

3-26-2015

Hyperspectral Based Skin Detection for Person of Interest Identification

Khoa A. Tang

Follow this and additional works at: <https://scholar.afit.edu/etd>

Recommended Citation

Tang, Khoa A., "Hyperspectral Based Skin Detection for Person of Interest Identification" (2015). *Theses and Dissertations*. 62.
<https://scholar.afit.edu/etd/62>

This Thesis is brought to you for free and open access by the Student Graduate Works at AFIT Scholar. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of AFIT Scholar. For more information, please contact richard.mansfield@afit.edu.



**HYPERSPECTRAL BASED SKIN DETECTION FOR PERSON OF INTEREST
IDENTIFICATION**

THESIS

Khoa A. Tang, Captain, USAF

AFIT-ENG-MS-15-M-031

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY**

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, the Department of Defense, or the United States Government.

This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

AFIT-ENG-MS-15-M-031

HYPERSPECTRAL BASED SKIN DETECTION FOR PERSON OF INTEREST
IDENTIFICATION

THESIS

Presented to the Faculty
Department of Electrical and Computer Engineering
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Electrical Engineering

Khoa A. Tang, B.S.E.E.

Captain, USAF

March 2015

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AFIT-ENG-MS-15-M-031

HYPERSPECTRAL BASED SKIN DETECTION FOR PERSON OF INTEREST
IDENTIFICATION

Khoa A. Tang, B.S.E.E.
Captain, USAF

Committee Membership:

Lt Col Jeffrey D. Clark, PhD
Chair

Maj Brian G. Woolley, PhD
Member

Richard K. Martin, PhD
Member

Abstract

An optimal skin identification system can have great impact in areas such as security and surveillance. In the field of security, skin identification can augment the effectiveness of other biometric security systems such as facial recognition and fingerprint identification [1]. In the case of surveillance, a real-time skin identification system will be extremely useful at tracking person of interest (POI). Detecting and tracking using a skin identification technique provides positive, real time acquisition of people as they exit a building and prevents inadvertent track loss in a large crowd. This thesis presents the results of an artificial neural network (ANN) that is created in MATLAB[®] using the Neural Network Toolbox to identify a POI based on their skin spectral data. A baseline model is used with the optimal feature set identified by Cain [2]. The baseline model is then modified and optimized to maximize a ANN capability to identify a POI. Gaussian noise is calculated and added to the data sets to simulate atmospheric noise of a real world scene. The simulated atmospheric noise reduced the neural network's accuracy by 14%. The neural network model is tested with real Hyperspectral imaging (HSI) images to verify the applicability of the ANN to identify a POI. The results for HSI testing are between 40-60% due to the illumination sources' angle of incidence and the standard deviation of skin reflectance associated with differing skin locations on the body.

Acknowledgements

A big thanks to my adviser, Lt Col Jeffrey Clark, for all your help and support. You made the thesis writing process less boring with your comments on revision. Thank you for putting up with my bad writing and grammars. I like to also thanks James Arneal and Shane Fernandes for being in the lab helping me and keeping me slightly sane while I work on this thesis.

Khoa A. Tang

Table of Contents

	Page
Abstract	iv
Acknowledgements	v
Table of Contents	vi
List of Figures	ix
List of Tables	xiii
List of Acronyms	xviii
I. Introduction	1
1.1 Background	2
1.2 Problem Statement	2
1.3 Justification	3
1.4 Assumptions	3
1.5 Scope	4
1.6 Approach	4
1.7 Materials and Equipment	4
1.8 Organization	5
II. Background	6
2.1 Hyperspectral Imaging	6
2.2 Feature Selection	6
2.2.1 Information Theory	7
2.2.2 Entropy	8
2.2.3 Mutual Information	8
2.2.4 Fast Correlation Based Filter	8
2.3 Feature Extraction	9
2.4 Supervised Learning	9
2.4.1 Classification	10
2.4.2 Multilayer Perceptron	10
2.4.2.1 Stochastic Gradient Descent Back-Propagation	10
2.4.2.2 Gradient Descent	12
2.4.2.3 Levenberg-Marquardt	12

	Page
2.4.3 Support Vector Machine	14
2.5 Skin	15
2.5.1 Melanin	16
2.5.2 Carotene	16
2.5.3 Hemoglobin	16
2.5.4 Bilirubin	17
2.5.5 Fitzpatrick Scale	17
2.6 Normalization	17
2.7 Image Registration	18
2.8 Related Works	20
III. Methodology	21
3.1 Introduction	21
3.2 Materials and Equipment	21
3.3 Data	24
3.4 Baseline Model Testing	25
3.5 Optimizing Model	26
3.6 Support Vector Machine	28
3.7 Testing New Model With Other Skin Tone Groups	29
3.8 Noise Testing	30
3.9 HSI Testing	30
IV. Results and Analysis	34
4.1 Model Testing	34
4.2 Parameter Testing	38
4.3 Group Testing	44
4.4 Simulated Noise	46
4.5 HSI Testing	49
V. Conclusion	57
5.1 Introduction	57
5.2 Summary of Results	57
5.3 Future Work	60
Appendix A: Model Optimization Results: Normalization	61
Appendix B: Model Optimization Results: Activation Function	64

	Page
Appendix C: Model Optimization Results: Training Function	77
Appendix D: MLP and SVM Results for Testing All Three Groups	79
Appendix E: MLP Results of Testing with Gaussian Noise	83
Appendix F: HSI Results	86
Bibliography	90

List of Figures

Figure	Page
2.1 Hyperspectral Cube: The colors underneath the image are the different layers. Each layer represents a wavelength range or band of the electromagnetic spectrum.	7
2.2 A multilayer perceptron network with a m -dimensional input and n -dimensional output. The i are the inputs and the o are the outputs. The a , b , and z are the hidden layers with p , q , and r hidden nodes in each hidden layer. The w are the weight associated with each input and output of each hidden layer node. There are also bias weights associated with each hidden node.	11
3.1 FieldSpec [®] 3 with contact probe and ThinkPad. The contact probe is connected to the FieldSpec [®] with the optical cable and the power cable. The power cable power the built in illumination source and the optical cable collect and transmit the electromagnetic energy to the FieldSpec [®] 3. The FieldSpec [®] 3 is used to process, analyze, and interpolate the data. The data cable transmits the processed data from the FieldSpec to the RS^3 software in the ThinkPad.	21
3.2 FieldSpec [®] 3 skin sample of the forearm.	23
3.3 ASD skin sample with Gaussian noise. The red line is the original clean skin sample and the blue line is the skin sample with Gaussian noise applied.	31
3.4 The test image is at the top and the truth image is the bottom. In the test image, the POI is outlined in green and the non-POIs are outlined in red. In the truth image, the white pixels are the POI and the green pixels are the non-POIs. There are 91 POI pixels and 357 non-POIs pixels in the truth image.	32

Figure	Page
3.5 Training Images. The POI is outlined in green and the non-POIs are outlined in red. Two non-POIs were not used due to the difficulty in distinguishing their skin pixels from their shirt or shadow pixels.	33
4.1 Distribution of all classes in Group 1. The vertical dashed line denotes the 685 nm wavelength used to divide the samples into their skin tone groups. The vertical solid lines indicate the features (1024 nm, 1014 nm, 1033 nm, and 1348 nm). Group one has the melanin concentration distribution of 8% or less. Each color line is a sample of a class.	35
4.2 Distribution of all classes in Group 2 and 3. The vertical dashed line denotes the 685 nm wavelength used to divide the samples into their skin tone groups. The vertical solid lines indicate the features (1024 nm, 1014 nm, 1033 nm, and 1348 nm). Group two has the melanin concentration distribution between 8% and 16%. Group three has the melanin concentration distribution of 16% or more. Each color line is a sample of a class.	36
4.3 The standard deviation for the noise calculated from the spectralon white panel in the HSI image taken during the Minor Area Motion Imagery (MAMI) data collect in 2013. The standard deviation is calculated for each of the waveband taken by the AisaDUAL. The increase in noise in the 1000nm range is due to the transitioning between the AisaEAGLE and AisaHAWK. The increase in noise in the 1300 nm and 1800 nm regions are due to the water in the atmosphere.	47
4.4 A layer in the HSI image at wave band 1007 nm. As can be seen in the figure, the ghosting effect indicates the transition from the AisaEAGLE to the AisaHAWK. This require image registration to correctly align the two sensors. .	48

Figure	Page
4.5 This shows the spectral data of a non-noisy spectralon white panel with Gaussian noise added, and the spectral data of the spectralon white panel from the HSI image. The red line is the spectralon white panel from the HSI image and the blue line is the non-noisy spectralon white panel with Gaussian noise added.	49
4.6 Training Images. The POI is outline in green and the non-POIs are outlined in red. Two non-POIs were not used due to the difficulty in distinguishing their skin pixels from their shirt or shadow pixels.	51
4.7 The test image is at the top and the truth image is the bottom. In the test image, the POI is outline in green and the non-POIs are outlined in red. In the truth image, the white pixels are the POI and the green pixels are the non-POIs. There are 91 POI pixels and 357 non-POI pixels in the truth image.	52
4.8 Result from using the optimal feature set (1014 nm, 1024 nm, 1033 nm, and 1348nm) with the neural network. Identified POI pixels are colored in white and non-POI pixels are colored in light blue. Most of the POI pixels are correctly identified, however, most of the non-POIs are also incorrectly identified as the POI.	53
4.9 Result of neural network with the top four features (804 nm, 1137 nm, 891 nm, 1343 nm) in the new feature set.	54
4.10 (a) shows the samples of the POI and non-POIs used for training the network. The samples shown are not normalized. (b) shows samples of the POI and non-POIs after normalization using normalization method number 1.	55
4.11 Samples of the POI and non-POIs after normalization using normalization 1.	56
E.1 MLP results of training with non noisy data and testing with noisy data for Group 1	83

Figure	Page
E.2 MLP results of training with non noisy data and testing with noisy data for Group 2	83
E.3 MLP results of training with non noisy and testing with noisy data data for Group 3	84
E.4 MLP results of training and testing with noisy data for Group 1	84
E.5 MLP results of training and testing with noisy data for Group 2	84
E.6 MLP results of training and testing with noisy data for Group 3	85
F.1 Result of using the optimal feature set with l^2 norm.	86
F.2 Result of using the optimal feature set with normalization method 6	86
F.3 Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with normalization method 1.	87
F.4 Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with l^2 norm.	87
F.5 Result of using the optimal feature set with normalization 1 and a bigger network size of one hidden layer and ten hidden nodes.	88
F.6 Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with normalization method 1 network size of one hidden layer and ten hidden nodes.	88
F.7 Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm, 1285 nm, 1255 nm, 1564 nm, and 1695 nm) with normalization method 1 network size of one hidden layer and ten hidden nodes.	89

List of Tables

Table	Page
2.1 [3] Fitzpatrick Scale is used to classify a person into a skin type based on how he/she scores on a series of questions. Questions are about a person’s genetic disposition and reaction to extended sun exposure. Each answer to a question has a score associated to it. Total score is added up from all the questions to determine the skin type.	18
2.2 Normalization methods investigated in this thesis.	19
3.1 Skin tone group and their corresponding melanin concentration	24
3.2 Baseline Model Parameters	25
3.3 Operating Characteristic Values	27
3.4 support vector machine (SVM) Kernels that are investigated. These are the kernels that are available for the <i>svmtrain</i> in MATLAB®. The kernel is specified with the <i>kernel_function</i> parameter.	29
3.5 Final Model	29
4.1 Accuracy of the five fold cross validation for the first four POIs of group one using the baseline model. The average is calculated from averaging the results of the five folds.	37
4.2 Generalized Accuracy: the generalized accuracy is obtained by using all the data in the training set of group one to train the neural network and test with the testing set.	37
4.3 Operating Characteristic values for POIs 1-4 for Table 4.2.	38
4.4 Top two normalization method results from testing different normalizations with one or two hidden layers. All results are calculated using Equal Weighted Accuracy.	39

Table	Page
4.5 Top two normalization method results from testing different normalizations with three hidden layers. All results are calculated using Equal Weighted Accuracy.	39
4.6 Top two normalization method results from testing different normalizations with four hidden layers. All results are calculated using Equal Weighted Accuracy.	40
4.7 Top two normalization method results from testing different normalizations with five hidden layers. All results are calculated using Equal Weighted Accuracy.	40
4.8 Top two activation functions as determined by normalization and multilayer perceptron (MLP) topology. The activation functions <i>radbas</i> and <i>tribas</i> are radial basis and triangular basis respectively. The results here are only from MLP topology with one hidden layer. Appendix A contains results for all other activation functions and hidden layers. All results are calculated Equal Weighted Accuracy generalized results.	42
4.9 Top two activation functions as determined by normalization and MLP topology. The activation functions <i>radbas</i> and <i>tribas</i> are radial basis and triangular basis respectively. The results here are only from MLP topology with one hidden layer. Appendix A contains results for all other activation functions and hidden layers. All results are calculated Equal Weighted Accuracy generalized results.	42
4.10 Results from testing different training functions. All results are Equal Weighted Accuracy calculated from using all data in the training set for training.	43
4.12 Accuracy of the five fold cross validation for the first four POI of group one. . .	44

