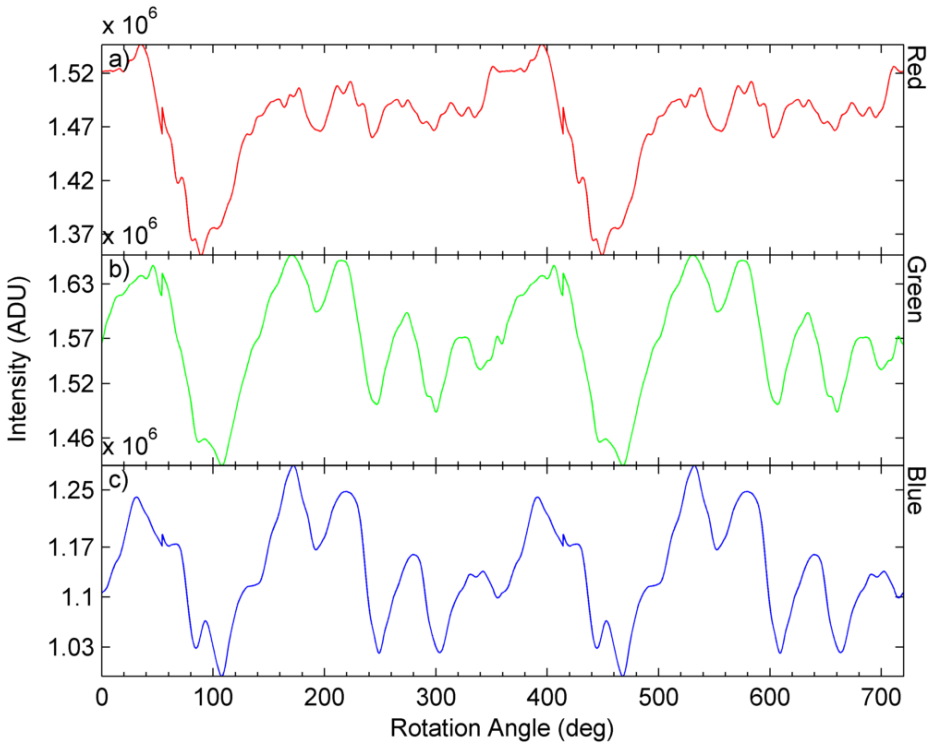

Figure 15. Model 6 again with a star background. Blue is similar, but red and green exhibit significant differenced due to glare reflected from the object against the screen.

The second set of plots was a comparison of our monochrome and color digital cameras. To make the comparison as relevant as possible, the color results were summed to give the total "white" intensity measured by the color camera. This, along with the monochrome light curve was plotted on a double y-axis graph. These plots are shown in Figures 16 - 17.

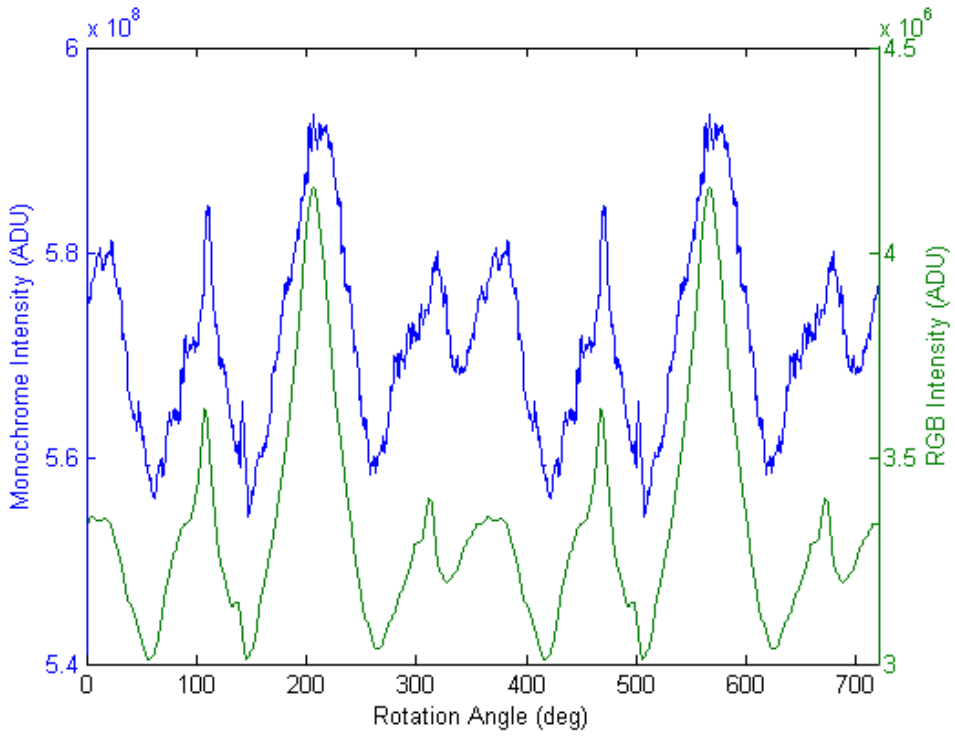

Figure 18. Object 3. Note that the peaks coincide accurately, though small details are lost due to the higher frame count of the monochrome camera.

