

Effect of Wavelength Selection on the Segmentation of Hyperspectral Images Using Thresholding of Gray Level Images

By: Derienne Black

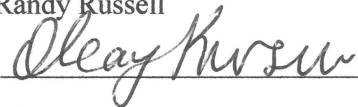
An Undergraduate Thesis Submitted to The University Honors Program Auburn University at Montgomery

In partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science



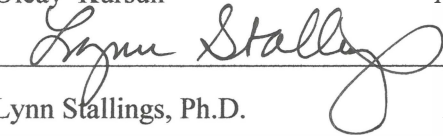
Randy Russell

April 25, 2023



Olcay Kursun

April 25, 2023



Lynn Stallings, Ph.D.

April 25, 2023

Director, University Honors Program

© Copyright by Derienne Black on April 25, 2023

I understand that my project will become part of the permanent collection of the Auburn University at Montgomery Library, and will become part of the University Honors Program collection. My signature below authorizes release of my project and thesis to any reader upon request.



Derienne Black

Table of Contents

Abstract	3
Introduction	3
Data Collection Details	3
Image Segmentation	6
Thresholding Methods	9
Conclusion	8
Appendices	9

ABSTRACT

Auburn University at Montgomery Chemistry Department has partnered with the Computer Science Department to participate in a National Science Foundation research project. This study utilizes a hyperspectral imaging camera to capture cloud data images. A calibrated spectrum comprised of 462 narrow wavelength bands is recorded at each pixel of the image. These images have been segmented using the software system Wolfram Mathematica. The following three types of thresholding techniques were implemented to configure the segmented images: Otsu, Kapur's Entropy, and Kittler-Illingworth Minimum Error. The purpose of this study is to determine whether normalizing the data by the average radiance yields a better segmentation. Each pixel of a sky image is represented by a spectrum showing how monochromatic radiance varies with wavelength. One channel was used to test normalization. This channel was proportional to total radiance for all sky conditions.

1 INTRODUCTION

Hyperspectral imaging cameras are used to capture the light spectrum of each pixel in a specified image. These cameras are used in a variety of applications such as the medical field, agriculture, government, etc. The information obtained by these analyses has contributed to countless discoveries. Hyperspectral imaging cameras have been used to characterize wounds, locate invasive weeds, and even for surveillance and

reconnaissance against hard to detect targets. In this study, the Resonon Pika XC2 Hyperspectral imaging camera was used to capture images of numerous types of clouds. The purpose of this study is to determine whether normalizing images by the average radiance yields a better image segmentation.



Spectral Range (nm)	400 - 1000
Spectral Channels	447
Spectral Bandwidth (nm)	1.3
Spectral Resolution - FWHM (nm)	1.9
Spatial Pixels	1600
Max Frame Rate (fps)	165
f/#	2.4

Figure 1: Resonon Pika XC2 Hyperspectral Imaging Camera with specifications

2 DATA COLLECTION DETAILS

2.1 Elevation and Azimuth

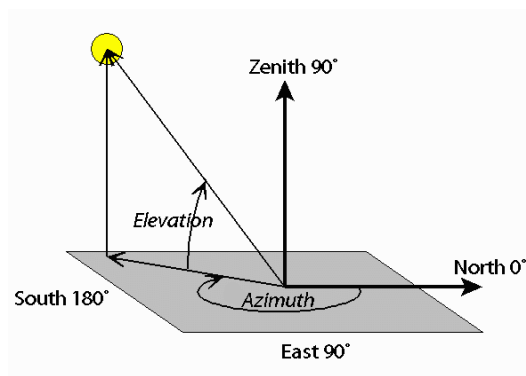


Figure 2: Elevation and Azimuth Diagram

To relate radiance to radiant energy received from clouds it is necessary to know the elevation of each pixel in the image. The azimuth of each pixel was also recorded. Azimuth is defined as the “angle between North, measured clockwise around the observer's horizon, and a celestial body (sun, moon).” [9] Using a compass to specify azimuth angle, we rotated the Resonon camera to various angles. Once the camera was positioned, we used the Spectronon Software to set parameters and begin scanning.

2.2 Resonon Pika XC2 Interior

The inside of the Resonon Pika XC2 houses collimating lens that assists in controlling the field of view and spatial resolution. This curved optical lens ensure that the light rays enter the Prism-Grating-Prism (PGP) are parallel.

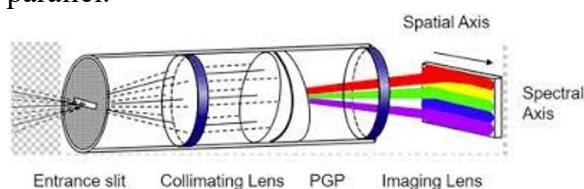


Figure 3: Resonon Pika XC 2 Interior

2.3 Spectronon Software- Camera Operation

The Spectronon Software allows us to program the specifics of how we want the camera to capture the image. There are three main parameters that were adjusted: integration time, frame rate and scan rate. Integration time is defined as “the length of time during which the sensing elements are collecting charge between transfers to the transport registers.”[1] In simpler terms, this is how long the sensor is exposed to light. Frame rate is the speed or frequency at which frames are captured. Frame rate is

measured in hertz or frames per second. Frame rate is responsible for the smoothness of an image. Scan rate describes the rate at which the camera rotates while frames are being taken. To synchronize the scan rate with the rate of which we are taking the frames, we adjusted the scan and frame rate. The scan rate was set to 0.93 seconds and the frame rate was set to approximately 45 frames per second. The integration rate occasionally varied but remained between 9.0-12.0 ms. Once these parameters are set, the Pika XC2 rotates around a vertical axis, sweeping the integration field of view across the sky.

2.4 Spectronon Software – Data Representation

The Spectronon software provides an RGB render of the data cube using three user-selected wavelengths. The user can then see the spectral graph of that point that shows the radiation levels at numerous wavelengths. Figure 4 is a scanned image that is pinpointed at sample 260, line 85. This sample and line combination is a bright section of the captured cloud.

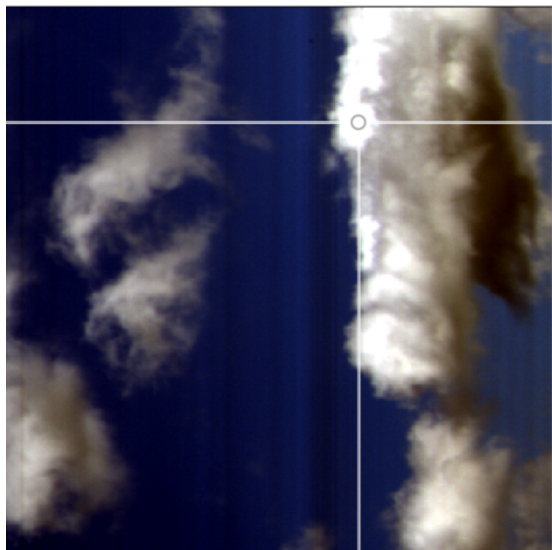


Figure 4: Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Sample 260 Line 85

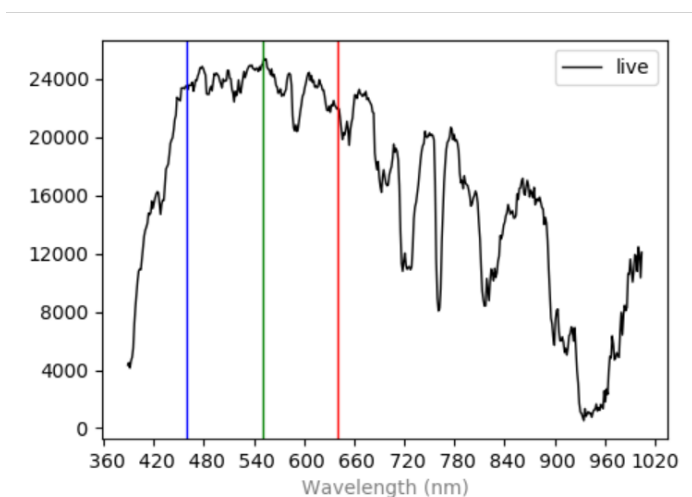


Figure 5: Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Spectra Graph Sample 260 Line 85

The corresponding spectrum also displays the wavelengths selected for producing the RGB render as seen in figure 5. The maximum monochromatic radiance level is above 24000 rad. Bright clouds can have higher levels of radiation because they reflect and scatter more of the incoming radiation from the sun than darker clouds.

Figure 6 displays a point at sample 359 line 192 which is a dark portion of cloud. The corresponding spectra in figure 7 shows how dark clouds have lower levels of radiation compared to bright clouds because they receive less incoming radiation from the sun. This is due to their higher opacity or ability to absorb light. The bottoms of high opacity clouds are shaded from direct sunlight by the cloud above, and little solar radiation is able to penetrate to the bottom of the cloud.