Table	Page
4.11 Final model from testing the various normalization methods, activation functions, training functions, and topologies available in MATLAB®. Final normalization method, activation function, and training function were determined by the best average accuracy from varying the topology.	45
4.13 SVM Results with polynomial kernel for the first four POIs in group 1. The results are for the five fold cross validation. All results are calculated with Equal Weighted Accuracy.	45
4.14 Classification results of training with non noisy data for Group 1	50
4.15 Classification results of training with noisy data for Group 1	50
A.1 Normalization methods investigated in this thesis.	61
A.2 MLP results from testing different normalizations with one and two hidden layers. All results are calculated using Equal Weighted Accuracy.	62
A.3 MLP results from testing different normalizations with three hidden layers. All results are calculated using Equal Weighted Accuracy.	62
A.4 MLP results from testing different normalizations with four hidden layers. All results are calculated using Equal Weighted Accuracy.	63
A.5 MLP results from testing different normalizations with five hidden layers. All results are calculated using Equal Weighted Accuracy.	63
B.1 MLP results for testing different activation functions with the top three normalization methods and one hidden layer	64
B.2 MLP results for testing different activation functions with the top three normalization methods and one hidden layers	65
B.3 MLP results for testing different activation functions with the top three normalization methods and two hidden layers	66

Table	Page
B.4 MLP results for testing different activation functions with the top three normalization methods and two hidden layers	67
B.5 MLP results for testing different activation functions with the top three normalization methods and three hidden layers	68
B.6 MLP results for testing different activation functions with the top three normalization methods and three hidden layers	69
B.7 MLP results for testing different activation functions with the top three normalization methods and three hidden layers	70
B.8 MLP results for testing different activation functions with the top three normalization methods and four hidden layers	71
B.9 MLP results for testing different activation functions with the top three normalization methods and four hidden layers	72
B.10 MLP results for testing different activation functions with the top three normalization methods and four hidden layers	73
B.11 MLP results for testing different activation functions with the top three normalization methods and five hidden layers	74
B.12 MLP results for testing different activation functions with the top three normalization methods and five hidden layers	75
B.13 MLP results for testing different activation functions with the top three normalization methods and five hidden layers	76
C.1 MLP results from testing different training functions with one and two hidden layers. All results are calculated using Equal Weighted Accuracy.	77
C.2 MLP results from testing different training functions with three hidden layers. All results are calculated using Equal Weighted Accuracy.	77

Table	Page
C.3 MLP results from testing different training functions with four hidden layers. All results are calculated using Equal Weighted Accuracy.	78
C.4 MLP results from testing different training functions with five hidden layers. All results are calculated using Equal Weighted Accuracy.	78
D.1 Neural network results for the first four POI of group one with the optimized configuration.	79
D.2 Neural network results for the first four POI of group two with the optimized configuration.	79
D.3 Neural network results for the first four POI of group three with the optimized configuration.	80
D.4 SVM Results for Group 1 with Linear Kernel	80
D.5 SVM Results for Group 2 with Linear Kernel	80
D.6 SVM Results for Group 3 with Linear Kernel	80
D.7 SVM Results for Group 1 with Radial Basis Function Kernel	81
D.8 SVM Results for Group 2 with Radial Basis Function Kernel	81
D.9 SVM Results for Group 3 with Radial Basis Function Kernel	81
D.10 SVM Results for Group 1 with Polynomial Kernel	81
D.11 SVM Results for Group 2 with Polynomial Kernel	82
D.12 SVM Results for Group 3 with Polynomial Kernel	82

List of Acronyms

Acronym	Definition
HSI	Hyperspectral imaging
SWIR	short-wave infrared
VIS	visible spectrum
PCA	principal component analysis
FCBF	Fast Correlation-Based Filter
POI	person of interest
ANN	artificial neural network
SVM	support vector machine
LM	Levenberg-Marquardt
EWA	Equal Weighted Average
SNR	Signal-to-Noise Ratio
ASD	Analytical Spectral Devices
MAMI	Minor Area Motion Imagery
NIST	National Institute of Standards and Technology
MLP	multilayer perceptron
UV	ultraviolet
RFE	recursive feature elimination
VNIR	visible and near-infrared
AFRL	Air Force Research Laboratory
ASCII	American Standard Code for Information Interchange

HYPERSPECTRAL BASED SKIN DETECTION FOR PERSON OF INTEREST IDENTIFICATION

I. Introduction

The human brain is an intricate organ that has developed unique capabilities and functions. One such unique function is object cognition. Object cognition is the ability to recognize and identify an object based on its physical properties [4]. Given an image that contains over a dozen people, the human brain can identify a person of interest (POI) in seconds. However, if the POI is in a large crowd of thousands of people, the time it takes to identify the POI would increase dramatically. This problem is exacerbated when trying to identify a POI in a video.

Technologies have advanced tremendously over the past two decades, e.g. computers in 1990s could barely perform gigaflops whereas now they can perform teraflops [5, 6]. Automated identification has progressed to the point where people can be identified based on their fingerprints, voice, or retina [7]. However, these identification techniques require a POI to willingly allow their fingerprints or retina to be scanned. Requiring compliance from a POI, in order to identify them from a crowd, is problematic. To bypass the need for compliance, remote sensing can be utilized. Remote sensing uses passive collection techniques to acquire information on an object or person. Cain [2] demonstrated the possibility of identifying a POI based on skin spectral data. She identified an optimal feature subset to be used with the hyperspectral data she collected using a spectroradiometer. Cain's artificial neural network (ANN) model is used as a baseline model for the work of this thesis.

1.1 Background

Nunez et al. [8, 9] determined the wavelengths needed to detect human skin with Hyperspectral imaging (HSI). Recent research by Cain [2] showed that it is possible to distinguish a POI in a diverse group of people based on skin spectral data. Using Fast Correlation-Based Filter (FCBF), Cain determined a set of wavelengths that allow POI identification. The wavelengths take advantage of the unique properties of human skin.

The color of human skin is determined by specific chromophores. Chromophores are molecules that absorb specific wavelengths of the visible spectrum while reflecting the rest. The chromophores that determine the color in human skin are produced in the body while others are produced by plants [10, 11]. Varying levels of these chromophores give skin its varying shades from extremely fair to dark.

1.2 Problem Statement

This thesis builds upon the optimal feature set and methodology of identifying a POI. The problem that this research will address is:

Is the baseline model capable of identifying a POI within their skin tone group? Is the baseline model capable of identifying a POI in a Hyperspectral Image?

Initial testing to determine the optimal ANN model is done with non-noisy data collected using a contact probe of a spectroradiometer. Simulation of real world conditions is accomplished by adding Gaussian noise to the spectroradiometer data. Training and testing for each skin tone group are accomplished in MATLAB[®] with the Neural Network Toolbox. Different normalization techniques, network structures, training functions, and activation functions are explored to determine the optimal configuration of the neural network. HSI images are collected with the AisaDUAL hyperspectral sensor system and are tested in MATLAB[®].

1.3 Justification

An optimal skin identification system can have great impact in areas such as security and surveillance. In the field of security, skin identification can augment the effectiveness of other biometric security systems such as facial recognition and fingerprint identification [1]. Skin identification can act as a validation system to other biometric systems because of how easily these systems can be bypassed. In the case of surveillance, a real-time skin identification system would be extremely useful at tracking threats. This would allow tracking of threats as they enter and leave a building or prevent inadvertent track loss in a large crowd. Skin identification can also be used to locate a missing or injured POI in the case of search and rescue situations [12].

1.4 Assumptions

All data collected with the spectroradiometer incorporates a calibrated illumination source. The onboard calibrated illumination source eliminates the need to set up an external light source. Having a calibrated light source also eliminates the need to take into account the various lighting sources and angles. Varying light sources and angles create a variety of reflectance issues that is not incorporated in this thesis [13].

Skin pigmentation can be altered due to various conditions which change the spectral data. Skin spectral data changes over a long period of time based on a person's gender, age, and ethnicity [14]. However, there are certain conditions that could change the spectral data in a short time period e.g. sun tan lotions or oils, sunscreen lotions, and moisturizers. Sunscreen lotions and moisturizing lotions are designed to protect the skin and moisturize it. However, these products contain chemical compounds that absorb certain ranges of wavelengths that changes the spectral data of the skin. Between data collections, the skin is assumed to stay constant and not experience any condition that would change its spectral data. When testing with HSI data, only skin pixels are being used. This assumes that the HSI image is already processed with a skin detection algorithm.

1.5 Scope

This thesis will focus on expanding the application of Cain's research [2]. Cain's research focused on determining a skin identification model that identifies a POI in a diverse group of people with varying skin tones and different ethnicities. This thesis extensively tests Cain's model to determine its capability to identify a POI within their skin tone group. Improvements to Cain's model are investigated to create a more accurate skin identifier for POI's within a similar skin tone group. Further testing incorporates HSI images to determine practical capability on real world data.

1.6 Approach

The optimal feature subset that Cain identified is centric to this research. A new HSI data set is incorporated from the Minor Area Motion Imagery (MAMI) data and is parsed into skin tone groups based on the Fitzpatrick scale and the melanosomes level for training and testing the neural network [15]. Cain's model is the baseline to conduct testing of the different skin tone groups. Different parameters are investigated to improve the classification accuracy, i.e. training functions, network sizes, and normalization methods. Once the optimal configuration is determined, Gaussian noise is added to simulate atmospheric noise based on the typical noise level for standoff data collects. An evaluation is conducted with skin pixels from HSI images.

1.7 Materials and Equipment

Data for this thesis is collected with the ASD SpecField[®] 3 spectroradiometer using a contact probe. The hyperspectral images are collected using an AisaDUAL hyperspectral sensor system. Raw data from the ASD and AisaDUAL is collected and processed by *RS³*, ViewSpec Pro 6.0, and ENVI 5.0 (Classic) before they can be used in MATLAB[®]. Feature selection and neural network training and testing are accomplished in MATLAB[®]. The neural network is created and optimized using the MATLAB Neural Network

Toolbox. Feature selection is accomplished in MATLAB[®] using a third party script that imports the WEKA[®] feature selection library files [16].

1.8 Organization

Chapter II discusses properties of skin, feature selection, classifiers, and background on hyperspectral imaging. Chapter III explains the methodological process for all the training and testing. Chapters IV and V present the results, analysis, and future works.

II. Background

2.1 Hyperspectral Imaging

Hyperspectral imaging (HSI) collects and processes information across a wide range of wavelengths in the electromagnetic spectrum [17]. HSI wavelengths typically range from the visible spectrum (VIS) to the short-wave infrared (SWIR) spectrum (0.4-2.4 micrometers (μm)) [18]. A hyperspectral image or data cube is composed of multiple layers. Each layer represents a wavelength range in the electromagnetic spectrum [19]. Figure 2.1 is an example of a hyperspectral data cube where it has n_b spectral bands and $n_1 \times n_2$ pixels. The variables n_1 and n_2 define the spatial information while n_b defines the spectral information.

HSI was originally developed for remote sensing in mining and geology [19]. However, over the last two decades, HSI has been adopted into other fields such as ecology and surveillance. The increased use and adoption of HSI into other fields has led to an increase in research of HSI applications [17]. HSI collects the reflected radiance, which is the amount of electromagnetic energy incident on the detector determined by its field of view [19]. Reflectance is the ratio of light reflected from a scene to the light striking the object [19].

2.2 Feature Selection

Feature selection is the process of selecting a subset of relevant features from the original set of features based on a certain evaluation criterion [20]. Feature selection is differentiated into three main categories: filter, wrapper, and embedded method [21].