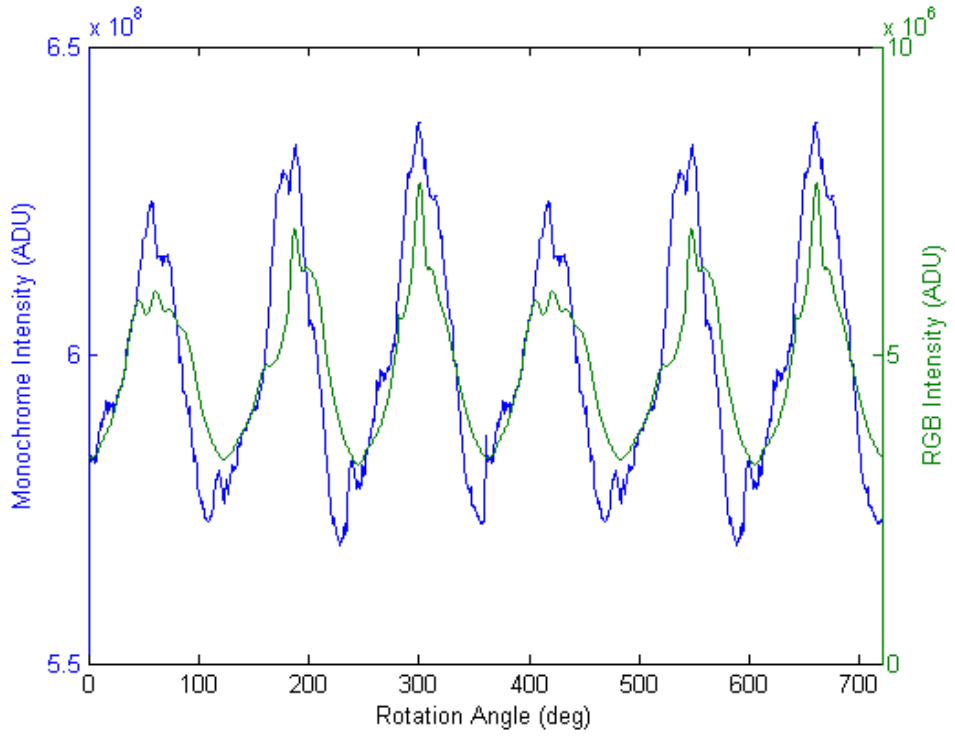

Figure 19. Object 4. Once again the peaks coincide, and have the same relative relationship to each other; however, the valleys of the monochrome images have additional nuances compared to the color images.

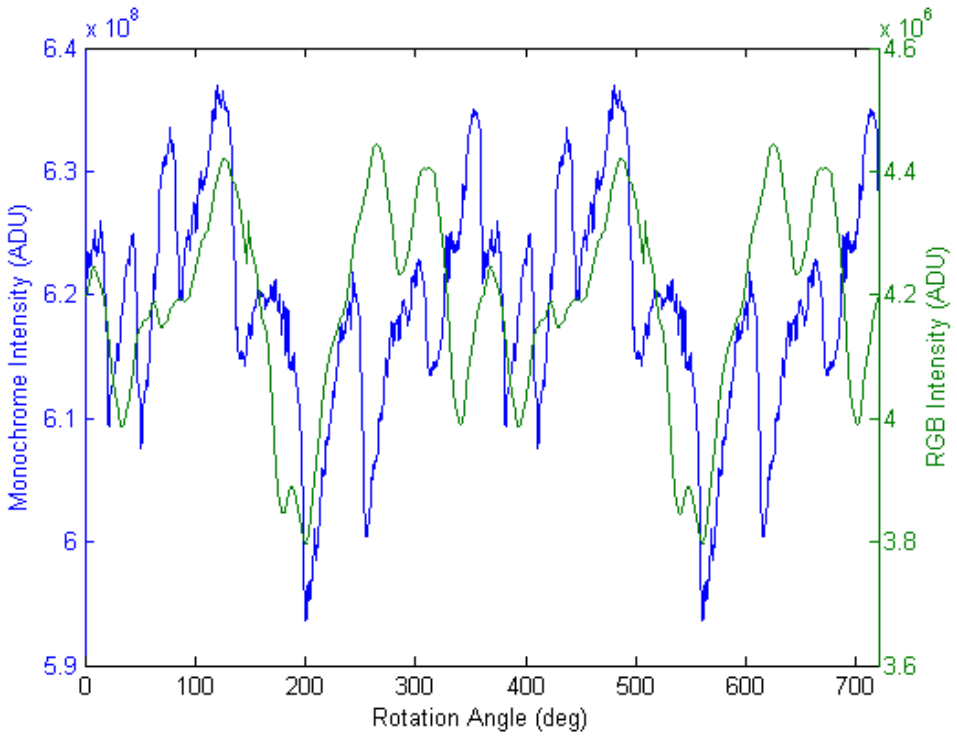

Figure 20. Object 6. This dataset exhibits the most variance between the two types of cameras. This object had significant reflection off of the screen, which was seen much more easily by the RGB camera than the monochrome one.

CHAPTER IV

DISCUSSION

Distortion values were well within the tolerances desired for such a system. In optical modeling systems, distortion is ideally less than 0.5 pixels in any direction. Thus, when the signal is discretized the values should round to the correct values. The distortion analysis for COSS found only a few dozen pixels which had distortion greater than 0.5 pixels.