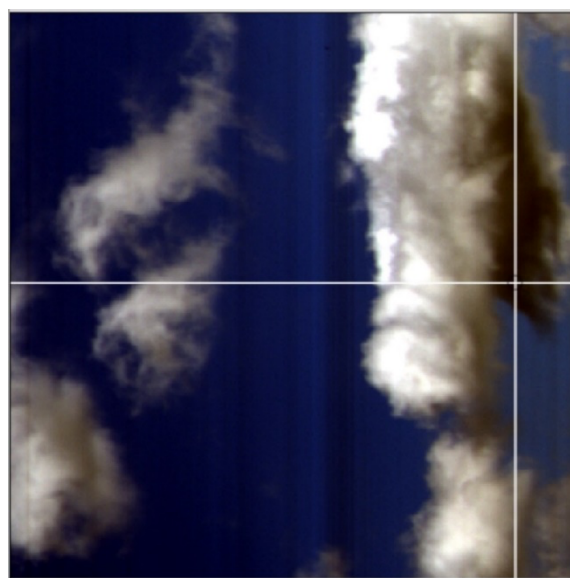


Figure 6: Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Sample 359 Line 192

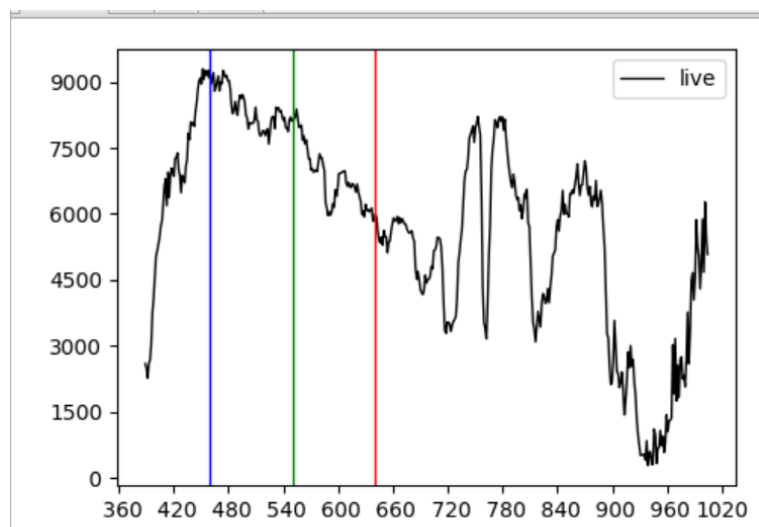


Figure 7: Scan_07-18-
2022_0946_AZ075_EL20_L_D-R1 Spectra Graph
Sample 359 Line 192

3 IMAGE SEGMENTATION PROCESS

3.1 Selection of Normalization Wavelength

Radiance is the rate at which radiant energy is received per solid angle subtended by an image pixel. To segment an image into clear sky and cloud, a normalized grayscale image is produced by dividing the radiance at each pixel by the radiance at the normalization wavelength. The monochromatic radiance at 586.3 nm is proportional to the average radiance over wavelength for pixels of all sky conditions. By selecting this wavelength as the normalization wavelength, the normalized radiance shows the distribution of radiant energy between the various wavelength bands. The spectra shown in figures 8 & 9 demonstrate the effect of normalization on spectra.

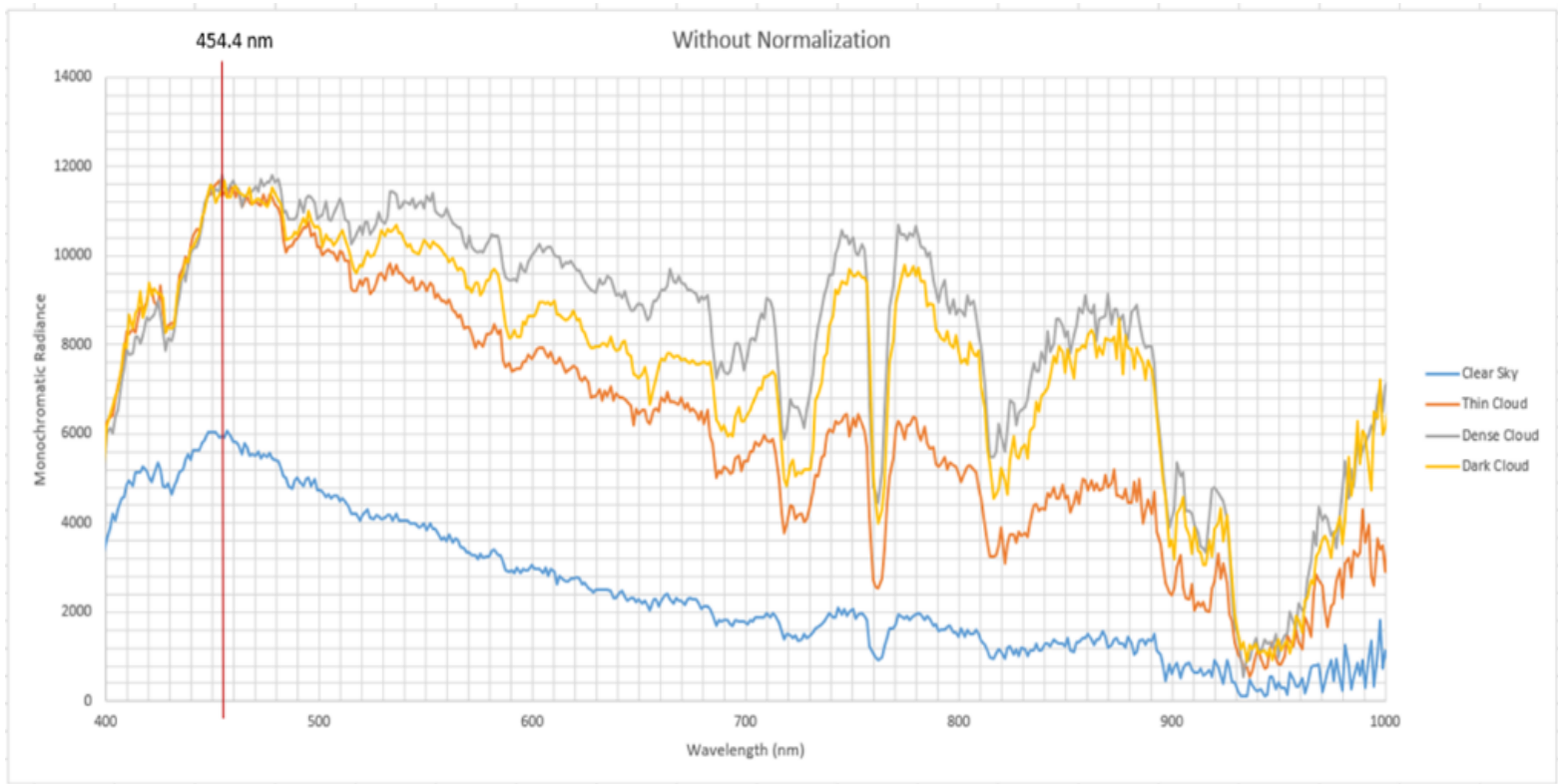


Figure 8: Unnormalized Cloud and Sky Definitions Comparison

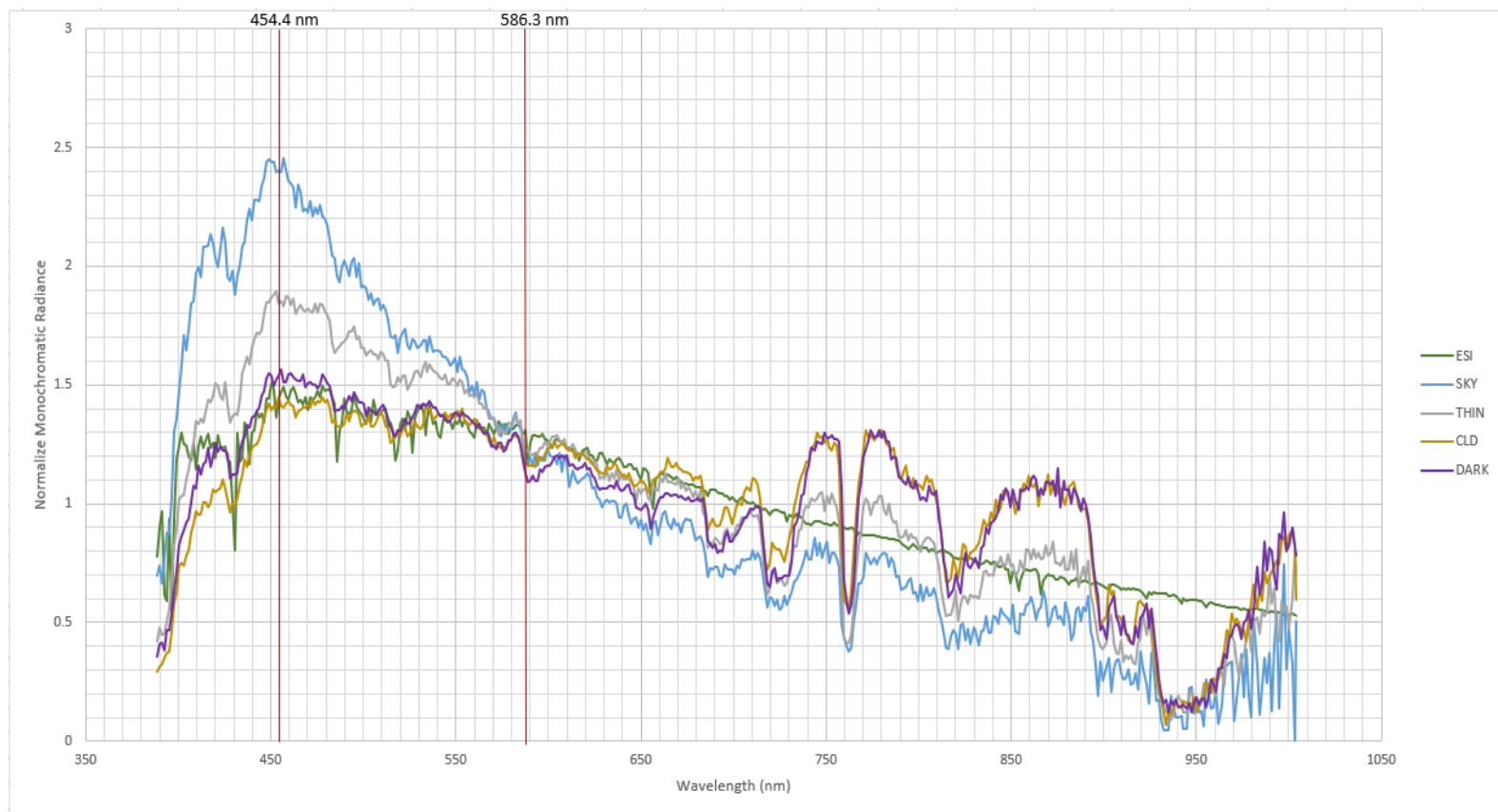


Figure 9: Normalized Cloud and Sky Definitions Comparison

3.2 Normalized Vs. Unnormalized Data

Normalization is the process of changing an image's pixel intensity range to reveal particular aspects of the data which make the data more understandable. Using the Wolfram Mathematica Software, we were able to create histogram data. Histogram data is used to demonstrate the distribution of pixel intensity values in a grayscale image. The x-axis represents bin size which is the intensity value range. The y-axis represents the total number of pixels or frequency. The grayscale histogram of an image can be determined by the following equation:

$$H(i) = \sum_{x=1}^M \sum_{y=1}^N [I(x, y) \in i] \quad (1)$$

In this equation, M and N represent the width and height of the image. “[$I(x, y) \in i$]” is a function that returns one (1) if the pixel at position (x, y) has an intensity value within a specified range for bin i , and zero (0) otherwise.” [2] Figure 10 is the histogram for an image that is unnormalized. Figure 11 is the histogram for that same image that has been normalized.