Filter methods are independent of learning algorithms. Filter methods rely on general characteristics of the data and an evaluation heuristic that is independent of the classifier. Filter methods are considered less computationally complex than wrapper and embedded

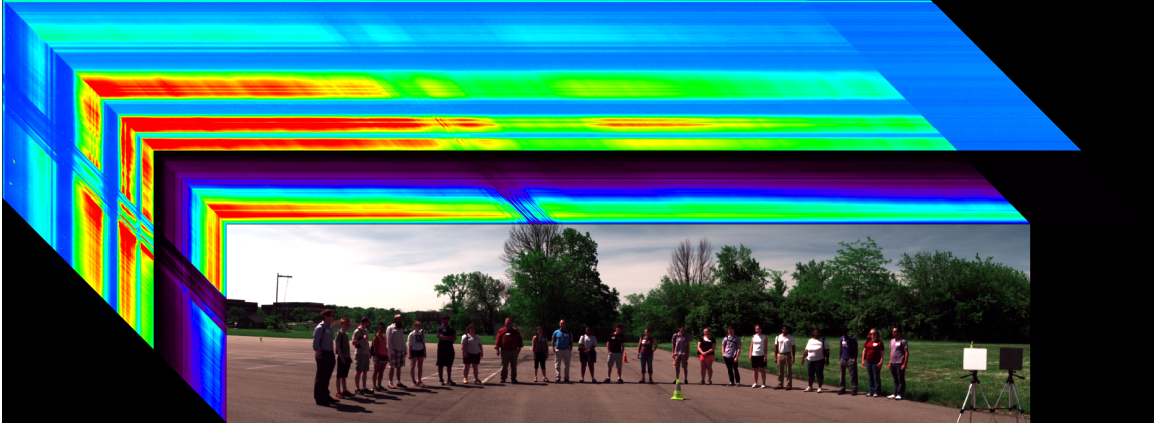


Figure 2.1: Hyperspectral Cube: The colors underneath the image are the different layers. Each layer represents a wavelength range or band of the electromagnetic spectrum.

methods. However, filter methods are computationally faster, but are considered less accurate than wrapper and embedded methods. [20]

Wrapper methods depend upon the classifier attributes. Wrapper methods have the advantage of ranking features based on how they perform with the chosen classifier characteristics. Wrapper methods generally have higher accuracy as compared to filter and embedded methods. However, as a trade off, they are computationally expensive. [20]

Embedded methods are encapsulated within a learning algorithm and is optimized specifically for that learning algorithm. One popular example of embedded method is the recursive feature elimination (RFE) algorithm. This algorithm is commonly used in support vector machines (SVMs) to recursively build a model and remove features. Embedded methods tend to be between filter and wrapper methods on computational complexity. [20]

2.2.1 Information Theory.

Information theory is a branch of mathematics that deals with the transmission, storing and processing of information [22]. In “A Mathematical Theory of Communication,”

Shannon defined two important quantities of information theory: entropy and mutual information [23].

2.2.2 Entropy.

Entropy is the most fundamental measure of information in information theory. Entropy measures the amount of uncertainty of a random variable [23]. The entropy of a random variable X is defined as [23]:

$$H(X) = - \sum_i P(x_i) \log_2(P(x_i)), \quad (2.1)$$

where $P(x_i)$ is the probability of $x = x_i$. When the logarithm in Equation 2.1 is in base 2, the information is measured in units of bits. In the case of two events, x and y , the joint entropy would be defined as [23]:

$$H(X, Y) = - \sum_j \sum_i P(x_i, y_j) \log_2(P(x_i, y_j)), \quad (2.2)$$

where $P(x_i, y_j)$ is the joint probability of x_i and y_j . Conditional entropy of X given Y , $H(X|Y)$, is defined as [23]:

$$\begin{aligned} H(X|Y) &= - \sum_j P(y_j) \sum_i P(x_i|y_j) \log_2(P(x_i|y_j)), \\ &= H(X, Y) - H(Y). \end{aligned} \quad (2.3)$$

2.2.3 Mutual Information.

Mutual information, or information gain, represents the reduction in uncertainty in X when Y is known [24]. Mutual information between two random variables, X and Y , is defined as [23]:

$$IG(X|Y) = H(X) - H(X|Y), \quad (2.4)$$

where $H(X)$ is the entropy of X and $H(X|Y)$ is the conditional entropy of X given Y .

2.2.4 Fast Correlation Based Filter.

Fast Correlation-Based Filter (FCBF) is a feature selection method that implements correlation between two random variables as a measure of goodness [25]. A feature is

“good” if it is highly correlated to the class but not to any other features. FCBF uses the information theory concept of entropy and mutual information to measure the correlation between two random variables instead of the linear correlation coefficient. Measuring goodness with information theory overcomes the shortcomings of the linear correlation coefficient.

Symmetrical Uncertainty is the weighted average of two uncertainty coefficients and is defined as [25]:

$$SU(X, Y) = 2 \left[\frac{IG(X|Y)}{H(X) + H(Y)} \right]. \quad (2.5)$$

Symmetrical Uncertainty is use as a measure for goodness in FCBF rather than IG to compensate for IG’s bias toward features with more values. SU also normalizes its value to 1 to indicate that X and Y are independent and 0 to indicate that X and Y are dependent [25].

2.3 Feature Extraction

Unlike feature selection, feature extraction transforms the data from high dimensional space to a lower dimensional space. Data transformation can be either linear or nonlinear. A common linear transformation technique is principal component analysis (PCA). PCA computes the eigenvectors and eigenvalues of a data matrix [26]. The eigenvectors are the principal components while the eigenvalues are their corresponding variances. PCA is set up in such a way that the first principal component is the one with the largest variance with the succeeding components as the second, third, fourth, etc. largest variance.

2.4 Supervised Learning

Supervised learning is the process of separating n -dimensional data into m classes using a discriminant function [27]. All data in supervised learning are labeled with their respective classes. The main objective of supervised learning is to find a discriminant function that will correctly classify future data. One of the most straightforward cases of

supervised learning is to separate n -dimensional vectors into two categories, which can be accomplished using a linear discriminant function.

2.4.1 Classification.

In machine learning, a classifier is a type of supervised learning [28]. All samples in a training data set are labeled with their correct classes. Some common classifiers are Naive Bayes, support vector machine (SVM), and Neural Networks. The classifiers used in this thesis is a multilayer perceptron and SVM.

2.4.2 Multilayer Perceptron.

A multilayer perceptron (MLP) is a neural network with one or more hidden layer [29]. Figure 2.2 is a directed graph of a MLP. Each neuron in the network includes a nonlinear activation function that is differentiable [30]. The training process for a MLP is split into two phases: the forward pass and backward pass. In the forward pass, the synaptic weights of the network are fixed and the input signal is propagated through the network, layer by layer, producing an output [30]. In the backward pass, the error is calculated from the outputs of the network, and is propagated back through the network, layer by layer. During the backward pass, the synaptic weights are adjusted based on the specific training method incorporated [30].

2.4.2.1 Stochastic Gradient Descent Back-Propagation.

Back-propagation is a popular training method for a MLP [30]. Back-propagation uses a stochastic gradient descent algorithm to update the weights. Stochastic gradient descent is an online learning algorithm, which updates the weights between each training sample [31]. The induced local field, $v_j(n)$, is calculated for each neuron using [30]:

$$v_j(n) = \sum_{i=0}^m w_{ji}(n)y_i(n), \quad (2.6)$$

where m is the total number of inputs, n is the iteration number, and w_{ji} is the weight of input i going into neuron j . Each neuron output is operated on by an activation function to

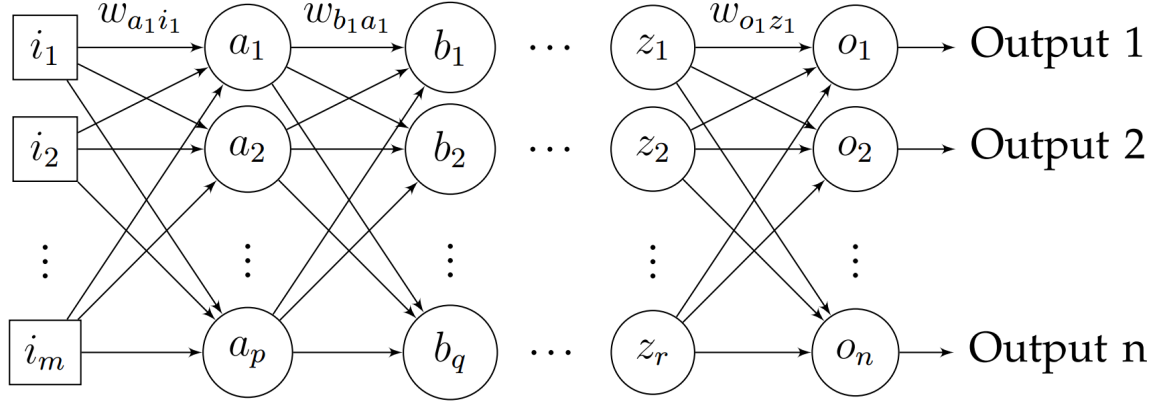


Figure 2.2: A multilayer perceptron network with a m -dimensional input and n -dimensional output. The i are the inputs and the o are the outputs. The a , b , and z are the hidden layers with p , q , and r hidden nodes in each hidden layer. The w are the weight associated with each input and output of each hidden layer node. There are also bias weights associated with each hidden node.

obtain the function signal $y_j(n)$ [30].

$$y_j(n) = \varphi_j(v_j(n)), \quad (2.7)$$

where $\varphi_j(*)$ is the activation function for neuron j . This is done for each neuron in the network, until the output layer is reached. The error signal is computed by [30]:

$$e_j(n) = d_j(n) - o_j(n), \quad (2.8)$$

where $d_j(n)$ is the desired response of neuron j and $o_j(n)$ is the output from the network at neuron j [30]. The local gradient \mathcal{L} is calculated for each neurons, depending on the neuron's location in the network, as shown [30]:

$$\mathcal{L}_j^{(l)}(n) = e_j^{(l)}(n) \varphi_j'(v_j^{(l)}(n)), \quad (2.9)$$

$$\mathcal{L}_j^{(l)}(n) = \varphi_j'(v_j^{(l)}(n)) \sum_k \mathcal{L}_k^{(l+1)}(n) w_{kj}^{(l+1)}(n), \quad (2.10)$$

where $\varphi'_j(\cdot)$ is the first order derivative of the activation function. Equation 2.9 calculates the gradient of neuron j of output layer L [30]. Equation 2.10 calculates the gradient of neuron j in a hidden layer l [30].

The Δw is calculated for each neuron and applied to their respective neuron weight as shown in Equation 2.11 [30].

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[\Delta w_{ji}^{(l)}(n-1)] - \eta \mathcal{L}_j^{(l)} y_i^{l-1}(n) \quad (2.11)$$

In Equation 2.11, η is the learning rate and α is the momentum constant [30]. The learning rate controls how much the weight and bias change and the momentum constant prevents the system from converging to a local minimum.

2.4.2.2 Gradient Descent.

Gradient descent applies adjustments to the weight vector in the direction opposite to the gradient vector. The gradient descent algorithm is described by

$$w(n+1) = w(n) - \eta g(n), \quad (2.12)$$

where η is the learning rate and $g(n)$ is the gradient vector evaluated at $w(n)$ [30]. $g(n)$ is defined as

$$g = \nabla \mathcal{E}(w) = \left[\frac{\partial \mathcal{E}}{\partial w_1}, \frac{\partial \mathcal{E}}{\partial w_2}, \dots, \frac{\partial \mathcal{E}}{\partial w_M} \right]^T, \quad (2.13)$$

where $\mathcal{E}(w)$ is the cost function [30]. Gradient descent is a batch learning algorithm, which updates the weights after the network processes a complete training data set [30].

2.4.2.3 Levenberg-Marquardt.

The Levenberg-Marquardt (LM) training method is a blend between the Gauss-Newton and gradient descent method [32]. LM is a batch learning algorithm that inherits the speed of the Gauss-Newton algorithm and the stability of the gradient descent method. The basic idea of the LM algorithm is that it switches between the Gauss-Newton and the gradient descent training method. The update rule for the LM algorithm is

$$w_{n+1} = w_n - (J_n^T J_n + \mu I)^{-1} J_n \mathbf{e}_n, \quad (2.14)$$

where μ is the combination coefficient, n is the iteration number, J is the Jacobian matrix, and I is the identity matrix [32]. The combination coefficient, μ , is what switches the algorithm between the Gauss-Newton and gradient descent method. When μ is small, Equation 2.14 essentially becomes a Gauss-Newton algorithm [33]. When μ is very large, Equation 2.14 approximates to the steepest descent algorithm [33]. The $J_n^T J_n + \mu I$ in Equation 2.14 approximates the Hessian matrix. The Hessian matrix is a second order derivative of the error function which is computationally complicated to calculate. The Jacobian matrix is defined as 2.15:

$$J = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_1} & \frac{\partial e_{1,1}}{\partial w_2} & \cdots & \frac{\partial e_{1,1}}{\partial w_N} \\ \frac{\partial e_{1,2}}{\partial w_1} & \frac{\partial e_{1,2}}{\partial w_2} & \cdots & \frac{\partial e_{1,2}}{\partial w_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_{1,M}}{\partial w_1} & \frac{\partial e_{1,M}}{\partial w_2} & \cdots & \frac{\partial e_{1,M}}{\partial w_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_{P,1}}{\partial w_1} & \frac{\partial e_{P,1}}{\partial w_2} & \cdots & \frac{\partial e_{P,1}}{\partial w_N} \\ \frac{\partial e_{P,2}}{\partial w_1} & \frac{\partial e_{P,2}}{\partial w_2} & \cdots & \frac{\partial e_{P,2}}{\partial w_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial e_{P,M}}{\partial w_1} & \frac{\partial e_{P,M}}{\partial w_2} & \cdots & \frac{\partial e_{P,M}}{\partial w_N} \end{bmatrix}, \quad (2.15)$$

where M is the number of outputs and P is the number of samples [32]. The \mathbf{e} in Equation 2.14 is the error vector and has the form of [32]

$$\mathbf{e} = \begin{bmatrix} e_{1,1} \\ e_{1,2} \\ \vdots \\ e_{1,M} \\ \vdots \\ e_{P,1} \\ e_{P,1} \\ \vdots \\ e_{P,M} \end{bmatrix}, \quad (2.16)$$

where $e_{P,M}$ is the error of sample P at output M .