There were several possible sources of distortion in COSS. The first and most concerning was the collimating lens. This lens was designed and fabricated specifically for this project, so verifying it was incredibly important to ensuring accurate results; however, we were quickly able to rule out the collimating lens as the source of distortion in this specific case. Lens distortion is usually symmetrical across some central axis unless damage has been done to the lens, but in this case the pixels with greater distortion are not radially symmetrical. It was determined by inspection that the lens was undamaged, which led us to the conclusion that the collimating lens was not a significant source of distortion.

Another source of distortion was in the glass pane half way through the optical path. This was caused by lab logistics, as we lacked a space to set out the entire project in an uninterrupted line. Thus, the glass pane had to be left in the way; however, the distortion's nature did not suggest the glass pane. Given that the glass pane was much larger than lens, distortion should be uni-directional; on the contrary, our data did not have uni-directional distortion. Rather, distortion seemed to be somewhat chaotic, occurring in differing directions, thus discounting the glass pane as a major source of distortion.

The third source of distortion which best fit our data is camera distortion. This distortion is caused by camera artifacts such stuck pixels, pattern noise, and heat based noise. We had great difficulty controlling heat based noise. The room where the camera and lens were located was not properly ventilated, so the heat from the computer controlling the camera was not properly removed from the room. This caused temperatures in excess of 80 degrees Fahrenheit for the ambient air temperature, and much higher temperatures on the camera itself. Thus, camera noise was large and very significant. This source of noise can be eliminated by using some sort of active cooling system on the camera itself.

A primary goal of this project was determining if adding a starfield behind an imaged object would drastically alter the light curves collected. Some variation change was expected since lighted points were added behind the object, but the backlighting of the monitor and other parameters were tested to ensure they did not alter the curve.

It was found that for objects 2, 3 and 4, the starfield caused more noise as expected, but the curve kept the same pattern, as seen comparing Figures 9 and 10, and 11 and 12. There were also magnitude differences in the collected data, but this was due to physical limitations of placing the light source at different angles with the starfield present or not.

Objects 1, 5, and 6 displayed drastically different plots for starfield versus no starfield. This is best shown by the difference between Figures 13 and 14. These figures differ greatly due to reflections off of the screen displaying the starfield itself; however, the major peaks remain

consistent with the no starfield data. Objects 5 and 6 are of octagon shape and have many sides to reflect the light at different angles, and object 1 has many fragmented pieces of solar panel to reflect light at varying angles. This is another physical limitation of the model that caused errors. Possible fixes would include having a larger starfield further back or a starfield on a non-reflective screen. If these stray reflections are eliminated, the data should line up for these objects too. Overall, having a starfield improves the accuracy of the data for a space scenario if physical placement limitations can be surmounted.

The starfield background did not cause any type of light curve compressing or otherwise cause the data to become significantly different. In fact, the most significant variance between model with a background and without was caused instead by the light reflecting off of the object and the screen. If the light curves were compressed or showed any kind of clipping of either maxima or minima, there would be reasonable doubt about the effects of the starfield on the data. However, in the absence of these data elements, it is safe to conclude that the overall brightness of the starfield does not significantly affect the light curve data.

Another concern with this project was whether or not color imagery would continue to be accurate. The monitor used for COSS was monochrome, which gave us additional bit depth; however, the primary camera we intended to image the monitor with was RGB. The key concern was that the color camera would not accurately capture the monochrome light from the monitor. Thus, in order to be sure that our conclusions about modeling models with and without starfields were valid, it was necessary to verify that there was no significant bias in the data relative to monochrome.

Our data showed significant correlation between light curves gathered by the monochrome camera and the sum of the light curves gathered by the RGB camera; conversely, the RGB data included in this paper looks “smoother” than the monochrome data. This is due to the fact that the RGB camera could take images at a 400 millisecond integration time, while the monochrome camera could image at a 95 millisecond integration time. Thus, we were able to capture many more data points per revolution for the monochrome data. While both cameras were capable of high frame capture rates, this particular scenario ended up requiring longer integration times than had been previously foreseen due to the dim lighting conditions. This caused a lower capture rate to be used for the color camera despite its capabilities

One question that needs to be asked about this is why use the color camera if it correlates with the monochrome data, but has a lower time resolution. The reason is that the color camera captures data not seen by the monochrome camera. Currently, we are working on using analysis of the color camera data to determine the material composition of the objects imaged. This is impossible to do for the monochrome camera, as it does not have any data other than overall intensity.

CHAPTER V

CONCLUSION

Object light curve modeling is a proven methodology, but in its current state it fails to account for potential background elements such as stars and planets. These elements should not be significant, but should somehow be factored into a modeling system in order to ensure its accuracy. One answer to integrating these elements is COSS. When a starfield simulation environment is used to populate the background, the end result is a simulation more closely approximating a space scenario. Raw light curves maintain their accuracy, allowing for the development of algorithms which take into account noise as a result of background light sources. Additionally, these results are independent of the type of camera used, as light curves are consistent for both monochrome and RGB cameras even if the starfield simulator uses a monochrome screen. Distortion factors for COSS are well within acceptable parameters, allowing our conclusions with regards to background generation to be applied more generally. By using a starfield in concert with a small object model, systems like COSS can accurately and rapidly generate physical lighting models that take real orbital lighting effects into account.