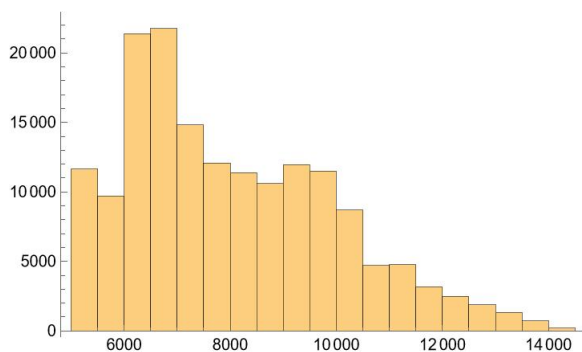


Figure 10: Scan_07-18-2022_0958_AZ270_EL35_L_D-R1 Un-Normalized Histogram 454p4nm

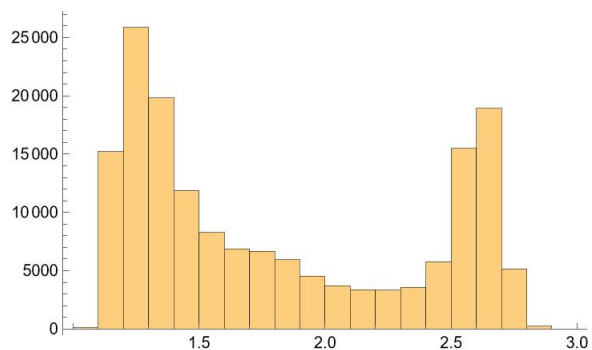


Figure 11: Scan_07-18-2022_0958_AZ270_EL35_L_D-R1 Normalized Histogram 454p4nm

While the frequency levels are similar in both histograms, the bin sizes are substantially different. The bin sizes in the unnormalized histogram range in the thousands. The most notable difference between the histograms is that the normalized histogram is clearly bimodal whereas the unnormalized histogram is not. Since the thresholding methods used in this study assume two overlapping distributions are present, the normalized histograms are better suited for segmentation by thresholding.

4 THRESHOLDING METHODS

4.1 Otsu's Method

Otsu's Method is an image thresholding method named after Nobuyuki Otsu. Otsu's method automatically determines an optimal threshold value for segmenting an image into two classes, typically foreground and background. The key idea is to find a threshold that maximizes the separation between foreground and background pixels, resulting in an optimal threshold for image segmentation.

The Otsu method algorithm works by calculating the optimal threshold value that maximizes the separability of the foreground and background regions. The algorithm assumes that the image contains two classes

of pixels: foreground and background and aims to find the threshold value that minimizes the within-class variance and maximizes the between-class variance.

The within-class variance is defined as the sum of the variances of the foreground and background classes, weighted by their respective probabilities, while the between-class variance is the squared difference between the mean intensities of the two classes, also weighted by their probabilities. The optimal threshold value is the one that maximizes the ratio of the between-class variance to the within-class variance.

Once the optimal threshold value is determined, the image is thresholded by assigning all pixels with intensity values above the threshold to the foreground and all pixels with intensity values below the threshold to the background. The resulting binary image can then be used for further analysis or processing.

For this study, a code was written using Wolfram Mathematica Software (found in appendix) that allows a user to import a subset bip image and get the output of the resulting histogram and thresholded image. Figures 12,13, and 14 illustrate an imported subset image, the output image thresholded using Otsu's method normalized, and the output image thresholded using Otsu's

method unnormalized. The normalized method produced a more accurate representation of the original image.



Figure 12 Original Scan_07-18-2022_0946_AZ075_EL20_L_D-R1



Figure 13 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Otsu segmented image normalized.



Figure 14 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Otsu segmented image Unnormalized

4.2 Kapur's entropy thresholding

Kapur's entropy-based method is an image thresholding technique that determines an optimal threshold value by maximizing the information content or entropy of the resulting binary image.

Kapur's entropy method is a thresholding technique used for image segmentation. The method aims to find the threshold value that minimizes the total entropy of the segmented image. In this method, the image is first divided into two classes based on a threshold value, and then the entropy of each class is calculated. The threshold value that minimizes the total entropy of the segmented image is selected as the optimal threshold.

The entropy of an image is a measure of the amount of information contained in the image. It is calculated using the probability distribution of the pixel intensities in the image. The pixels with intensities below the threshold value are assigned to one class, and the pixels with intensities above the threshold value are assigned to the other class. The formula for Kapur's entropy thresholding method is as follows:

$$T = \operatorname{argmax}(T) H(C1, T) + H(C2, T) \quad (2)$$

where T is the threshold value, $\operatorname{argmax}(T)$ denotes the threshold value that maximizes the sum of entropies ' $H(C1, T)$ ' and ' $H(C2, T)$ ', and ' $C1$ ' and ' $C2$ ' are the two classes defined by the threshold ' T '. The entropy of a class C is calculated as:

$$H(C) = - \operatorname{sum}(p(i) \log(p(i))) \quad (3)$$

where ' $p(i)$ ' is the probability of pixel intensity ' i ' belonging to the class ' C '.

In practice, Kapur's method involves computing the histograms of the image and iteratively calculating the entropies and the threshold value until convergence. Once the optimal threshold value is determined, the image is segmented into two classes based on the threshold, with all pixels below the threshold classified as background and all pixels above the threshold classified as foreground.

Figures 16 and 17 show the entropy thresholding normalized and unnormalized image pertaining to figure 15. The normalized entropy image produced a clearer representation of the data cube image.



Figure 15 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1



Figure 16 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 entropy segmented image normalized



Figure 17 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 entropy segmented image unnormalized

4.3 Kittler-Illingworth minimum error thresholding

Kittler-Illingworth minimum error thresholding is an image thresholding technique that aims to find an optimal threshold value by minimizing the classification error between foreground and background pixels. The method involves calculating the probabilities of foreground and background pixels for each possible threshold value, and then computing the error associated with each threshold by considering misclassifications. The error is minimized by selecting the threshold that results in the lowest overall classification error. Kittler-Illingworth thresholding takes into account both the intensity distribution and the local variances of the foreground and background pixels, and it is particularly effective in handling images with uneven lighting conditions or complex intensity

distributions. In summary, Kittler-Illingworth minimum error thresholding minimizes classification error by considering probabilities, intensity distribution, and local variances, and selects the threshold that results in the lowest overall error for image segmentation.

As seen in figures 19 and 20, Kittler-Illingworth minimum error produces a normalized segmentation that underestimates clear sky regions in places, but correctly identifies the dark cloud as a cloud region. The unnormalized segmentation misrepresents the dark cloud as clear sky. The normalized image shows signs of overcorrection as some of the clear sky is merged with portions of the sky as seen in figure 19.



Figure 18 Original Scan_07-18 2022_0946_AZ075_EL20_L_D-R1



Figure 19 Scan_07-18 2022_0946_AZ075_EL20_L_D-R1 minimum error segmented image normalized

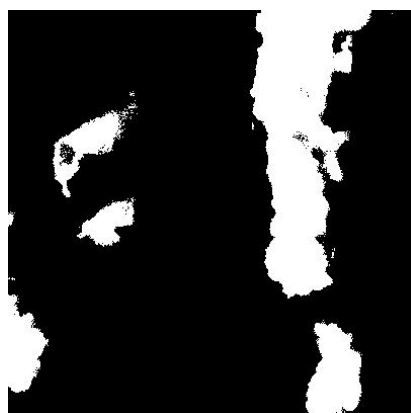


Figure 20 Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 minimum error segmented image unnormalized

4.4 Normalized Mutual Information and the Adjusted Rand Index

Similarities between the clusters produced by the three methods were compared using Normalized Mutual Information (NMI) and the Adjusted Rand Index (ARI). NMI measures the mutual information between two clustering's while taking into account the sizes of the clusters. Mutual information is a measure of the amount of information shared between two variables. In the context of clustering, NMI measures the mutual information between two different cluster assignments of the same set of data points. The NMI value ranges from 0 (no mutual information) to 1 (perfect agreement).

ARI, on the other hand, measures the agreement between two clustering's while adjusting for chance agreement. The ARI value ranges from -1 (no agreement) to 1 (perfect agreement), with 0 indicating the amount of agreement that would be expected by chance. ARI considers all pairs of data points and counts how many pairs are assigned to the same cluster in both clustering's, or to different clusters in both clustering's.