2.4.3 Support Vector Machine.

SVM is a supervised learning algorithm that classifies data by constructing a hyperplane that maximizes the margin between two classes in the feature space [34, 35]. A hyperplane can be written as a set of points that satisfy

$$\mathbf{w} \cdot \mathbf{x} - b = 0, \quad (2.17)$$

where \mathbf{x} is the set of points, b is the offset, and \mathbf{w} is the normal vector to the hyperplane [34, 35]. The norm of \mathbf{w} , $\|\mathbf{w}\|$, must be minimized in order to maximize the margin (ρ) between the two classes which is define as $\rho = \frac{2}{\|\mathbf{w}\|}$ [34]. These margins are defined as

$$\mathbf{w} \cdot \mathbf{x}_i - b \geq 1, \quad (2.18)$$

for \mathbf{x}_i of the first class and

$$\mathbf{w} \cdot \mathbf{x}_i - b \leq -1, \quad (2.19)$$

for \mathbf{x}_i of the second class and i is the sample of the class [34, 35].

2.5 Skin

The skin, or integumentary system, covers the external body and accounts for about 7% of body weight [10, 11]. The skin consists of two main layers: the dermis and epidermis. The epidermis is the outer layer and is composed of epithelial cells. The dermis is the underlying layer and is mostly fibrous connective tissue. The epidermis consists of four distinct cell types: keratinocytes, melanocytes, Langerhans (dendritic) cells, and Merkel (tactile) cells. Keratinocytes make up 90% of the epidermal cells where their main role is to produce keratin. Melanocytes make up 8% of the epidermal cells and mainly produce Melanin. [11]

The epidermis is divided up into four to five layers depending on the thickness of the skin. Skin thickness varies based on location of the body, e.g. the palm, fingertips, and sole of the feet are thicker than other areas [10]. The five layers are stratum basale, stratum spinosum, stratum granulosum, stratum lucidum, and stratum corneum. The stratum lucidum is only present in thick skin. [11]

There are three chemical compounds that contribute to skin color: melanin, carotene, and hemoglobin. Melanin is produced in the skin by melanocytes and give the color ranging from reddish yellow to brownish black. Carotene is produced by plant products and ranges from yellow to orange. Hemoglobin is a metalloprotein that gives either a pinkish hue or a bluish color depending on the oxygenation of the blood. [10]

The skin also differs between genders as mentioned by Giacomoni et al. [14]. Giacomoni's paper is a compilation of various studies on the human skin to list the gender-linked differences in human skin. The gender-linked differences in skin include: metabolism and reaction to sex hormone, concentration of sebaceous glands, rate of sweating, skin pH, skin thickness, concentration of melanocytes, and rate at which the skin darkens when exposed to sunlight [14].

2.5.1 Melanin.

Melanin is produced by melanocytes in the skin using the amino acid tyrosine and an enzyme called tyrosinase [10, 11]. The synthesis occurs in a cytoplasmic organelle in melanocytes called a melanosome. Exposure to ultraviolet (UV) light increases production of melanin in the skin. Aside from contributing to skin color, melanin also protects the skin from UV light by absorbing the UV radiation. The synthesis of melanin is catalyzed by tyrosinase. In the presence of zinc, it produces eumelanin which gives the black to brown color range. Pheomelanin and trichromes are results from addition of cysteine to dopaquinone. Pheomelanin gives the yellow to reddish brown color. [11] Both eumelanin and pheomelanin absorb electromagnetic energy between at 400-700nm [36]. Different ethnicity groups have different melanin concentration producing a wide variety of skin colors [37].

2.5.2 Carotene.

Carotene is a precursor to vitamin A which is needed for synthesis of color in vision [10, 11]. Carotene produces a yellow-orange color like egg yolk and carrot. Carotene is stored in the stratum corneum and fatty areas of the hypodermis. It is commonly found in nature as beta-carotene. Other form of carotenes are cryptoxanthene, alpha-carotene, and gamma-carotene. Pigments from carotene are most obvious in the palm and the soles where the stratum corneum is thickest. [10] Carotene absorbs wavelengths at 400-500nm and is synthesized only by plants [38].

2.5.3 Hemoglobin.

Hemoglobin is an oxygen-carrying protein that is found in red blood cells or erythrocytes and gives blood the red color pigment [10, 11]. Each red blood cell contains about 280 million hemoglobin molecules. Each hemoglobin molecule is made up of a globin protein and nonprotein pigments called a heme. The globin protein is composed of four polypeptide chains: two alphas and two betas. A heme is bound to each of the

polypeptide chain. At the center of each heme is an iron ion that is capable of binding to one oxygen molecule. Hemoglobin carries oxygen from the lung to other parts of the body and carries carbon dioxide back to the lung to be exhaled. Oxygenated hemoglobin gives blood a bright red color while deoxygenated hemoglobin gives blood a dark red color. [10] Oxygenated hemoglobin absorbs wavelengths of 940nm and deoxygenated blood absorbs wavelengths at 660nm [39].

2.5.4 Bilirubin.

Bilirubin is a yellow-orange pigment and is the breakdown product of the heme [10, 11]. When the hemoglobin proteins are degraded, the polypeptides are hydrolyzed into amino acids while the heme groups are converted into bilirubin. Bilirubin is excreted into bile after it is coupled with glucuronic acid in the liver. When bile production reaches a certain concentration it is diffused into tissues and produces jaundice. Bile built up in the tissues gives the yellow coloration of the skin. [11]

2.5.5 Fitzpatrick Scale.

The Fitzpatrick Scale was developed in 1975 as way to classify skin type based on its response to ultraviolet light exposure [3]. There are six skin types in the Fitzpatrick Scale. On one end of the scale, Type I is the type that always burns when exposed to sunlight and never tans. At the other end of the scale, Type VI is the type that never burns and tans easily. A person is classified into a skin type based on how he/she scores on a series of questions about his/her genetic disposition and reaction to extended sun exposure. [40] Table 2.1 lists the skin types and scores based on the Fitzpatrick Scale.

2.6 Normalization

Data normalization is a technique used to reduce data to standard normal form. There are various way of normalizing data. Depending on the type of data, different normalizing techniques will work better than others. Normalization helps to eliminate any anomalies

Table 2.1: [3] Fitzpatrick Scale is used to classify a person into a skin type based on how he/she scores on a series of questions. Questions are about a person’s genetic disposition and reaction to extended sun exposure. Each answer to a question has a score associated to it. Total score is added up from all the questions to determine the skin type.

Type	Score	Characteristic
Type I	0-6	Pale white; blond or red hair; blue eyes; freckles. Always burns, never tans.
Type II	7-13	White; fair; blond or red hair; blue, green or hazel eyes. Usually burns, tans minimally.
Type III	14-20	Cream white; fair with any hair or eye color; quite common. Sometimes mild burn, tans uniformly.
Type IV	21-27	Moderate brown; typical Mediterranean skin tone. Rarely burns, always tans well.
Type V	28-34	Dark brown; Middle Eastern skin types. Very rarely burns, tans very easily.
Type VI	35+	Deeply pigmented dark brown to black. Never burns, tans very easily.

that might arise when collecting samples. For skin data, normalization will counteract the drift in the spectral measurement due to the skin heating up during data collection.

Table 2.2 lists the normalization methods investigated in this thesis.

2.7 Image Registration

Image registration is the process of aligning two or more images in a scene to overcome rotation, scaling, and skewing [41, 42]. Image registration is commonly used

Table 2.2: Normalization methods investigated in this thesis.

Normalization Method	Formula	Description
Max Samples	$\frac{\mathbf{X}}{X_{max}}$	Normalization by dividing the sample X by the maximum value of the sample.
Area Normalization	$\frac{\mathbf{X}}{X_{area}}$	Normalization by dividing the sample X by the area under the curve of the hyperspectral sample.
Unary Features	$\frac{X_f - f_{min}}{f_{max} - f_{min}}$	Normalization by bounding samples from 0 to 1 in each feature. X_f is the feature f in sample \mathbf{X} and f_{min} and f_{max} are the minimum and maximum values of that feature vector in the data set respectively.
Max Features	$\frac{X_f}{X_{fmax}}$	Normalization by dividing by the max value of each feature from the feature vectors of all samples.
Euclidean	$\frac{\mathbf{X}}{\ \mathbf{X}\ }$	Normalizes by dividing each sample by its L-2 norm.
Unary Samples	$\frac{\mathbf{X} - X_{min}}{X_{max} - X_{min}}$	Normalization by bounding each samples from 0 to 1 in each sample. X_{min} and X_{max} are the minimum and maximum values of the sample \mathbf{X} respectively.

with medical and satellite imagery when images are taken from different camera sources. For this thesis, image registration is needed to match up the information from the different sensors in the HSI camera. In order for the HSI camera to collect data in the VIS to the SWIR wavelength, it requires the use of multiple sensors. These sensors are positioned

together along the horizontal axis so rotation and scaling of the images are not needed during image registration. For this thesis, a standard edge detection technique was used for image registration.

2.8 Related Works

Previous works have shown that not only is it possible to detect skin with HSI, it is also possible to distinguish a POI within a diverse group [2, 8, 9, 43]. Nunez et al. first introduced a physics-based model to describe the reflectance of the human skin [9]. Nunez used the Kubelka-Munk theory to create his model of human skin. The model he created took into account the various level of collagen, oxygenated and deoxygenated hemoglobin, melanosomes, water, skin depth, bilirubin, and beta-carotene. The model also took into account the different layers of the skin and their scattering, absorption, refraction, and reflection properties.

In 2014, Cain proved that it was feasible to identify a dismount in a diverse group with a high accuracy using a neural network as a classifier [2]. Cain created a database containing skin spectral data from a diverse group of people with different skin tone and ethnicity backgrounds. Cain then used the FCBF feature selection method and determined an optimal feature subset to reduce the feature space. Using the feature subset list, Cain trained a neural network using gradient descent with momentum and was able to achieve a high classification accuracy when trying to identify a POI from a diverse group.

III. Methodology

3.1 Introduction

The baseline model is used to determine whether it can identify a person of interest (POI) within their skin tone group or not. Parameters in the baseline model are adjusted to improve classification accuracy. The final model is compared to support vector machine (SVM) with various kernels. Gaussian noise is added to the training and testing data sets to simulate atmospheric noise from the day the Hyperspectral imaging (HSI) images are taken. HSI testing is done with the final model to determine the practical application in real life.

3.2 Materials and Equipment



Figure 3.1: FieldSpec[®] 3 with contact probe and ThinkPad. The contact probe is connected to the FieldSpec[®] with the optical cable and the power cable. The power cable power the built in illumination source and the optical cable collect and transmit the electromagnetic energy to the FieldSpec[®] 3. The FieldSpec[®] 3 is used to process, analyze, and interpolate the data. The data cable transmits the processed data from the FieldSpec to the *RS³* software in the ThinkPad.

The materials and equipment used consists of an Analytical Spectral Devices (ASD) FieldSpec[®] 3 [44] spectroradiometer with contact probe, Specim AisaDUAL hyperspectral sensor [45], MATLAB[®] 2014a, RS^{3TM} [46], ViewSpecTM Pro 6.0 [47], ENVI 5.0 [48] (Classic), and Weka version 3.7.11 [49]. The FieldSpec[®] 3 with contact probe is used to collect skin spectral data. Figure 3.1 shows the FieldSpec[®] 3 with the contact probe. The contact probe reduces the effect of noise due to atmospheric environment by direct contact with the skin. The contact probe has a built in illumination source to control illumination variance. The FieldSpec[®] 3 collects samples from 350 nm to 2500 nm with a sampling interval of 1.4 nm and 2 nm for the spectral region of 350 nm - 1000 nm and 1000nm - 2500 nm respectively. Figure 3.2 is an example of a spectral skin sample from the FieldSpec[®] 3.