REFERENCES

1. Pollock, Thomas C., Peter M. Grubb, Imaging of Non-Resolved Objects Using the Fine Scale Optical Range. Advanced Maui Optical and Space Surveillance Conference Proceedings. 2012.

2. David S. F. Portree, Joseph P. Loftus, Jr., Orbital Debris: A Chronology. NASA publication NASA/TP-1999-208856, 1999.

3. V.M. Smirnov et al, "Study of Micrometeoroid and Orbital Debris Effects on the Solar Panels Retrieved from the Space Station 'MIR'", Space Debris, Volume 2 Number 1 (March, 2000), pp. 1 – 7. doi:10.1023/A:1015607813420.

4. William J. Broad and David E. Sanger. "Flexing Muscle, China Destroys Satellite in Test." New York Times. January 19th, 2007. <http://www.nytimes.com/2007/01/19/world/asia/19china.html?_r=1&pagewanted=all>.

5. "Debris from explosion of Chinese rocket detected by University of Chicago satellite instrument", University of Chicago press release, 10 August 2000.

6. Antony Milne, Sky Static: The Space Debris Crisis, Greenwood Publishing Group, 2002, ISBN 0-275-97749-8, p. 86.

7. Ettouati, I., Mortari, D., and Pollock, T.C. "Space Surveillance Using Star Trackers: Simulations," Paper AAS 06-231 of the 2006 AAS Space Flight Mechanics Meeting Conference, Tampa, FL, January 22-26, 2006 <http://aeweb.tamu.edu/mortari/pubbl/2006/Space%20Surveillance%20using%20Star%20Trackers%20-%20Part%20I%20-%20Simulations.pdf>

9. Chee Kai Chua; Kah Fai Leong, Chu Sing Lim (2003). Rapid Prototyping. World Scientific. p. 124. ISBN 978-981-238-117-0.

10. ASTM F2792-10 Standard Terminology for Additive Manufacturing Technologies, copyright ASTM International, 100 Barr Harbor Drive, West Conshohocken, PA 19428.

11. Carl Zeiss Microscopy. "Bayer mask." <http://www.zeiss.de/c1256b5e0047ff3f/Contents-Frame/777d91c572>

APPENDIX A

This data was taken as part of the FiScOR project[1]. It illustrates two broad points: that the data gathered from color imagery of small models has considerable detail in it, and that virtual lighting models such as Phong are highly inaccurate. Results are shown in Figures A1-A4.