Both measures are widely used in clustering and segmentation evaluation. However, NMI is more suitable for datasets with clusters of different sizes, while ARI is more suitable for datasets with similar cluster sizes. The choice between NMI and ARI depends on the specific application and the

desired level of sensitivity to cluster size and chance agreement.

The graphs in figure 21 display the average NMI results of the compared three thresholding methods used in this study. Overall, Kapur Entropy Thresholding and Otsu method at normalization in both ARI and NMI have an average value that is closer to 1. This means that Entropy and Otsu have better perfect agreement. These results confirm the image results that were obtained by the Mathematica Software. Otsu and Entropy images were more similar compared to the minimum error images.

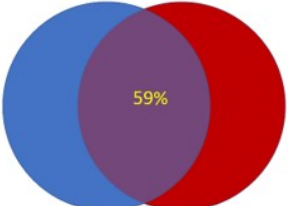
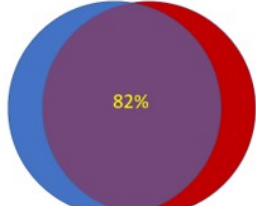
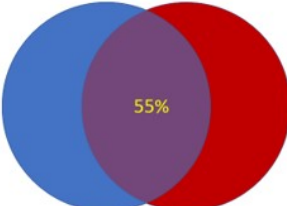
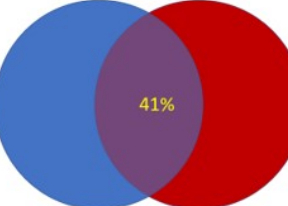
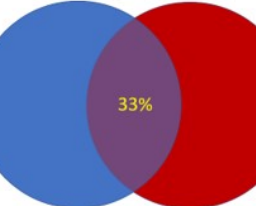
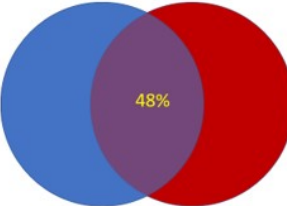
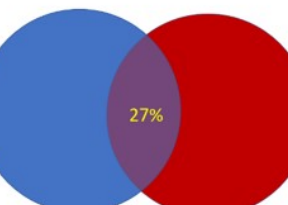
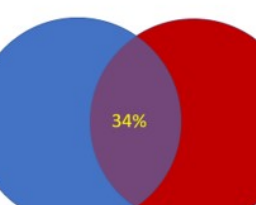
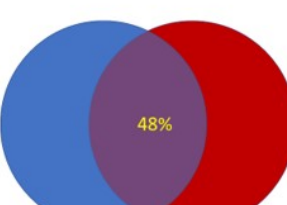
NMI	Kapur vs Kittler-Illingworth	Kapur vs Otsu	Otsu vs Kittler-Illingworth
Normalized 454.4 nm			
Unnormalized 454.4 nm			
Unnormalized 586.3 nm			

Figure 21: Normalized Mutual Information
Table of 5 captured images

5 CONCLUSION

5.1 Results and Future Work

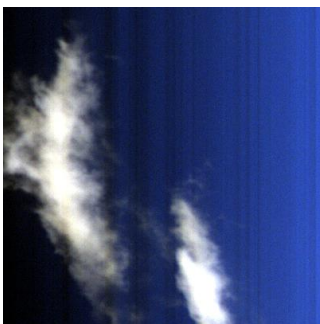
The distribution of normalized radiance among image pixels is clearly bimodal, while the distribution of unnormalized radiance is not. Because thresholding methods assume the presence of two overlapping distributions, normalization results in more accurate and more consistent segmentations. Segmentations based on normalized radiance utilize primarily chromatic information. Compared to segmentations based on spectrally averaged radiance, segmentations performed using normalized radiance were more accurate.

Hyperspectral imaging continues to improve and excel in potential capabilities. In the future, this process could be used to find the amount of radiation reflected from clouds compared to that which is scattered by clear sky. Researchers may also be able to use these cameras to classify cloud types as well as monitoring radiative properties of clouds to determine cloud climate.

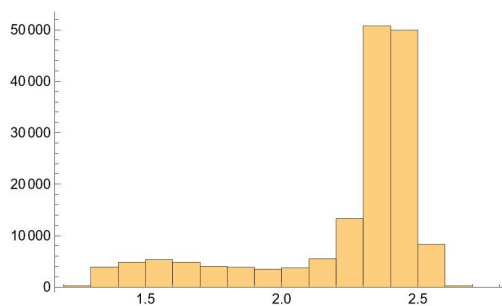
REFERENCES

- [1] "Definition of Integration Time." *Photonics.com*, https://www.photonics.com/EDU/integration_time/d4790
- [2] Baeldung, W. by: (2023, April 14). Understanding Otsu's method for image segmentation. Baeldung on Computer Science. Retrieved April 20, 2023, from <https://www.baeldung.com/cs/otsu-segmentation>
- [3] Kapur, J. N., Sahoo, P. K., & Wong, A. K. C. (2006, July 18). *A new method for gray-level picture thresholding using the entropy of the Histogram*. Computer Vision, Graphics, and Image Processing. Retrieved April 18, 2023, from <https://www.sciencedirect.com/science/article/pii/0734189X85901252>
- [4] Kittler, J., & Illingworth, J. (2003, May 19). *Minimum Error Thresholding*. Pattern Recognition. Retrieved April 18, 2023, from <https://www.sciencedirect.com/science/article/abs/pii/0031320386900> 300
- [5] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," in IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62-66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.
- [6] "What Are the 'Azimuth and Elevation' of a Satellite?: Memorial Spaceflights." *What Are the "Azimuth and Elevation" of a Satellite? | Memorial Spaceflights*, <https://www.celestis.com/resources/faq/what-are-the-azimuth-and-elevation-of-a-satellite/>
- [7] Hong Yan Department of Electrical Engineering, Yan, H., R.C., G., P.K., S., S.U., L., N., O., J., K., S.K., P., L.K., H., & C.W., T. (1996, December 1). Unified formulation of a class of image thresholding techniques. ACM Digital Library. Retrieved May 4, 2023, from <https://dl.acm.org/doi/10.1016/S0031-3203%2896%2900050-7>
- [8] Kittler, J., and J. Illingworth, "On threshold selection using clustering criteria," in IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-15, no. 5, pp. 652-655, Sept.-Oct. 1985, doi: 10.1109/TSMC.1985.6313443.
- [9] Deloney, M. L. (2021, June 1). What is azimuths surveying: What is bearings surveying: Difference between Azimuths and bearings in surveying. CivilJungle. Retrieved May 4, 2023, from <https://civiljungle.com/azimuths-and-bearings/>

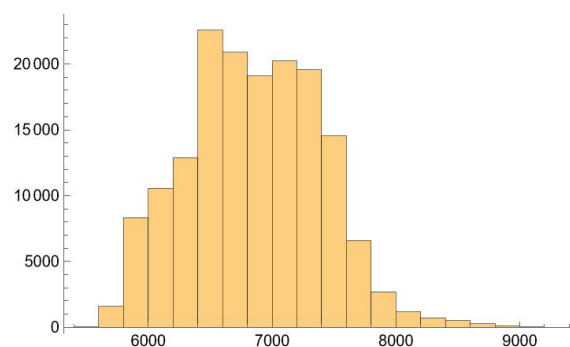
Appendix A: Scan_07-18-2022_0931_AZ075_EL20_L_D_R5 Data



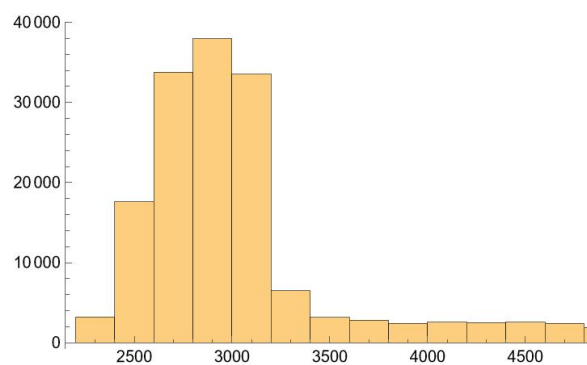
(Original Data Cube Image)