To control and minimize noise when collecting data, the contact probe is calibrated prior to each collection through optimization and white reflectance. Optimization calibrates the FieldSpec[®] 3 to a light source. When the contact probe is used with the FieldSpec[®] 3, optimization calibrates the FieldSpec[®] 3 to the built-in light source of the contact probe. Optimization is accomplished when the ASD is initialized. White reflectance calibration is performed to normalize the reading. White reflectance is done every time before samples are taken. Optimization and white reflectance calibrations are controlled and performed in RS^{3TM}. RS^{3TM} is a proprietary software used by ASD to control the hyperspectral data collect with the FieldSpec[®] 3. All data collected by the FieldSpec[®] 3 using RS^{3TM} are saved as *'asd'* file format which is converted by ViewSpecTM to American Standard Code for Information Interchange (ASCII) *'txt'* files for processing in MATLAB[®]. MATLAB[®] is used to import and format the data collected for feature selection. The Neural Network Toolbox in MATLAB[®] is be used to train and test the artificial neural network (ANN).

HSI data is collected using the Specim AisaDUAL hyperspectral sensor. The AisaDUAL hyperspectral sensor is composed of two detectors: the AisaEAGLE and the AisaHAWK. The AisaEAGLE collects data in visible and near-infrared (VNIR) range (400-970nm) with a spectral resolution of 2.9 nm. The AisaHAWK collects data in the short-wave infrared (SWIR) range (970-2500nm) with a spectral resolution of 8.5nm. AisaDUAL collects images in radiance, and ENVI is used to remove atmospheric noise and convert the images from radiance to reflectance.

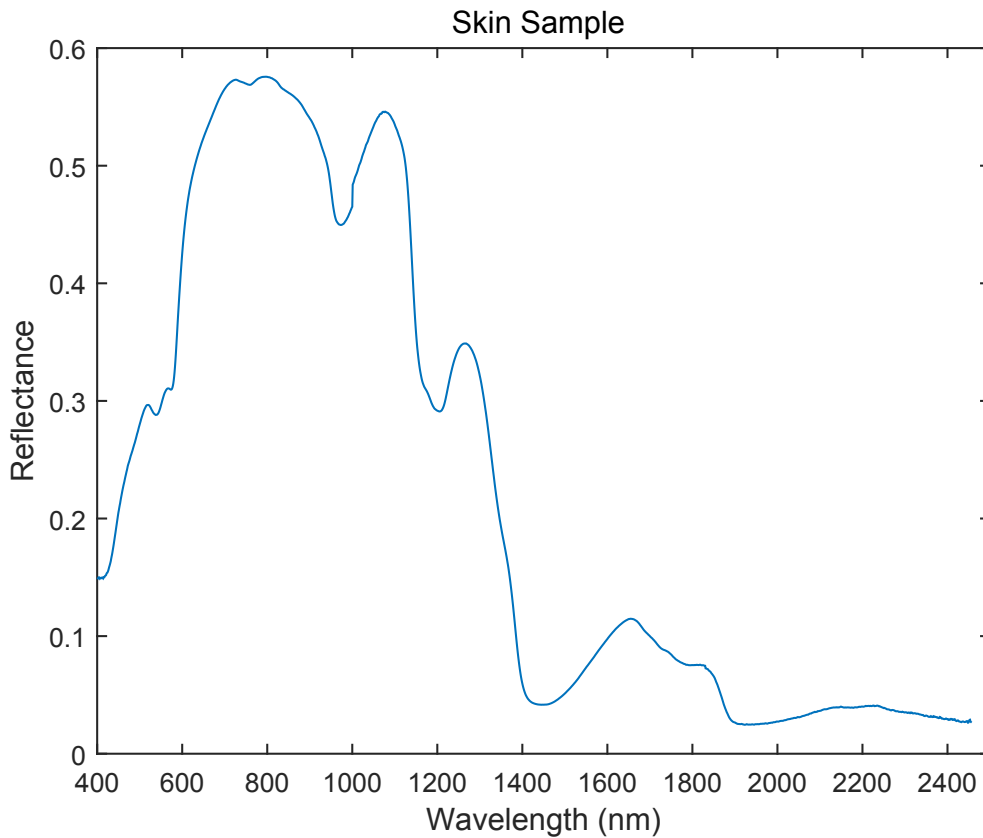


Figure 3.2: FieldSpec[®] 3 skin sample of the forearm.

3.3 Data

All skin data collected with the ASD is split into specific skin tone groups based on the melanin concentration. Jablonski and Chaplin identified that melanin is the dominating chromophore at 685nm [36]. Therefore, reflectance at 685nm is used to split the skin data into three skin tone groups. The melanin concentration cutoff range for group one is at 8% while the cutoff range between group two and three is at 16%. The reflectance value for these cut off ranges are calculated using Nunez's model [9]. Nunez's model calculates the spectral reflectance of a person based on the melanin concentration, blood level, oxygenation level, dermal thickness, carotene level, and subcutaneous tissue reflectance [9]. Calculation from Nunez's model represents data taken when the light source and sensor are normal to the skin [9]. This is ideal for partitioning the data into groups because the sensor used to collect the data is normal to the skin. Table 3.1 shows the skin tone group with their corresponding Fitzpatrick Skin type and melanin concentration.

Table 3.1: Skin tone group and their corresponding melanin concentration

Skin Tone Group	Fitzpatrick Skin Type	Melanin Concentration
Group 1	Type I & II	0 – 8%
Group 2	Type III & IV	8 – 16%
Group 3	Type V & VI	> 16%

Once the data is split into groups, 20% of the data from each group is randomly selected to create a validation set for each group for the generalization error. The other 80% of the data is used for training. When training and testing the model, a k-fold cross validation technique is used in reporting the classification accuracy. To obtain the generalization error, all data in the training set is used to train the ANN and validation set

is used for validation. No data is held out during training when calculating the generalization error.

3.4 Baseline Model Testing

The configuration for the baseline model is shown in Table 3.2. Table 3.2 defines a configuration of a neural network that is modeled in MATLAB[®] using the Neural Network Toolbox.

Table 3.2: Baseline Model Parameters

Parameter	Value
Inputs	f , number of features
Outputs	c , number of classes
Neurons in hidden layer	$h = \frac{N+O}{2}$
Hidden layers	1
Training Function	Gradient Descent
Activation Function	Sigmoid: $f(x) = (1 + e^{-x})^{-1}$
Learning rate	0.3
Momentum	0.2
Epochs	500
Cross Validation (K)	5
Normalization Method	$l^2 - norm$
Features	1024,1014,1033,1348

When modeling in MATLAB[®], the training function is set to *traingdm*. MATLAB[®]'s *traingdm* algorithm is a gradient descent function with momentum backpropagation training technique. Equation 3.1 is a weight update equation with the momentum variable

α for the backpropagation technique as defined by Robert Jacobs [50], and explained in Section 2.4.2.1.

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[\Delta w_{ji}^{(l)}(n-1)] - \eta \mathcal{L}_j^{(l)} y_i^{l-1}(n) \quad (3.1)$$

Skin tone group one is tested to determine the validity of the baseline model. During data collection, ten samples are taken from each individual. For group one, there are a total of 400 samples in the training set and 100 samples in the validation set or 40 and 10 dismounts/classes respectively. When a POI is tested, all ten of their samples are extracted from the training set and replicated to match the number of non-POI. For example, when the first POI from group one is tested, all ten of the POI's samples are extracted and are replicated 38 times to match the 390 non-POIs' samples. This is done to prevent biasing the network when training the model. All POI samples are labeled as 1 and all non-POI samples are labeled as 0. The POI samples are also duplicated to match the number of non-POI samples in the validation set. All results are reported as Equal Weighted Average (EWA) which is define as

$$EWA = \frac{P_{POI} + P_{non-POI}}{2}, \quad (3.2)$$

where P_{POI} is the probability of detecting the POI and $P_{non-POI}$ is the probability of detecting non-POI. The results for the evaluation set are presented in a confusion matrix format and the terms and formulas for the operating characteristic values are defined in Table 3.3.

3.5 Optimizing Model

To optimize the network model, various parameters are adjusted. The normalization methods, transfer functions, network topology, and training functions are investigated to determine the best neural network model. To determine the best normalization method for the ANN, the training and activation function remain constant while different

Table 3.3: Operating Characteristic Values

Term	Description/Formula
True Positive (TP)	Number of correctly identified POI
True negative (TN)	Number of correctly identified non-POI
False Positive (FP)	Number of non-POI's identified as POI's
False Negative (FN)	Number of POI's identified as non-POI's
Sensitivity or True Positive Rate (TPR)	$\frac{TP}{TP+FN}$
Specificity (SPC) or True Negative Rate (TNR)	$\frac{TN}{FP+TN}$
Precision or Positive Predictive Value (PPV)	$\frac{TP}{TP+FP}$
Negative Predictive Value (NPV)	$\frac{TN}{TN+FN}$
False Omission Rate (FOR)	$\frac{FN}{FN+TN}$
Fall-out or False Positive Rate (FPR)	$\frac{FP}{FP+TN}$
False Discovery Rate (FDR)	$1 - PPV$
Miss Rate or False Negative Rate (FNR)	$\frac{FN}{FN+TP}$
Accuracy (ACC)	$\frac{TP+TN}{TP+FN+FP+TN}$

normalization and network topologies are varied. The top two normalization methods are kept for further testing based on their average classification accuracy. Because there are infinite possible combinations for topologies, the maximum number of hidden layers and hidden nodes are limited to five hidden layers and five hidden nodes per layer. This limiting the maximum number of hidden layers and hidden nodes narrows down the topology search space to 3905 combinations. When varying topologies, 15 different combinations are picked for testing. The topologies are picked by randomly picking three combinations from each possible number of hidden layers. For example, three combinations are picked when there is only one hidden layer with a maximum of five hidden nodes in the layer.

Based on the top two normalization methods, the best activation function is determined. The training function remains constant while the topology and the activation function change. The number of topology combinations tested is the same as the normalization test. The number of activation functions tested is 15. The activation functions investigated are: competitive, elliot symmetric sigmoid, hard-limit, symmetric hard limit, log-sigmoid, inverse, positive linear, linear, radial basis, normalized radial basis, saturating linear, symmetric saturating linear, soft max, hyperbolic tangent sigmoid, and triangular basis. Each activation function is tested with the top two normalization methods to narrow down the best normalization technique. This process is repeated to determine the best training function and best topology. During topology testing, normalization method, activation function, and training function are held constant while the topology changes.

3.6 Support Vector Machine

A SVM with different kernels is tested to compare to the final neural network model presented in Section 3.5. The SVM kernels that are investigated are linear, radial basis function, and polynomial. The same data sets used for the neural network model are used with the SVM. The SVM is implemented in MATLAB[®] with the *svmtrain* and *svmclassify* functions. Table 3.4 shows the Kernel and their corresponding equations. There are various parameters that can be set for *svmtrain*, however, only the *autoscale* and *kernel_function* parameters are adjusted. All other parameters are left at default as specified in the MATLAB[®] documentation for *svmtrain*. The parameter *autoscale* is set to specify whether *svmtrain* automatically centers the data points at their mean and scales them to have unit standard deviation. This parameter is set false because the data sets are already normalized. The *kernel_function* specifies the kernel function that *svmtrain* uses to map the data. This parameter is set to either *linear*, *polynomial*, or *rbf*.

Table 3.4: SVM Kernels that are investigated. These are the kernels that are available for the *svmtrain* in MATLAB®. The kernel is specified with the *kernel_function* parameter.

Kernel	Equation
Gaussian or Radial Basis Function	$G(x_1, x_2) = \exp(-\ x_1 - x_2\ ^2)$
Linear	$G(x_1, x_2) = x_1^T x_2$
Polynomial	$G(x_1, x_2) = \exp(1 + x_1^T x_2)^p$

3.7 Testing New Model With Other Skin Tone Groups

After the best network configuration is determined, POIs from other skin tone groups are tested to ensure the model accuracy for all groups. Similar to the testing of group one, four POIs are chosen and data are duplicated for training and testing in each group.

Table 3.5 shows the settings for the final neural network model.