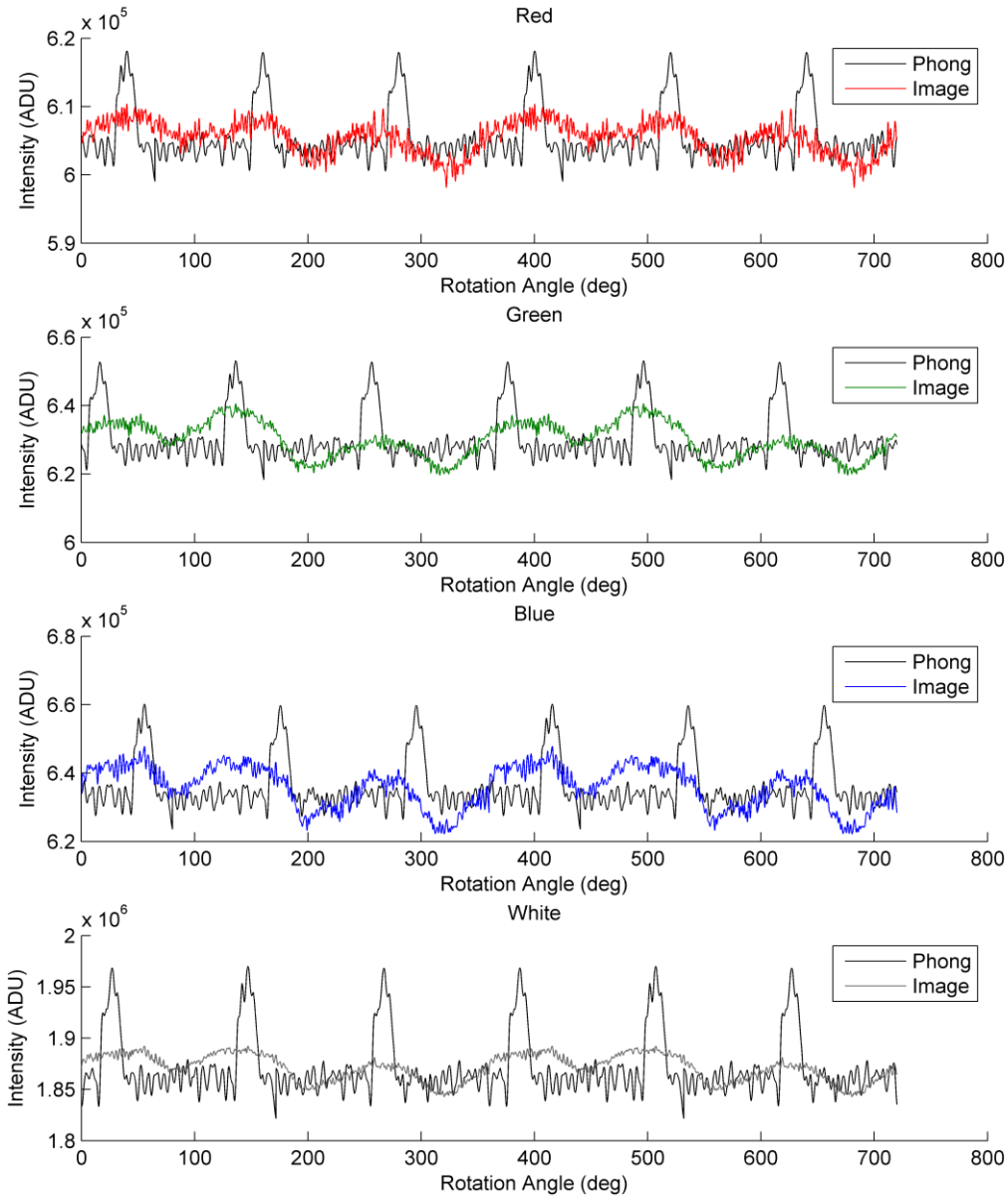

Figure A1. White diffuse cube, -10 deg horizontal, 16.5 deg altitude lighting. Note that the Phong parameters trend similar to the real data, but the real data has more significant peaks.