(Normalized Histogram Data) 454.4 nm

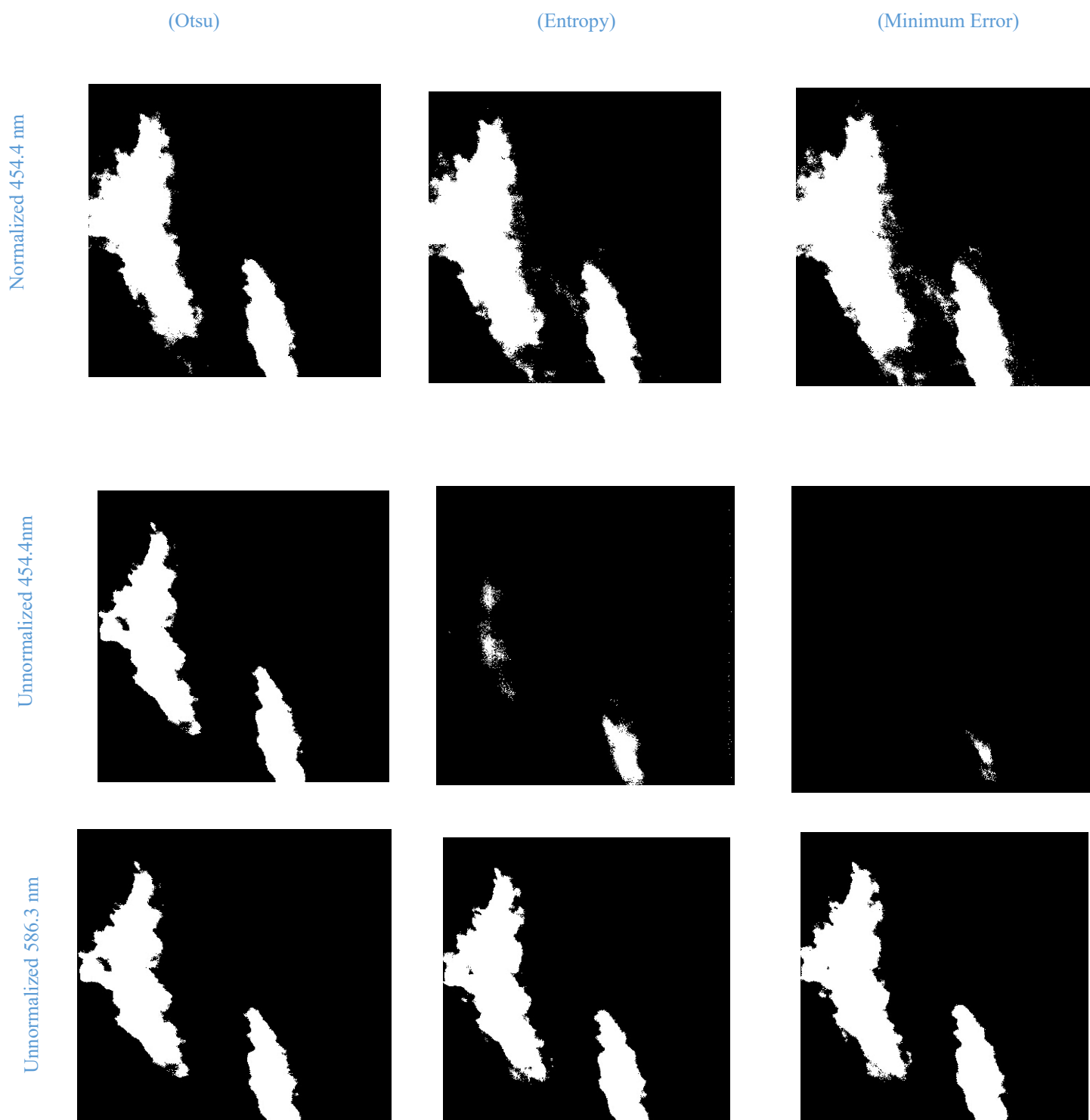


(Unnormalized Histogram Data) 586.3 nm

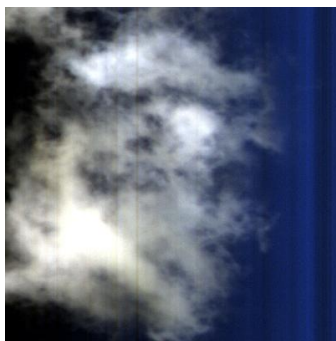


(Unnormalized Histogram Data) 454.4nm

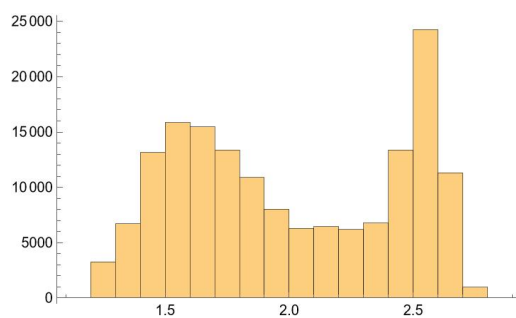
Appendix A: Scan_07-18-2022_0931_AZ075_EL20_L_D_R5 Data



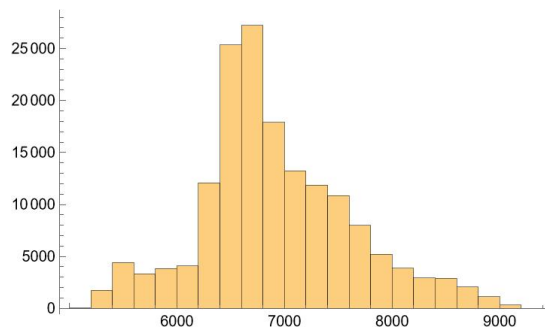
Appendix B: Scan_07-18-2022_0938_AZ315_EL30_L_D-R1 Data



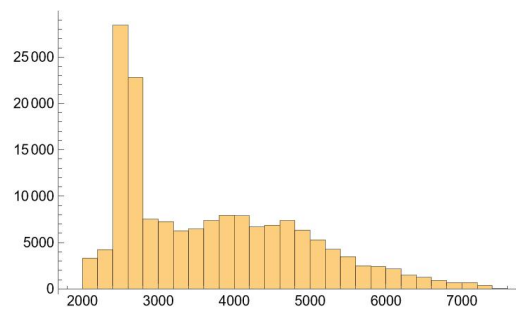
(Original Data Cube Image)



(Normalized Histogram Data) 454.4 nm

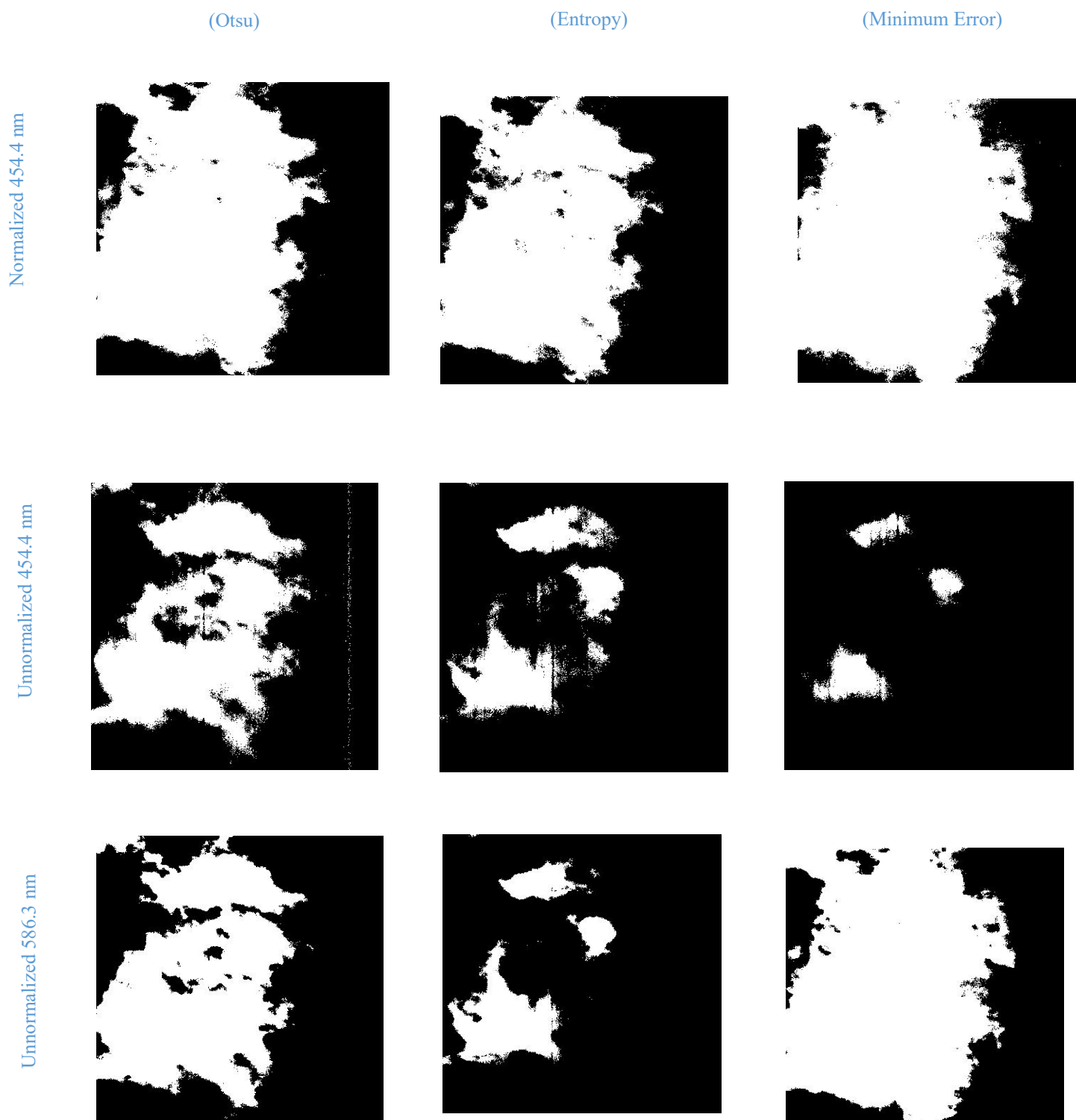


(Unnormalized Histogram Data) 454.4 nm



(Unnormalized Histogram Data) 586.3 nm

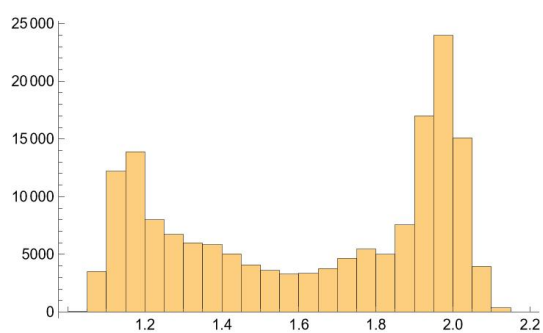
Appendix B: Scan_07-18-2022_0938_AZ315_EL30_L_D-R1 Data