Table 3.5: Final Model

Parameter	Value
Inputs	4
Outputs	2
Neurons in hidden layer	3
Hidden layers	1
Training Function	Levenberg-Marquardt
Activation Function	Radbas: $f(x) = \exp(-x^2)$
Epochs	500
Cross Validation (K)	5
Normalization Method	Dividing by the max of each sample
Features	1024,1014,1033,1348

3.8 Noise Testing

Before testing with real HSI images, the network model is tested with noisy data. The noisy data are created by adding Gaussian noise to the clean FieldSpec[®] 3 data. The Signal-to-Noise Ratio (SNR) of the Gaussian noise is determined by calculating the standard deviation from the atmospheric noise determined from a National Institute of Standards and Technology (NIST) certified Spectralon[®] white panel in the HSI image. Noises are generated and added to the samples in the training and validation sets. Two noise simulation tests are done to determine the effect of atmospheric noise on the model. For one test, non-noisy FieldSpec[®] 3 data are used for training and noisy FieldSpec[®] 3 data are used for validation. Other tests focus on both training and validation data incorporating noise. Figure 3.3 is an example of skin data with Gaussian noise applied.

3.9 HSI Testing

The HSI images that are used for testing were collected using the AisaDUAL during the Minor Area Motion Imagery (MAMI) data collect in 2013 at Air Force Research Laboratory (AFRL). The AisaDual system is made up of the AisaHawk and AisaEagle sensors mounted together in a dual sensor bracket mount. Both sensors are vertically aligned. All images had to be registered to align data from both sensors in the image. Image registration is done in MATLAB[®] with code written by Lt James Arneal. The image registration code used the Sobel method for edge detection to determine how many pixels needed to be shifted for the data to line up.

For HSI testing, two images are used for training and a third is used for testing. Skin pixels for the first two images are hand selected and registered to create the training set. For the testing image, each individual in the image is outlined and registered to create a testing set. The results are compared to a truth image to determine the accuracy. Figure 3.4 contains the testing image with the POI and non-POIs outlined and the

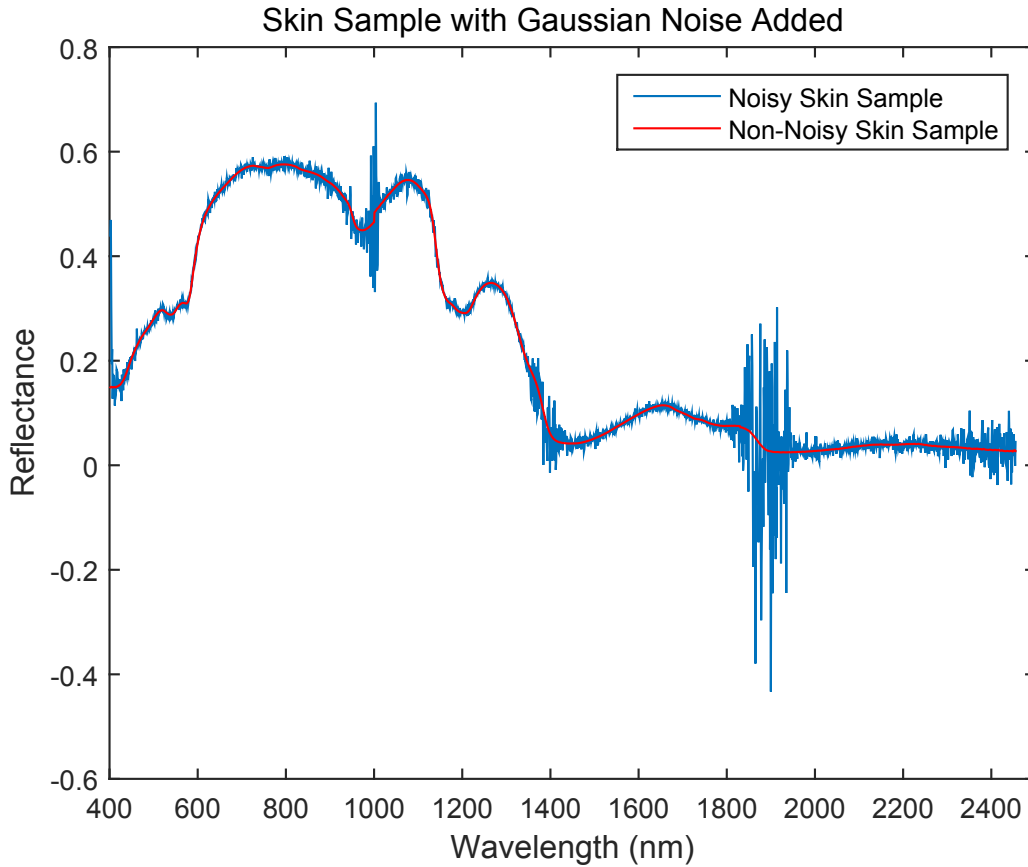


Figure 3.3: ASD skin sample with Gaussian noise. The red line is the original clean skin sample and the blue line is the skin sample with Gaussian noise applied.

corresponding truth image. The POI is outlined in green and the non-POIs are outlined in red. There are a total of 91 POI pixels and 357 non-POIs pixels in the truth image.

Figure 3.5 shows the two training images with the POI and non-POIs used for the training set outlined. The POI is outlined in green and the non-POIs are outlined in red. Two non-POIs are omitted from the training set from both images due to the difficulty of discerning their skin pixel from their shirt and shadow pixels.

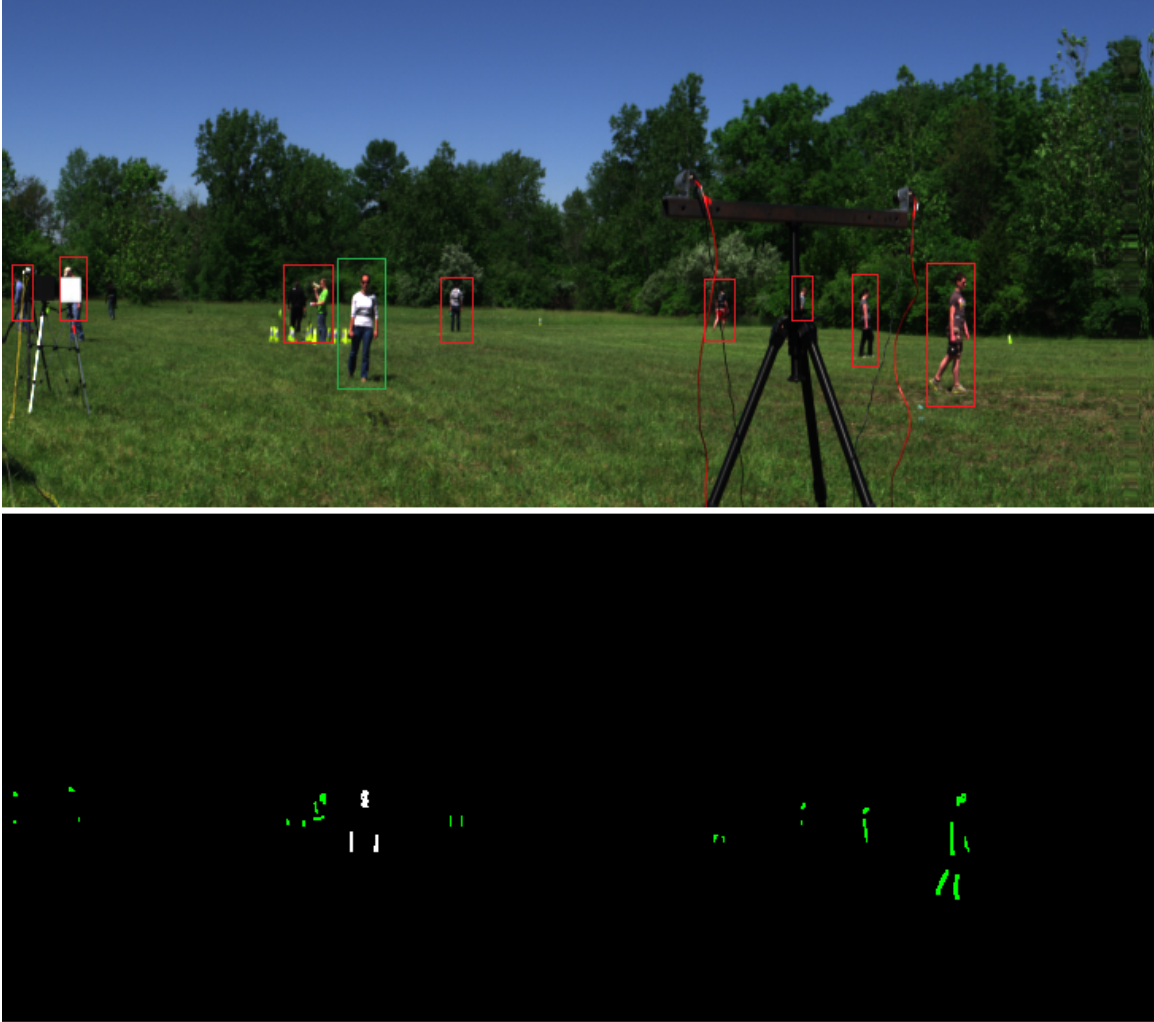


Figure 3.4: The test image is at the top and the truth image is the bottom. In the test image, the POI is outlined in green and the non-POIs are outlined in red. In the truth image, the white pixels are the POI and the green pixels are the non-POIs. There are 91 POI pixels and 357 non-POIs pixels in the truth image.

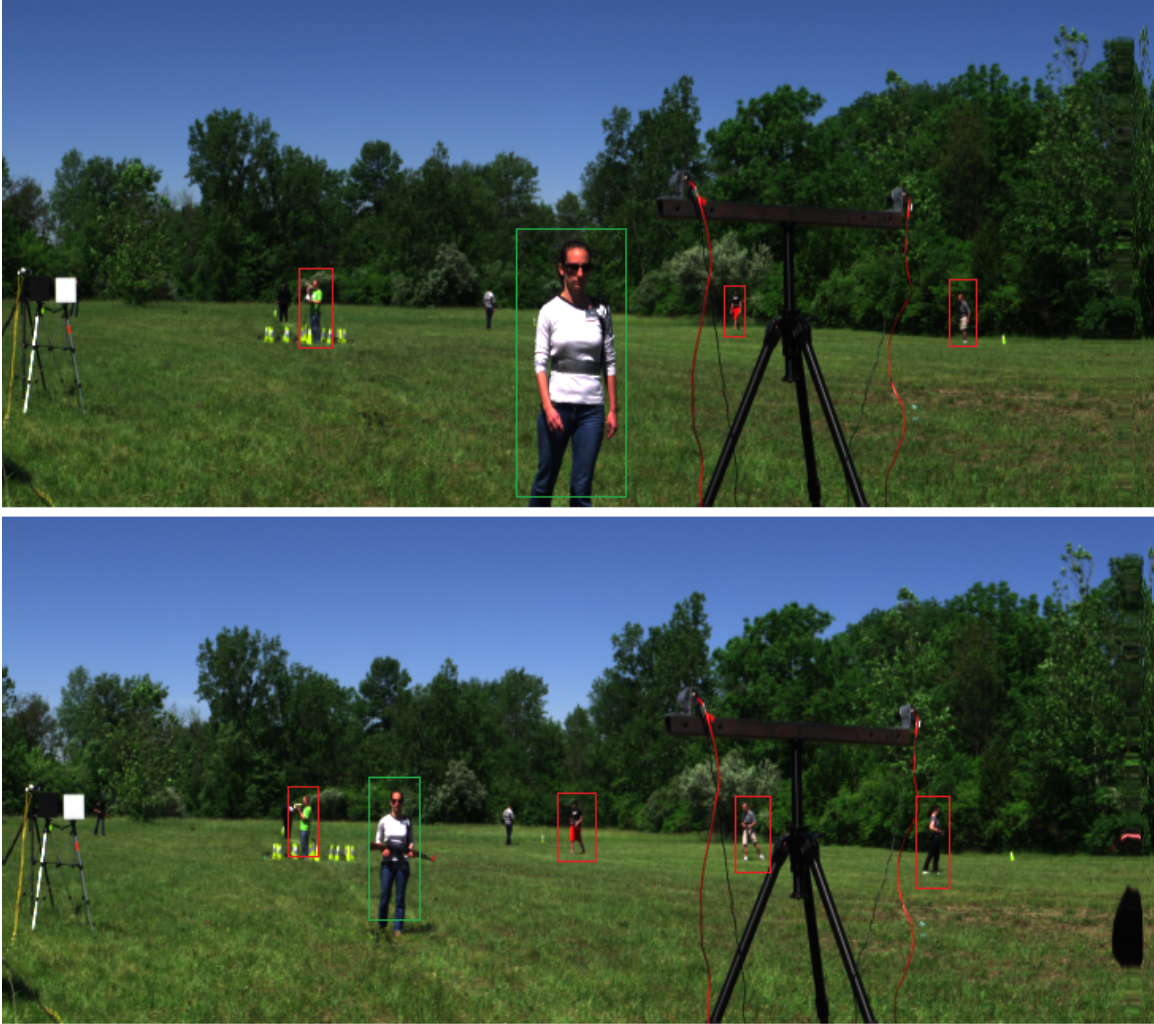


Figure 3.5: Training Images. The POI is outlined in green and the non-POIs are outlined in red. Two non-POIs were not used due to the difficulty in distinguishing their skin pixels from their shirt or shadow pixels.