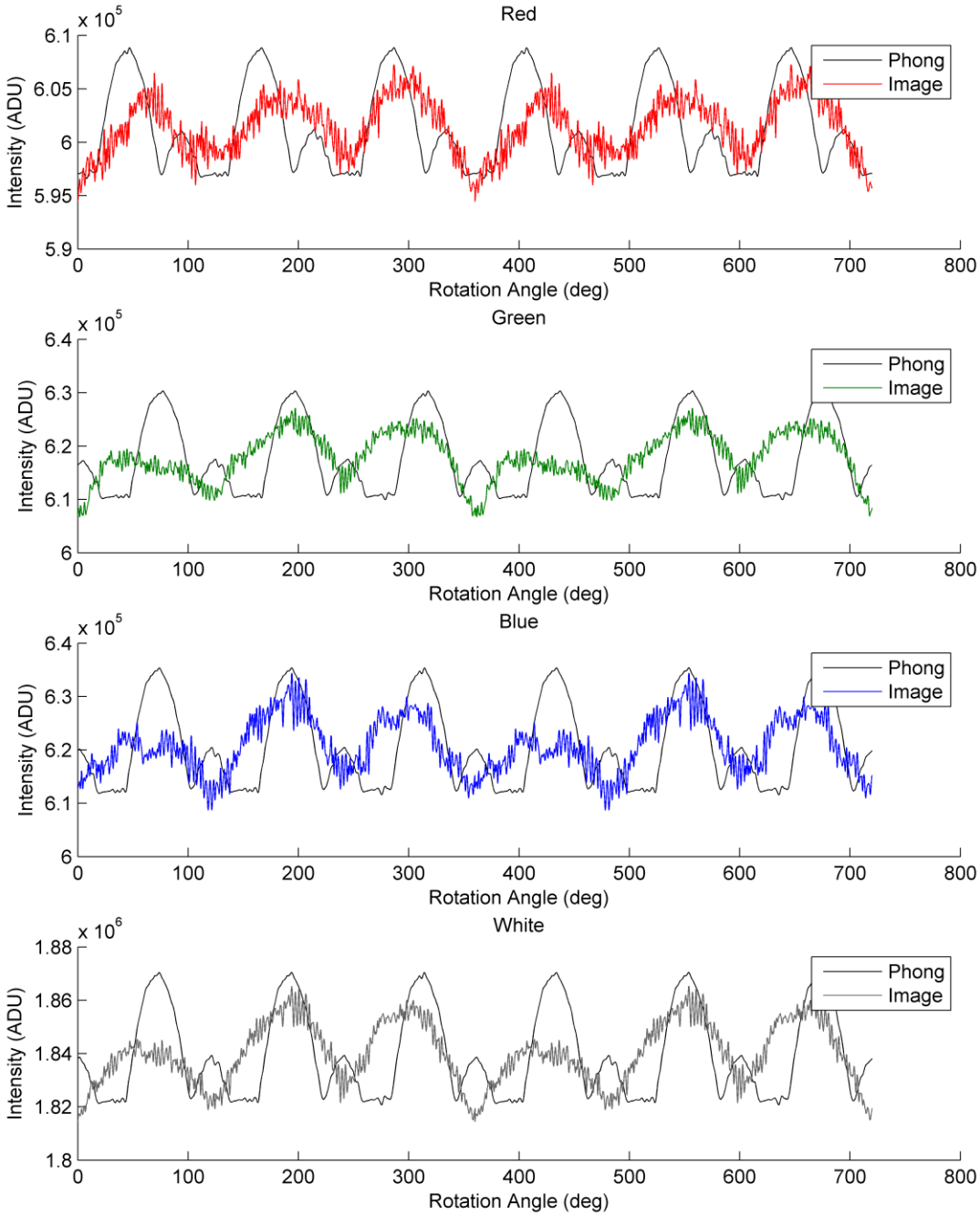

Figure A2. White diffuse cube, -45 deg horizontal, 16.5 deg altitude lighting

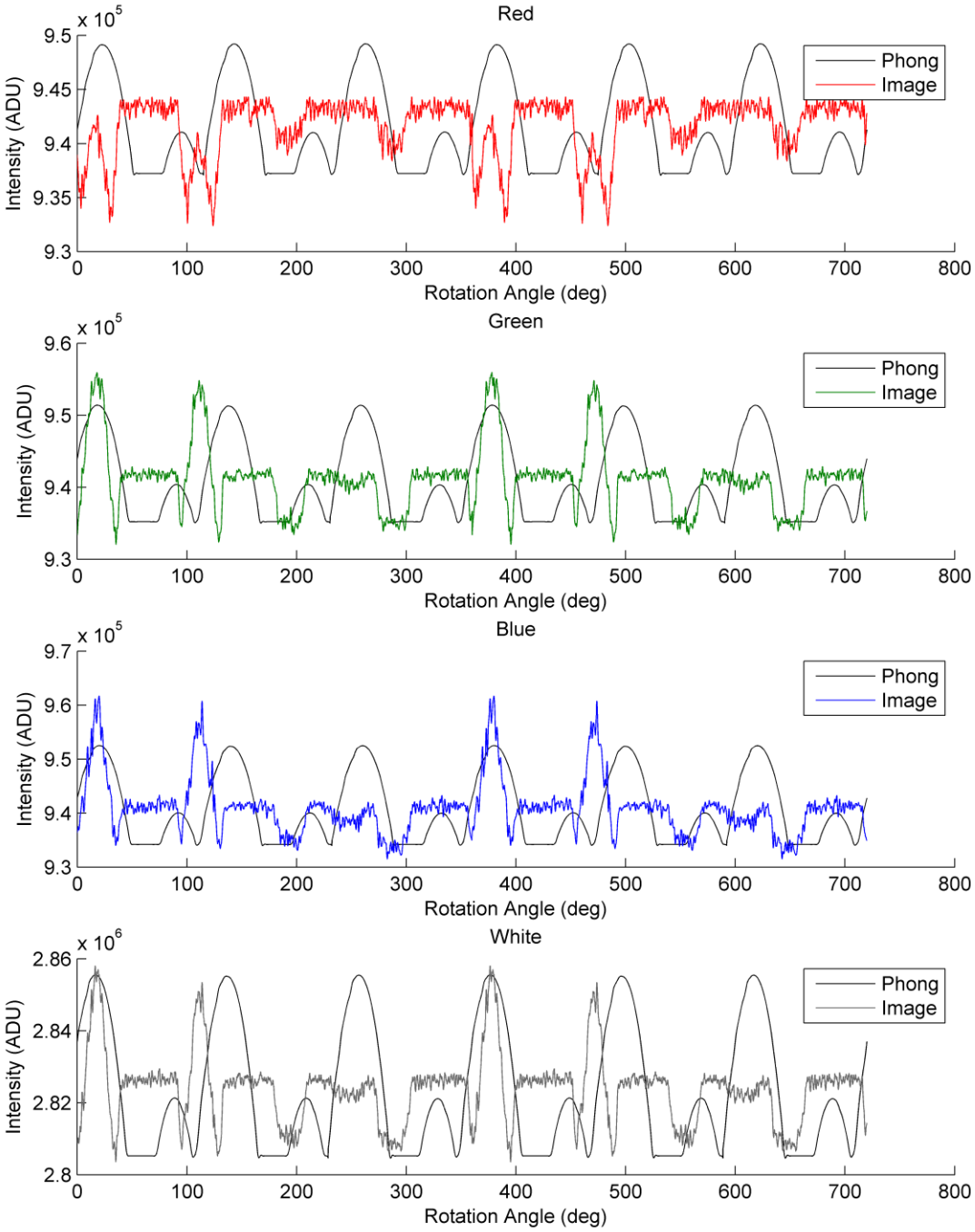

Figure A3.White diffuse cube, -135 deg horizontal, 16.5 deg altitude lighting angle

APPENDIX B

The two tables B1 and B2 each represent the x variances and y variances of every centroid in Figure 21. To find the total variance from the ideal center, go to a given cell in each table and pair.

Table B1. Variances along the x axis.