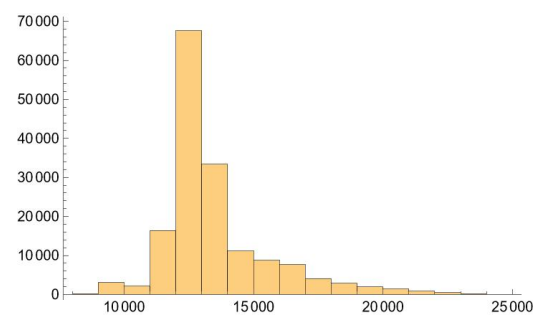
Appendix C: Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Data at 454.4 nm



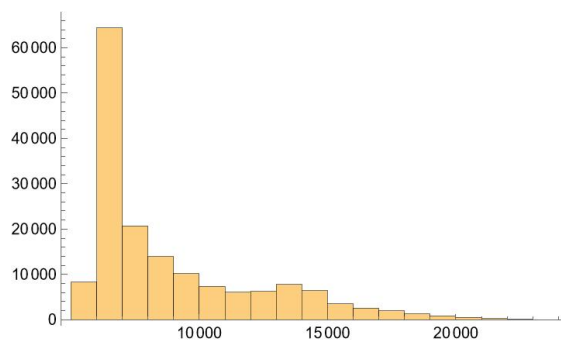
(Original Data Cube Image)



(Normalized Histogram Data) 454.4 nm

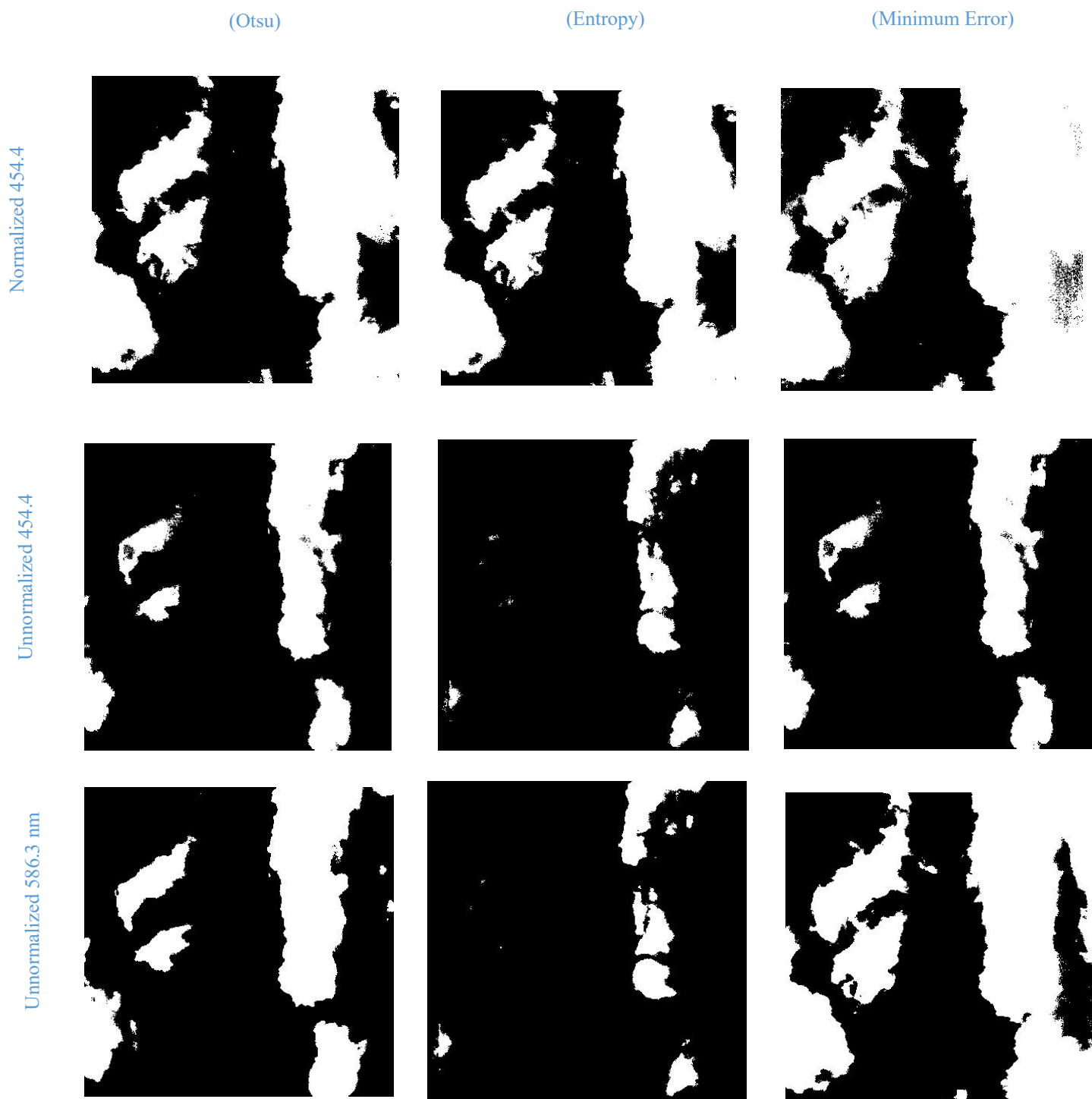


(Unnormalized Histogram Data) 454.4nm

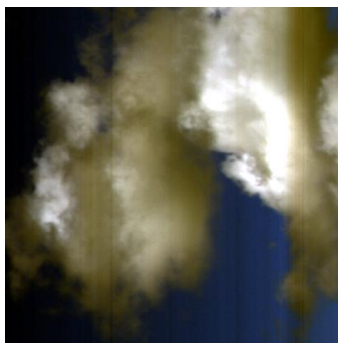


(Unnormalized Histogram Data) 586.3 nm

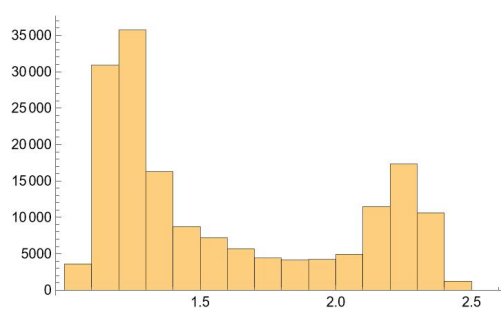
Appendix C: Scan_07-18-2022_0946_AZ075_EL20_L_D-R1 Data



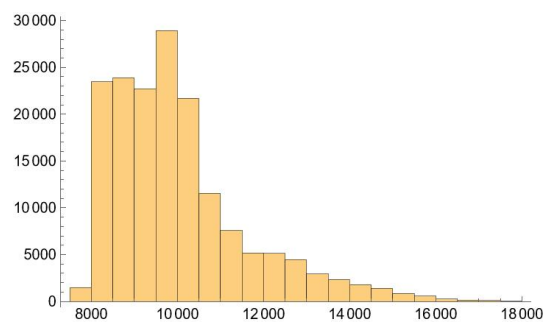
Appendix D: Scan_07-18-2022_0951_AZ225_EL20_L_D-R1 Data at 454.4 nm



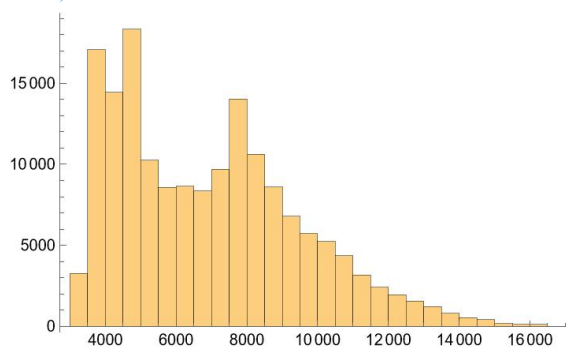
(Original Data Cube Image)



(Normalized Histogram Data)