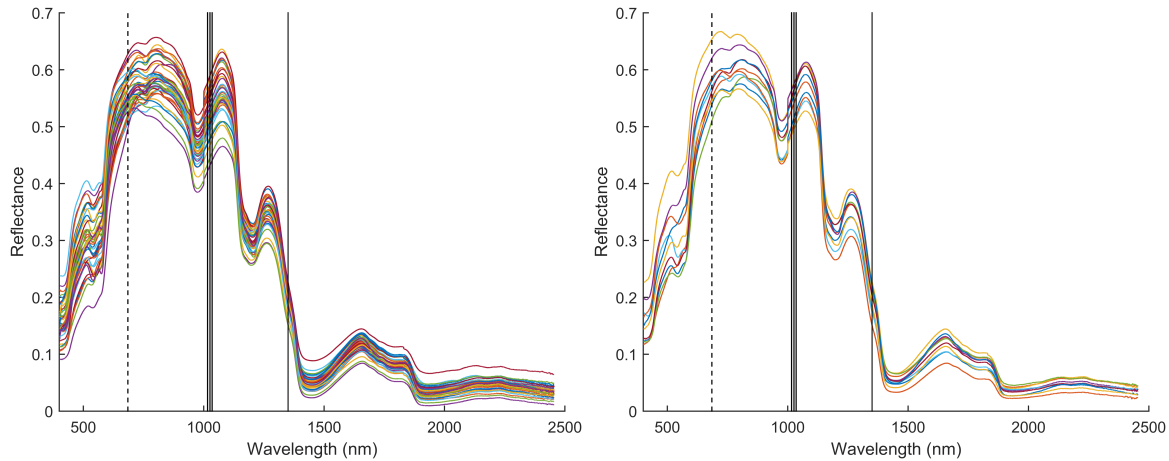
IV. Results and Analysis

This chapter is divided into five sections. Section 4.1 presents the results from testing the baseline model with the different skin tone groups. In Section 4.2 the results from testing the different normalization methods, activation function, training function, and network size in order to determine an optimal model are shown. The results from testing with different persons of interest (POIs) from different skin tone groups are shown in Section 4.3. Section 4.4 presents the results from testing the network with Gaussian noise. Results from testing the model with Hyperspectral imaging (HSI) data are presented in Section 4.3.

4.1 Model Testing

All data, used with the baseline model, is collected using an ASD FieldSpec 3 spectroradiometer with a contact probe. Each data sample is collected from 350 nm to 2500 nm wavelength with spectral resolution of 3 nm at 700 nm, 10 nm at 1400 nm, and 10 nm at 2100 nm. Ten samples are collected from each individual. The data set contains sample ranges from fair to dark skin from 101 individuals. The data set is parsed into three groups based on the melanin concentration at 685 nm. Jablonski and Chaplin identified that melanin is the dominating chromophore at 685 nm [36]. There are a total of 500 samples for group one, 390 samples for group two, and 120 samples for group three. For each skin tone group, 20% of the samples are extracted to create the validation set for their respective group. Figure 4.1 and 4.2 show the distribution of the samples in the training and testing sets. The figures also contain vertical lines to indicate the four features that Cain identified as the optimal features at identifying a POI in a diverse skin tone group as well as the location of the 685 nm wavelength that is used to divide them into groups. The vertical solid lines indicate the features while the vertical dash line denotes

the 685 nm wavelength used to divide the samples into their skin tone groups. Group one has the melanin concentration distribution of 8% or less, group two is between 8-16%, and group three is above 16% [51].



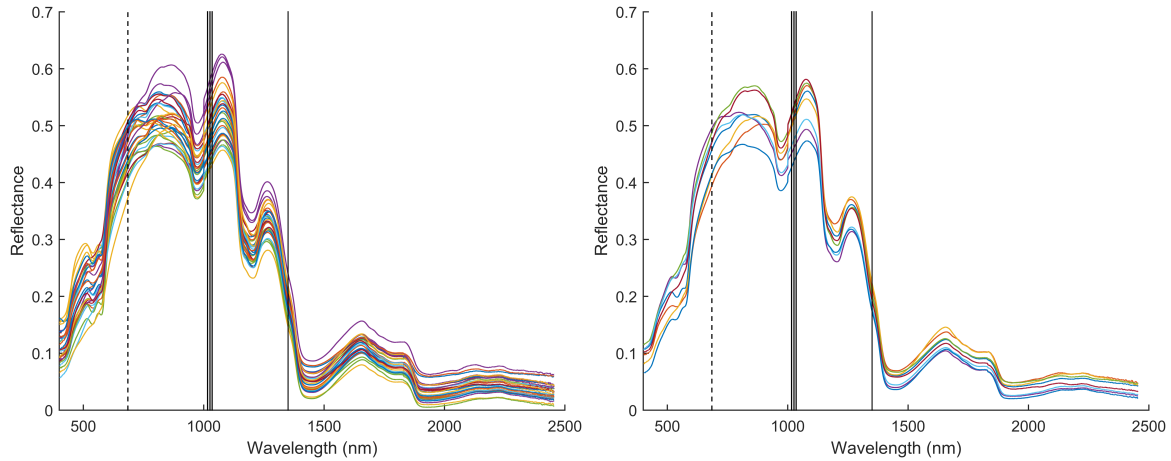
(a) Group 1 Training Set

(b) Group 1 Validation Set

Figure 4.1: Distribution of all classes in Group 1. The vertical dashed line denotes the 685 nm wavelength used to divide the samples into their skin tone groups. The vertical solid lines indicate the features (1024 nm, 1014 nm, 1033 nm, and 1348 nm). Group one has the melanin concentration distribution of 8% or less. Each color line is a sample of a class.

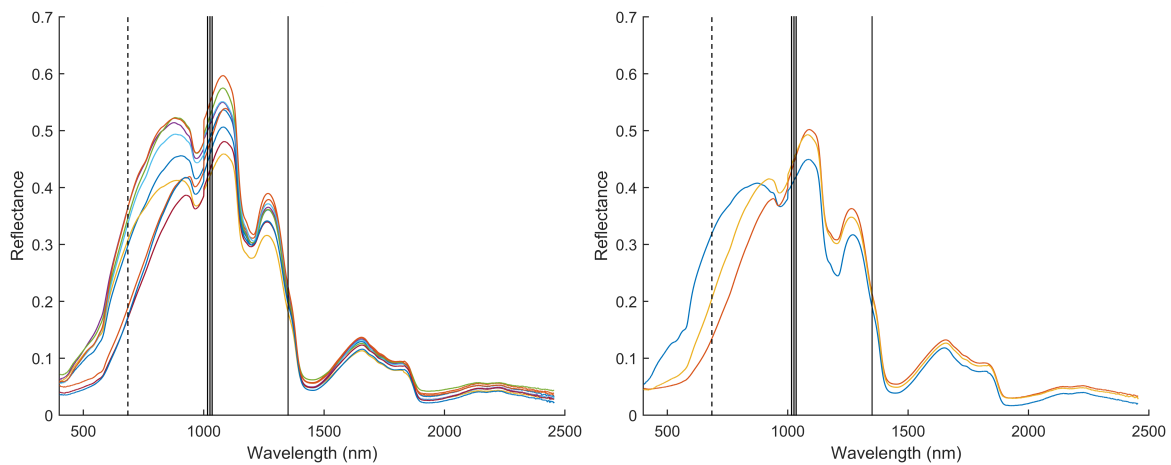
The baseline model is set up with the parameters listed in Table 3.2. Table 4.1 shows the classification accuracy of the 5-fold cross validation as well as the average accuracy of the 5-fold cross validation for the first four POIs in group one. Figure 4.2 shows the confusion matrix with the general classification accuracy for the first four POIs in group one. The general classification accuracy is obtained by using all the data in the training set to train the neural network and test with the testing set.

From the results in Table 4.2 and Table 4.3, the baseline model demonstrated an accuracy of 69-90%. Table 4.3 contains the operating characteristic values calculated from the POI in Table 4.2. Under closer inspection, out of the 32 misclassifications for POI 1,



(a) Group 2 Training Set

(b) Group 2 Validation Set



(a) Group 3 Training Set

(b) Group 3 Validation Set

Figure 4.2: Distribution of all classes in Group 2 and 3. The vertical dashed line denotes the 685 nm wavelength used to divide the samples into their skin tone groups. The vertical solid lines indicate the features (1024 nm, 1014 nm, 1033 nm, and 1348 nm). Group two has the melanin concentration distribution between 8% and 16%. Group three has the melanin concentration distribution of 16% or more. Each color line is a sample of a class.

20 of them are full misclassifications. A full misclassification is when all ten samples of a non-POI are classified as the POI and a partial misclassification is when there is less than

Table 4.1: Accuracy of the five fold cross validation for the first four POIs of group one using the baseline model. The average is calculated from averaging the results of the five folds.

POI	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
1	75%	83%	83%	83%	80.5%	80.9%
2	79.5%	86%	75%	83%	73%	79.3%
3	90%	81%	90.5%	90%	91%	88.5%
4	73%	81%	83.5%	81.5%	73%	78.4%

Table 4.2: Generalized Accuracy: the generalized accuracy is obtained by using all the data in the training set of group one to train the neural network and test with the testing set.

		POI 1		Total			POI 2		Total
		Truth					Truth		
Classified As		POI	nonPOI		Classified As		POI	nonPOI	
	POI	100	32	132		POI	100	38	138
	nonPOI	0	68	68		nonPOI	0	62	62
Total		100	100		Total		100	100	

		POI 3		Total			POI 4		Total
		Truth					Truth		
Classified As		POI	nonPOI		Classified As		POI	nonPOI	
	POI	100	20	120		POI	100	61	161
	nonPOI	0	80	80		nonPOI	0	39	39
Total		100	100		Total		100	100	

Table 4.3: Operating Characteristic values for POIs 1-4 for Table 4.2.

	POI 1	POI 2	POI 3	POI 4
Sensitivity or True Positive Rate (%)	100	100	100	100
Miss Rate or False Negative Rate (%)	0	0	0	0
Fall-Out or False Positive Rate (%)	32	38	20	61
Specificity or True Negative Rate (%)	68	62	80	39
Precision or Positive Predictive Value (%)	75.8	72.5	83.3	62.1
False Omission Rate (%)	0	0	0	0
False Discovery Rate (%)	24.2	27.5	16.7	37.9
Negative Predictive Value (%)	100	100	100	100
Equal-Weighted Accuracy (%)	84	81	90	69.5

ten samples of a non-POI that are misclassified. In the case of POI 1, two non-POIs were misclassified as the POI. The other 12 misclassification are partial misclassifications. For the other POIs in group one, there are 3 full misclassifications for POI 2, 2 full misclassifications for POI 3, and 6 full misclassifications for POI 4. The number of full misclassifications are high and are not consistent throughout. For real world application the network needs to be adjusted to reduce the number of misclassifications and be consistent.

4.2 Parameter Testing

There are numerous parameters that can be adjusted to improve the accuracy of the neural network. However, only four parameters are explored in this thesis: normalization methods, activation functions, training functions, and topology. Table 2.2 lists the different normalization methods that are investigated. The top two normalization methods are determined by taking the average accuracy from the 15 combinations and are kept for

further testing. The results for the top two normalization methods are in Tables 4.4 to 4.7. Results for other methods are presented in Appendix A. All percent accuracy results are Equal Weighted Accuracy calculated from the generalized results. The topologies are denoted as $[i, l_1, l_2, l_3, \dots, l_n, o]$ where the i is the number of inputs, the o is the number of outputs, and l_n is the number of nodes in the n^{th} hidden layer. The ‘Max Samples’ normalization has an average accuracy of 72.36% and ‘Unary Samples’ normalization has an average accuracy of 73.83%.