-0.25678	-0.15635	-0.15174	-0.03152	-0.12746	0.024242	0.353197	-0.10533	0.345084	-0.05879	0.048824	0.361535	0.22677	0.285942	0.503358
-0.3069	-0.18943	-0.08625	-0.1783	0.041681	0.028858	-0.44488	0.039055	-0.03195	0.067057	0.004781	0.071016	0.17178	0.375698	0.312248
-0.19847	-0.26353	-0.16036	-0.15497	-0.09561	-0.25989	-0.03811	0.172287	0.106727	0.172913	0.340665	0.134248	0.22217	0.300093	0.304219
-0.26444	-0.30686	-0.21273	-0.09944	-0.14817	-0.07506	-0.09769	-0.24333	0.037651	0.114673	0.106115	0.127422	0.043648	0.184212	0.222616
-0.21399	-0.29512	-0.13662	-0.13467	-0.25174	0.07634	0.199424	0.094448	-0.01296	0.020666	0.252709	0.098815	0.264075	0.246585	0.297485
-0.28729	-0.11027	-0.13788	-0.08284	-0.00199	-0.16514	-0.08138	0.06642	0.18327	0.12001	0.176996	0.283966	0.271383	0.323514	0.413323
-0.32142	-0.20606	-0.23375	-0.08794	-0.05913	0.180009	0.067994	0.048309	-0.06305	0.259025	0.282902	0.399833	0.163625	0.348707	0.34906
-0.18613	-0.19228	-0.16332	-0.13364	-0.06613	-0.19533	-0.01932	0.134069	0.004638	0.254051	0.173374	0.240924	0.28254	0.195601	0.432892
-0.26094	-0.33596	-0.1744	-0.14067	-0.11674	-0.13747	0.012557	0.096285	0.215905	0.119733	0.090112	-0.13697	0.293591	0.171599	0.11684
-0.20811	-0.20156	-0.20696	-0.10803	-0.05969	0.204516	0.056528	0.283355	0.300355	0.070454	0.245848	0.123907	0.244955	0.188488	0.63242
-0.20968	-0.09231	0.175284	0.003348	-0.14139	-0.24003	-0.00084	-0.20238	0.026372	0.010322	0.090192	0.282047	0.187419	0.346212	-0.05706
-0.16234	-0.15933	-0.21038	-0.11976	-0.05047	0.057301	0.090957	0.178389	0.240761	0.426009	0.101111	0.232592	0.312321	0.456659	0.484129
-0.37254	-0.20783	-0.16848	-0.11665	-0.03138	-0.01794	0.000681	0.125826	0.026102	0.23653	0.583658	0.261234	0.193258	0.47111	0.388964
-0.36613	-0.11969	-0.12977	-0.13572	-0.15817	0.003279	-0.09759	-2.21E-05	0.012688	0.025399	0.154984	0.216404	0.412416	0.043323	0.16132
-0.26801	-0.12129	-0.18088	-0.14554	-0.09754	-0.05823	-0.03794	0.071346	-0.00072	-0.07548	0.176155	0.076017	0.267252	0.504729	0.402615
-0.16297	-0.47453	-0.09288	0.006181	-0.07071	-0.20031	0.015174	-0.05023	0.081166	0.287125	-0.14011	0.264769	0.26113	0.272899	0.540794
-0.10227	-0.1431	-0.3633	-0.05982	-0.21387	-0.14384	-0.02216	-0.11608	-0.19188	-0.21548	0.076517	-0.02244	-0.13786	0.199632	0.233507
-0.32118	-0.20416	-0.19881	0.029125	-0.06784	-0.00678	-0.03982	0.084066	-0.02306	0.17327	0.196522	0.50512	0.641955	-0.11576	0.254468
-0.22255	-0.15717	-0.16273	-0.19779	-0.03975	0.021096	-0.05405	0.011947	0.102634	0.302408	0.141478	-0.20449	-0.05561	0.347283	0.490679
-0.26202	-0.1996	-0.19399	-0.16667	-0.07119	-0.10858	0.013984	0.010543	0.019774	-0.12358	0.309434	0.345102	0.248071	0.504103	0.507691
-0.28136	-0.21253	0.094424	-0.0131	-0.15797	-0.08641	-0.01484	0.025914	0.046804	0.105846	0.166262	0.515656	0.34757	0.213144	0.250978
-0.21927	-0.21746	-0.35535	-0.39345	-0.02528	0.010119	-0.14643	0.136389	-0.03187	0.031397	0.094478	0.143843	0.310985	0.297643	0.23812
-0.20199	-0.15668	-0.2191	-0.12801	-0.02876	-0.02517	-0.00675	0.076813	0.243279	0.149972	0.162664	0.08126	0.488756	0.196766	0.393727
-0.27519	-0.15671	-0.15908	-0.1108	-0.11121	-0.06177	0.01687	0.150231	-0.06752	0.267851	0.226839	0.138731	0.074038	0.346513	0.325193
-0.10374	-0.30071	-0.12073	-0.06616	-0.01268	0.03397	0.014386	-0.0418	0.029813	0.116546	0.221952	0.279495	0.238445	0.317109	0.245702
-0.18385	-0.14063	-0.13862	-0.11881	-0.04189	-0.06344	-0.02843	-0.02034	0.047	0.034279	0.115287	0.572338	0.485439	0.42539	0.412989
-0.13869	-0.09473	-0.00313	-0.0933	0.060218	-0.0298	0.029875	0.070864	0.135902	0.203682	-0.02535	-0.17885	0.263274	0.555775	0.421005

Table B2. Variances along the y axis.