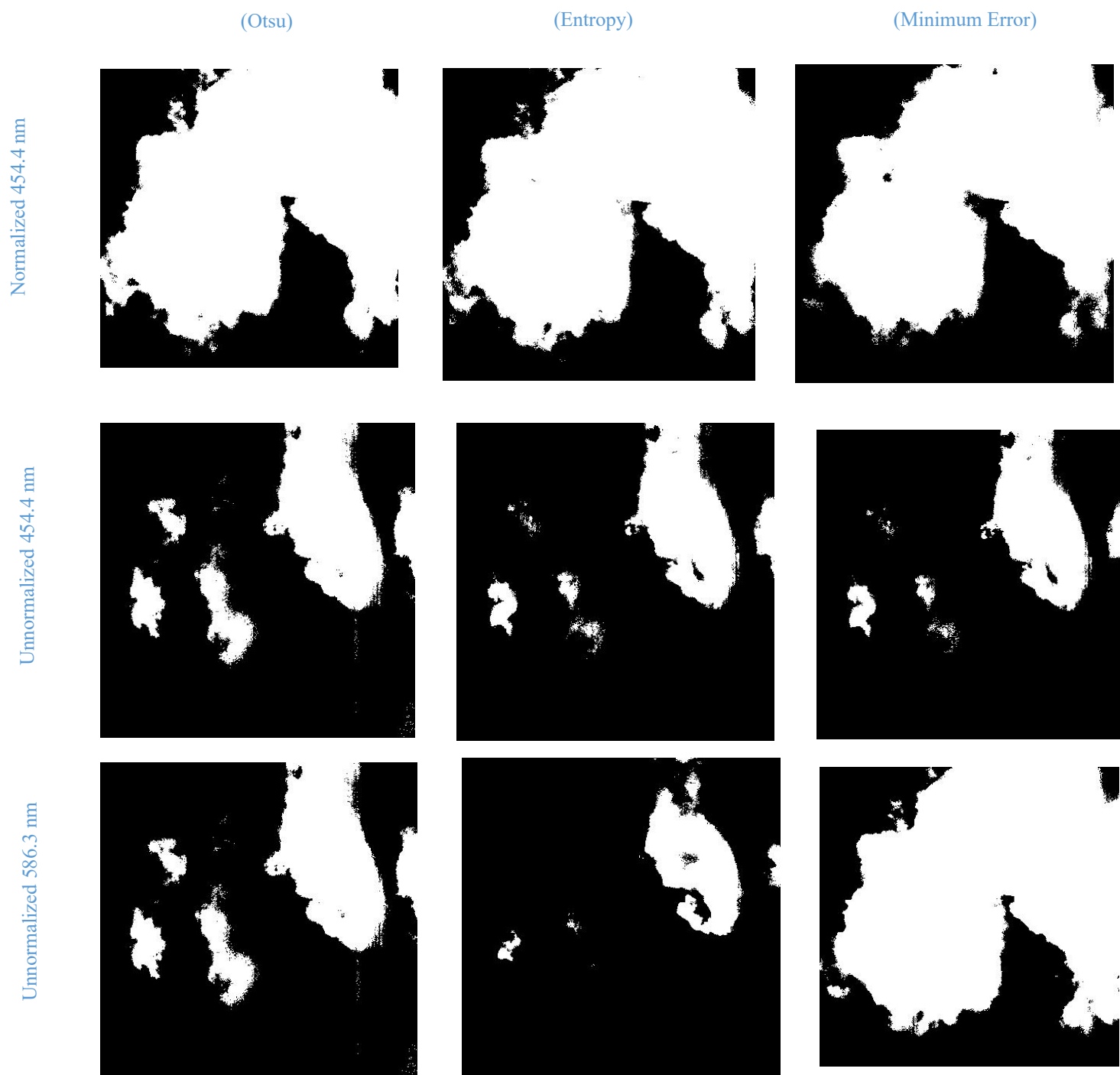


(Unnormalized Histogram Data)

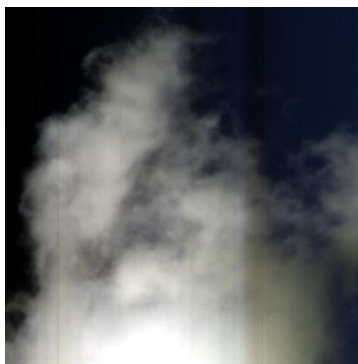


(Unnormalized Histogram Data) 586.3 nm

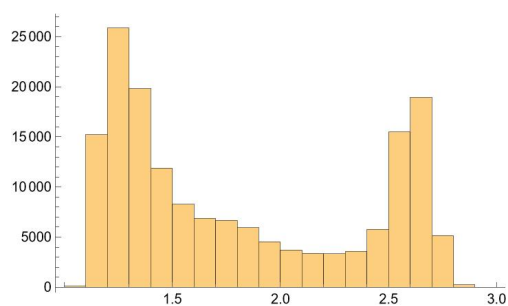
Appendix D: Scan_07-18-2022_0951_AZ225_EL20_L_D-R1 Data at 454.4 nm



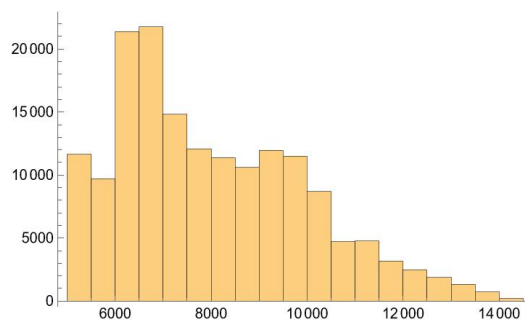
Appendix E: Scan_07-18-2022_0958_AZ270_EL35_L_D-R1 Data at 454.4 nm



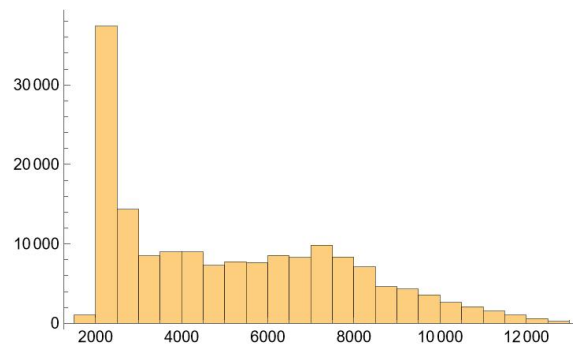
(Original Data Cube Image)



(Normalized Histogram Data)

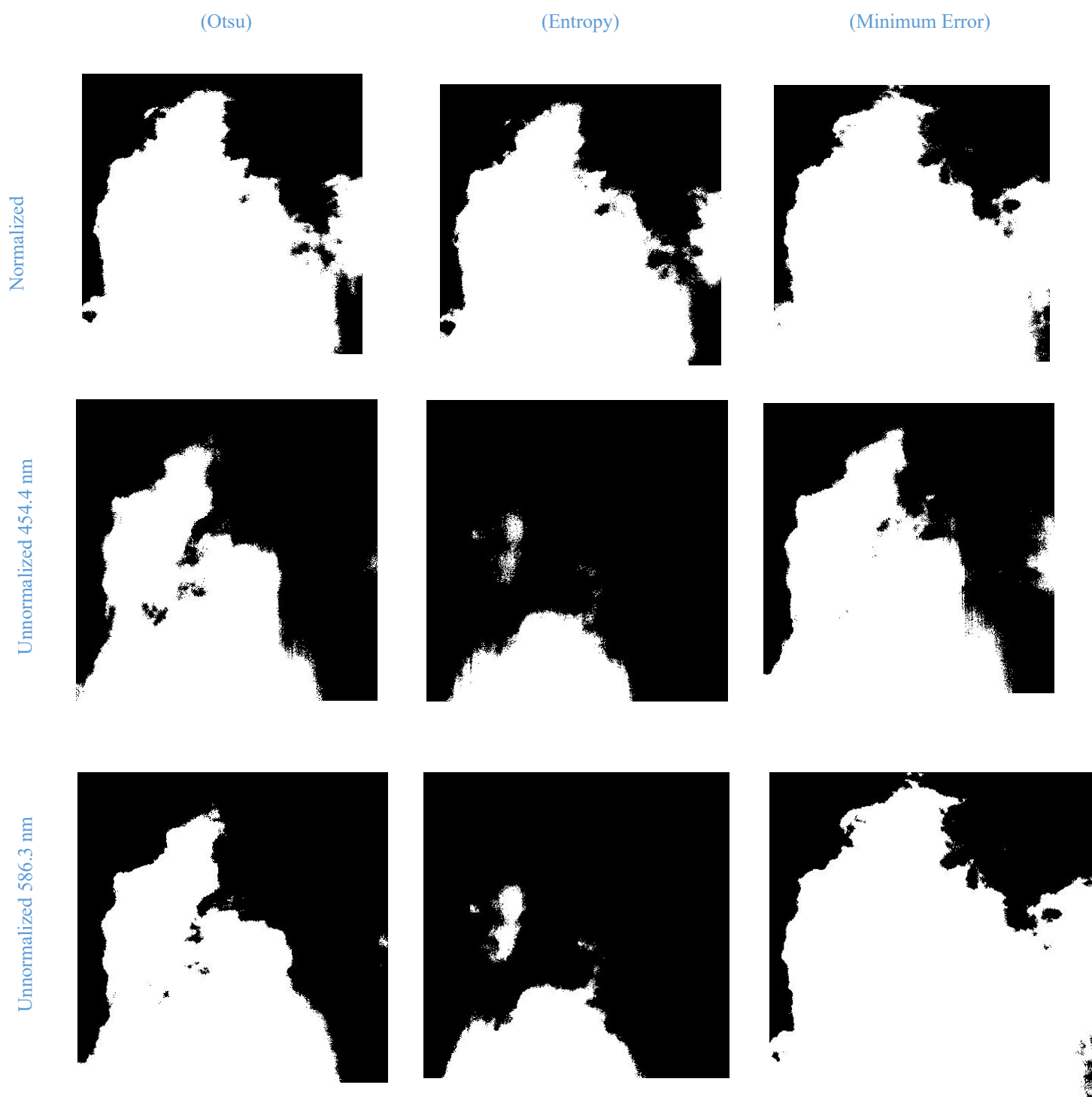


(Unnormalized Histogram Data)



(Unnormalized Histogram Data) 586.3 nm

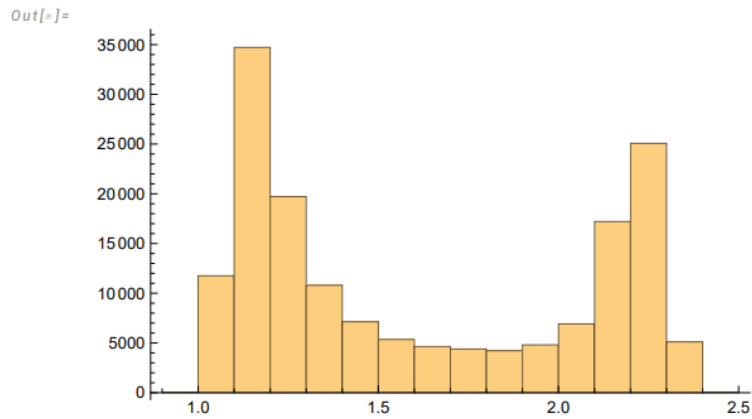
Appendix E: Scan_07-18-2022_0958_AZ270_EL35_L_D-R1 Data at 454.4 nm