0.059755	0.011161	-0.04338	0.164422	0.33078	0.097668	-0.00619	-0.06383	0.244434	-0.07679	-0.19987	0.421619	0.298374	0.196046	0.242983
0.077202	0.035326	0.057931	0.053215	-0.24431	0.135818	-0.03485	0.094149	-0.17409	-0.0552	-0.04085	0.373165	0.243092	0.153172	0.113609
-0.06169	-0.00701	-0.00465	0.046696	0.185958	-0.21878	-0.09999	0.432809	0.061928	-0.17556	0.139745	0.28757	0.014836	-0.04928	0.099484
-0.13697	-0.00919	-0.10819	-0.00079	-0.12615	-0.4148	-0.33997	0.371905	0.078662	-0.01645	0.333158	0.099658	0.019045	-0.05756	-0.02642
-0.00448	-0.03633	-0.16691	0.00325	-0.02543	0.136562	-0.04084	0.161	0.026852	-0.00458	-0.02548	-0.15465	-0.19169	-0.05965	-0.04685
0.021294	-0.03246	-0.04832	-0.02475	-0.06942	-0.04302	-0.04294	0.056753	0.189906	0.018721	-0.04172	-0.29904	-0.01657	-0.01969	-0.17216
0.004713	-0.06442	-0.19781	-0.16001	-0.2588	0.146158	0.29467	-0.01758	0.364809	-0.44763	-0.02401	-0.35224	0.125017	-0.02389	-0.10599
-0.10269	-0.16494	-0.04016	-0.03059	-0.05463	-0.1472	0.051748	0.150471	-0.02112	0.052331	-0.16024	-0.46896	0.069201	0.320034	0.052894
-0.14358	-0.01104	-0.25648	-0.08513	-0.15182	-0.112	0.109317	0.230564	-0.09584	0.486297	-0.25114	-0.36846	-0.1082	0.178607	-0.10059
-0.09766	-0.04525	-0.04327	-0.12067	-0.21522	-0.05748	-0.10568	-0.0508	0.240173	0.190175	-0.44897	-0.07372	-0.11898	0.011236	-0.24014
0.083147	-0.06535	-0.12049	-0.17507	-0.04047	-0.0551	-0.07095	-0.16773	0.280496	-0.20279	-0.79003	0.098522	-0.21177	-0.02314	-0.15082
-0.14848	-0.0833	-0.0691	0.009697	-0.21995	-0.19478	-0.19976	-0.12818	-0.10208	0.06654	-0.47117	0.033763	-0.13417	0.054407	-0.15822
-0.0714	-0.09311	-0.21203	-0.04452	-0.31719	-0.1716	-0.11916	-0.05815	-0.14599	-0.41472	0.152858	-0.48338	-0.08541	-0.30789	-0.33063
-0.23913	-0.21011	-0.05191	0.125292	-0.33109	-0.1261	0.277828	-0.11986	-0.13328	-0.14669	-0.33426	-0.58139	0.095371	-0.24691	-0.12198
-0.34718	-0.37482	-0.10109	-0.10572	-0.14497	0.077546	-0.19181	0.223683	-0.12056	0.16699	-0.27627	-0.28486	-0.19503	-0.05867	-0.32233
-0.24914	-0.11615	-0.02821	0.041509	-0.09006	-0.5006	-0.42464	-0.00082	0.493559	-0.62384	-0.19644	-0.02981	-0.25859	-0.12108	-0.28627
0.004902	-0.0606	-0.34889	-0.07214	-0.07001	-0.30981	-0.53808	-0.22722	0.426411	0.106031	0.035371	0.000707	-0.38638	-0.58715	-0.21669
0.161533	-0.16146	-0.06724	0.076428	-0.01477	-0.14295	-0.49208	-0.45334	0.106237	-0.19679	0.014452	-0.25036	-0.13187	-0.72641	-0.15343
0.051073	-0.31885	-0.27434	0.128816	-0.19926	-0.1382	-0.42632	-0.42384	-0.06458	-0.42756	-0.39968	0.001207	-0.03579	-0.35232	-0.16071
-0.03111	-0.29233	-0.26305	-0.10155	-0.37548	-0.18731	-0.32796	-0.25311	-0.04486	-0.16541	-0.49489	-0.28894	-0.18396	-0.02323	-0.21642
-0.19413	-0.31116	-0.348	-0.08066	-0.38594	-0.13823	-0.22435	-0.37205	-0.08204	-0.01485	-0.36154	-0.64781	0.061689	-0.50089	-0.2551
-0.32585	-0.12939	-0.15399	-0.33867	-0.30662	-0.29237	-0.24971	-0.25162	-0.05449	-0.03358	-0.07379	-0.3791	-0.11288	-0.3178	-0.16228
-0.3396	-0.12595	-0.23836	-0.28464	-0.46448	-0.25587	-0.42526	-0.54507	-0.11948	-0.21586	-0.08414	0.015689	-0.12422	-0.26775	-0.26002
-0.22008	-0.13103	-0.24419	-0.18718	-0.3039	-0.26349	-0.31057	-0.17905	-0.32178	-0.81191	-0.14385	-0.12883	-0.45627	-0.06918	-0.23531
-0.17433	-0.18861	-0.16657	-0.13491	-0.18299	-0.27792	-0.31411	0.138031	-0.6368	-0.69508	-0.21693	-0.40807	-0.39463	0.087274	-0.37045
-0.43882	-0.38895	-0.27845	-0.289	-0.23286	-0.40324	-0.43208	0.051473	-0.50087	-0.47844	-0.04848	-0.41969	-0.62545	-0.04035	-0.34533
-0.30299	-0.40438	-0.40197	-0.13776	-0.11356	-0.26434	-0.61801	-0.06383	-0.35914	-0.30925	0.111014	-0.10607	-1.00941	-0.14438	-0.29529