Appendix F: Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

```
In[ ]:= SetDirectory[
  "e:\\NSF Research Data\\Machine Learning Experiment\\Data\\07-18-2022_0931-R4"];
data = Import["Scan_07-18-2022_0931_AZ075_EL20_L_D-R4-454p4nm.bip",
  "Binary", "DataFormat" -> "UnsignedInteger16"];
norm = Import["Scan_07-18-2022_0931_AZ075_EL20_L_D-R4-586p3nm.bip",
  "Binary", "DataFormat" -> "UnsignedInteger16"];
```

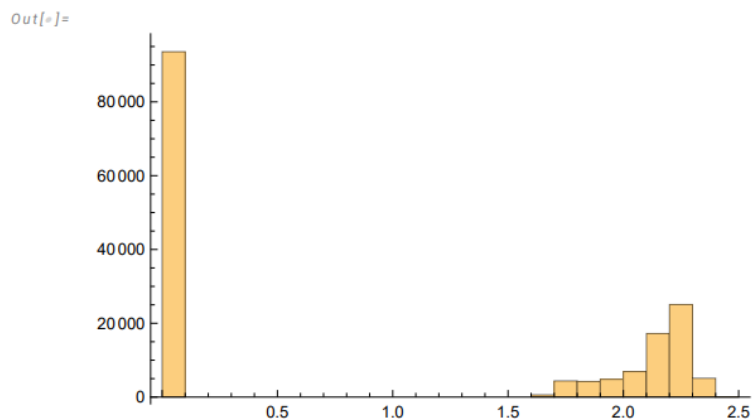
```
In[ ]:= npix = Length[data];
normdata = Table[data[[i]] / norm[[i]], {i, 1, npix}];
Histogram[normdata, 20]
```



```
In[ ]:= Export["Normalized Histogram 454p4nm.jpg", Histogram[normdata, 20]];
```

```
In[ ]:= otsuthreshold = FindThreshold[normdata]
threshdata = Threshold[normdata, otsuthreshold];
Histogram[threshdata, 20]
```

Out[]:=
1.68604



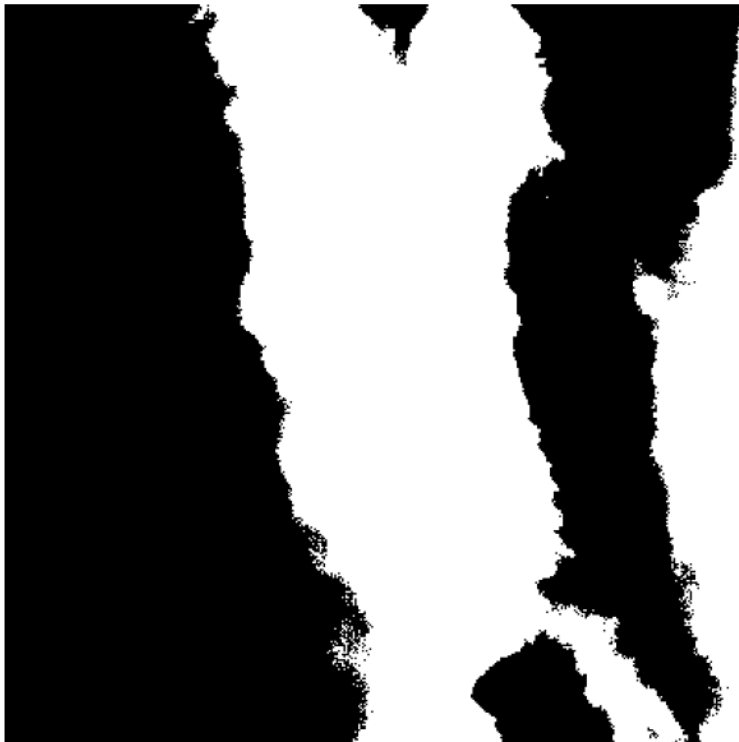
Appendix F : Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

```
In[ ]:= threshdatahistlist = HistogramList[threshdata];
cldpix = threshdatahistlist[[2, 1]];
clrpix = Total[threshdatahistlist[[2]]] - cldpix;
cldfrac = cldpix / (cldpix + clrpix);
N[cldfrac]
```

```
Out[ ]:=
0.57767
```

```
In[ ]:= nsamp = 402;
nline = 403;
imgdata = Table[normdata[Table[j, {j, (i - 1) * nsamp + 1, i * nsamp}]], {i, 1, nline}];
In[ ]:= binimgdata = ImageData[Binarize[Image[imgdata], otsuthreshold]];
binimg = Image[binimgdata]
```

```
Out[ ]:=
```



Appendix F : Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

```
In[ ]:= segimage = ColorNegate[bining]
```

```
Out[ ]=
```



```
In[ ]:= Export["otsu segmented image.jpeg", segimage]
```

```
Out[ ]=
```

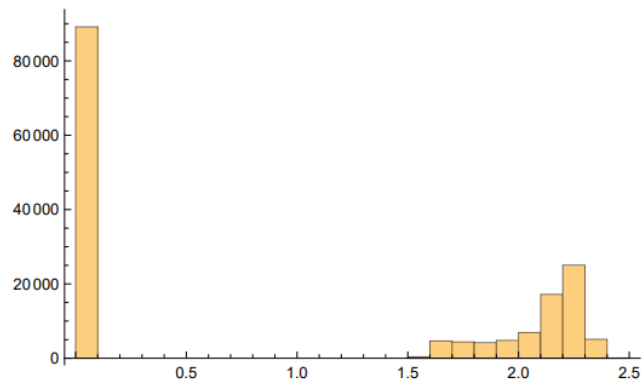
```
otsu segmented image.jpeg
```

```
In[ ]:= entropythreshold = FindThreshold[normdata, Method -> "Entropy"]  
threshdata = Threshold[normdata, entropythreshold];  
Histogram[threshdata, 20]
```

```
Out[ ]=
```

```
1.59265
```

```
Out[ ]=
```



Appendix F : Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

Out[]=*



In[]:=* `Export["entropy segmented image.jpeg", segimage]`

Out[]=*

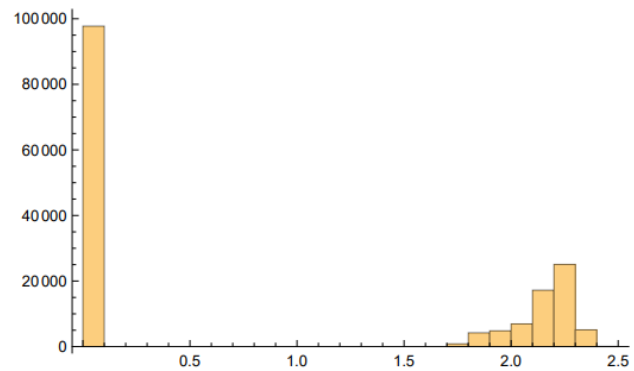
entropy segmented image.jpeg

In[]:=* `minerrthreshold = FindThreshold[normdata, Method -> "MinimumError"]`
`threshdata = Threshold[normdata, minerrthreshold];`
`Histogram[threshdata, 20]`

Out[]=*

1.77942

Out[]=*



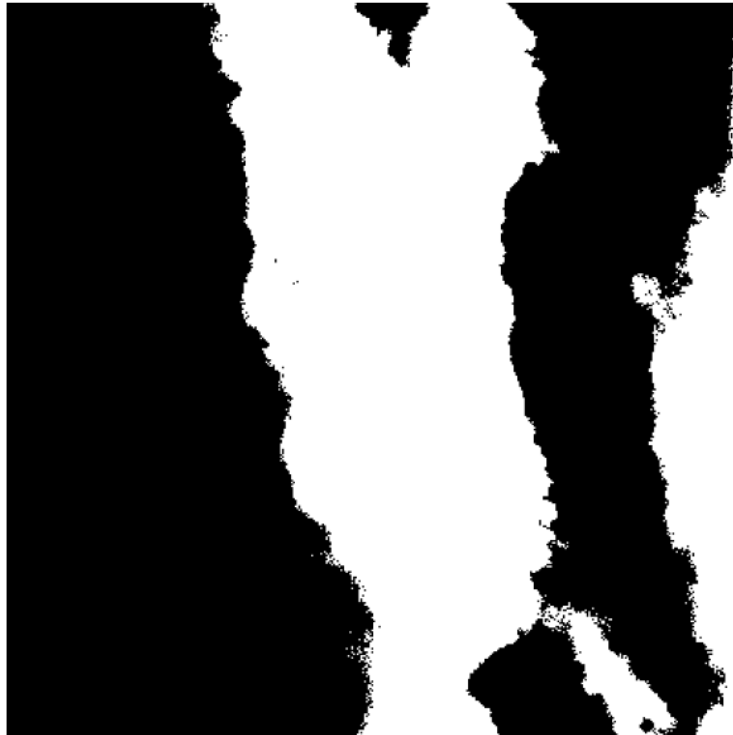
Appendix F : Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

```
In[ ]:= threshdatahistlist = HistogramList[threshdata];  
cldpix = threshdatahistlist[[2, 1]];  
clrpix = Total[threshdatahistlist[[2]]] - cldpix;  
cldfrac = cldpix / (cldpix + clrpix);  
N[cldfrac]
```

```
Out[ ]:=  
0.60336
```

```
In[ ]:= binimgdata = ImageData[Binarize[Image[imgdata], minerrthreshold]];  
binimg = Image[binimgdata]  
segimage = ColorNegate[binimg]
```

```
Out[ ]:=
```



Appendix F : Mathematica Software Code Used for Outputting Normalized Histogram Data and Segmented Images

Out[]=



```
In[ ]:= Export["minerr segmented image.jpeg", segimage]
```

Out[]=

```
minerr segmented image.jpeg
```