Table 4.4: Top two normalization method results from testing different normalizations with one or two hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology					
		[4 1 2]	[4 4 2]	[4 5 2]	[4 3 2 2]	[4 3 4 2]	[4 5 3 2]
Norm Method	Max Samples	40%	61.5%	78%	50%	65.5%	67.5%
	Unary Samples	74.5%	78.5%	79%	40.5%	84%	83.5%

Table 4.5: Top two normalization method results from testing different normalizations with three hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 3 1 1 2]	[4 5 3 5 2]	[4 1 2 1 2]
Norm Method	Max Samples	93%	85.5%	83%
	Unary Samples	82.5%	93.5%	45%

Table 4.6: Top two normalization method results from testing different normalizations with four hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 4 3 1 3 2]	[4 5 4 2 5 2]	[4 2 2 3 5 2]
Norm Method	Max Samples	82.5%	50%	83%
	Unary Samples	83%	56.5%	97%

Table 4.7: Top two normalization method results from testing different normalizations with five hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 2 1 4 1 2 2]	[4 4 4 4 4 5 2]	[4 5 4 4 1 4 2]
Norm Method	Max Samples	78%	85%	83%
	Unary Samples	50%	75%	85%

There are five stopping criteria in the Neural Network Toolbox in MATLAB[®] that can be set to prevent overfitting. The five stopping criteria are *min_grad*, *max_fail*, *time*, *goal*, and *epochs*. Only the *min_grad*, *max_fail*, and *epochs* are changed from the MATLAB[®] default values. The *time* and *goal* parameters are kept at the MATLAB[®] default values. The *min_grad* parameter is the Minimum Gradient Magnitude, and *max_fail* parameter is the Maximum Number of Validation Increases. The Maximum Number of Validation Increases represents the number of successive iterations that the validation performance fails to increase [52]. During these tests, the *epochs*, *min_grad*, and *max_fail* are set to

500, 0, and 50 respectively. The neural network continues training on the data samples until one of the five stopping criteria is met.

Testing of the activation and training functions is executed similarly to testing of the normalization methods. Similar to the testing of the normalization methods, only the parameter that is being tested and the topology are changed while all other parameters are kept constant per Table 4.1. The topology is limited to 15 randomly selected combinations similar to the normalization test. However, the activation functions will also be tested with the top two normalization methods that were determined. Different training functions are tested with the top normalization and activation function with varying topologies while all other parameters are kept constant per Table 4.1. Tables 4.8 to 4.9 shows the top two results from testing all the different activation functions available in the MATLAB® Neural Network Toolbox. Only the results for the topologies with only one hidden layer are shown in Tables 4.8 to 4.9. Results for other activation functions and hidden layers combinations are presented in Appendix A. The activation function *radbas*, radial basis, performed the best overall with an average accuracy of 87.23% when combined with ‘*Max Samples*’ normalization. When used in combination with ‘*Unary Samples*’ normalization, *radbas* produces an average accuracy of 84.83%.

Table 4.10 shows the results from testing three different training functions in MATLAB®: *traingdm*, *traingdx*, and *trainlm*. The training function *traingdm* is the gradient descent with momentum. The function *traingdx* is the same as *traingdm* except the learning rate decays. The *trainlm* function utilizes the Levenberg-Marquardt algorithm instead of gradient descent to train the network. For the *traingdm* and *traingdx*, the momentum and learning rate are set to 0.3 and 0.2 respectively. The normalization method and activation function are set to ‘*Max Samples*’ and *radbas* respectively. For *trainlm*, all parameters kept at MATLAB® default values with normalization and activation functions set to ‘*Max Samples*’ and *radbas*. Averaging the results of the different topologies,

Table 4.8: Top two activation functions as determined by normalization and MLP topology. The activation functions *radbas* and *tribas* are radial basis and triangular basis respectively. The results here are only from MLP topology with one hidden layer. Appendix A contains results for all other activation functions and hidden layers. All results are calculated Equal Weighted Accuracy generalized results.

		Normalization Method			
		Max Samples	Unary Samples	Max Samples	Unary Samples
		[4 5 2]	[4 5 2]	[4 4 2]	[4 4 2]
Activation Function	radbas	88%	84%	91.5%	83.5%
	tribas	97.5%	84%	88%	75%

Table 4.9: Top two activation functions as determined by normalization and MLP topology. The activation functions *radbas* and *tribas* are radial basis and triangular basis respectively. The results here are only from MLP topology with one hidden layer. Appendix A contains results for all other activation functions and hidden layers. All results are calculated Equal Weighted Accuracy generalized results.

		Normalization Method	
		Max Samples	Unary Samples
		[4 1 2]	[4 1 2]
Activation Function	radbas	74%	69.5%
	tribas	89%	50%

trainlm achieved an average accuracy of 95.55% while *traingdm* and *traingdx* achieved 84.3% and 91.7% respectively.

Table 4.10: Results from testing different training functions. All results are Equal Weighted Accuracy calculated from using all data in the training set for training.

		Topology		
		[4 5 2]	[4 4 2]	[4 1 2]
Training Function	traingdx	88%	91.5%	74%
	traingdm	88.5%	91%	65%
	trainlm	97.5%	80.5%	60%

When testing for the best topology, parameters for normalization, activation, and training function are ‘*Max Samples*’, ‘*radbas*’, and ‘*trainlm*’ respectively. Other parameters remain the same as the baseline model while the number of hidden layers and hidden nodes are varied. In MATLAB[®], when a topology has more than one hidden layer, an activation function must be specified for each hidden layer. If no activation function is specified then MATLAB[®] would use the default activation function of *tansig*. A test was accomplished with no hidden layer or node to determine if it is beneficial to have hidden layers and nodes. The results shows that having hidden layers and nodes does have a large effect on classification accuracy. However, it appears that the numbers of hidden layers and hidden nodes have little effects on the classification accuracy. This is due to the fact that after adjusting the normalization, activation, and training functions, the neural network was able to achieve a relatively high classification accuracy causing any possible gain from a larger topology to be minimal.

4.3 Group Testing

Table 4.11 is the final network configuration determined from the results presented in Section 4.2. Using this network configuration, POIs from group two and three are tested to validate effectiveness of the network for all skin tone groups. Table 4.12 shows the results of the classification accuracy from testing the first four POIs in group one. Results for group two and three are in Appendix B. Tests are accomplished with SVM using different kernels to compare the effectiveness of the neural network. Table 4.13 shows the SVM result for group one with the polynomial kernel. Results from group two and three as well as other kernels can be seen in Appendix B. Changing the kernels in the SVM did not appreciatively affect the accuracy. Overall the neural network performs better than SVM at identifying skin. The parameters adjustment based on the results from previous tests also improved the classification accuracy of the neural network.

Table 4.12: Accuracy of the five fold cross validation for the first four POI of group one.

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	96%	100%	100%	100%	100%	100%
	POI 2	96%	92%	100%	97.5%	100%	91.5%
	POI 3	90%	83.5%	100%	81.5%	99.5%	99.5%
	POI 4	85%	100%	100%	95%	73.5%	94.5%

Table 4.11: Final model from testing the various normalization methods, activation functions, training functions, and topologies available in MATLAB®. Final normalization method, activation function, and training function were determined by the best average accuracy from varying the topology.

Parameter	Value
Inputs	4
Outputs	2
Neurons in hidden layer	3
Hidden layers	1
Training Function	Levenberg-Marquardt
Activation Function	Radbas: $f(x) = \exp(-n^2)$
Epochs	500
Cross Validation (K)	5
Normalization Method	Dividing by the Max
Features (nm)	1024,1014,1033,1348

Table 4.13: SVM Results with polynomial kernel for the first four POIs in group 1. The results are for the five fold cross validation. All results are calculated with Equal Weighted Accuracy.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
POI 1	73%	73%	73%	66%	72.5%	72%
POI 2	55%	55%	55%	55%	55.5%	59.5%
POI 3	90%	90%	90%	90%	90%	90%
POI 4	51.5%	56%	61.5%	61.5%	56.5%	62.5%

Group 1

4.4 Simulated Noise

Before testing with HSI images, the network model is tested with noisy ASD data. Gaussian noise is added to the non-noisy ASD data to simulate the atmospheric noise for the specific day the HSI images were taken. To determine the noise standard deviation, spectral data from a spectralon white panel in the HSI image is extracted and the standard deviation is calculated. The spectralon white panel is National Institute of Standards and Technology (NIST) certified with a known reflectance of 1 from the 350 nm to 2500 nm wavelength. Noise in the data recorded from the spectralon is considered as atmospheric noise. The HSI images were taken with an AisaDUAL hyperspectral sensor. The AisaDUAL consist of the AisaHAWK and the AisaEAGLE. The AisaEAGLE is a VNIR sensor that collects spectral data in the 400-970 nm range with a spectral resolution of 3.3 nm. The AisaHAWK is a short-wave infrared (SWIR) sensor that collect spectral data in the 970-2500 nm range with a spectral resolution of 12 nm. However, even though the AisaDUAL specification listed it as capable of collecting spectral data between 400-2500 nm, actual HSI images were only able to collect in the 402-2455 nm range. The HSI images taken have spectral bands of 4.6 nm in the visible and near-infrared (VNIR) range and 5.8 nm in the SWIR range. This gives a total of 361 spectral bands from the 402-2455 nm range. The standard deviation was calculated for each spectral band in the HSI between the 402-2455 nm range using the *std* function in MATLAB®. Figure 4.3 shows the standard deviation of the noise for each band of the spectralon white panel.

There are three wavelength ranges that show an increase in noise. The increase of noise in the 927-1017 nm range is due to the transitioning between the two sensors in the AisaDUAL. Figure 4.4 is a layer in the HSI image at wave band 1007 nm which shows the transition from the AisaEAGLE to the AisaHAWK. The increase of noise in the 1329-1454 nm and 1797-1991 nm is due to the water in the atmosphere. Given that there are only 361 band from the HSI image, interpolation of the data was needed to account for

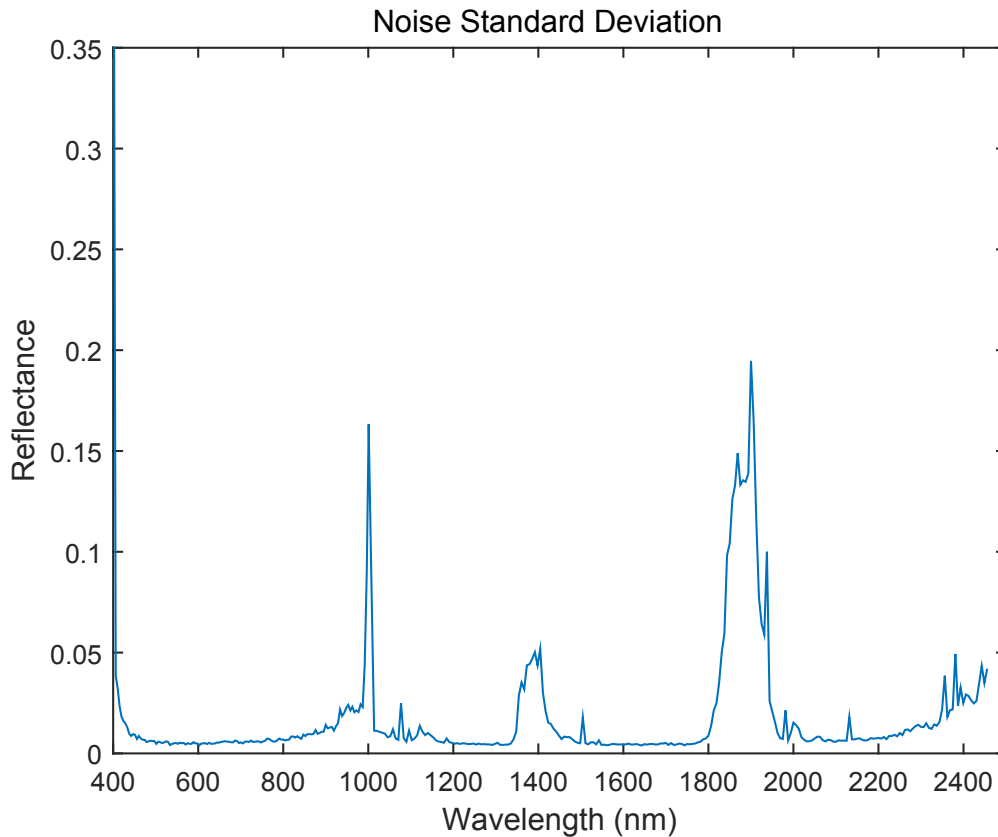


Figure 4.3: The standard deviation for the noise calculated from the spectralon white panel in the HSI image taken during the MAMI data collect in 2013. The standard deviation is calculated for each of the waveband taken by the AisaDUAL. The increase in noise in the 1000nm range is due to the transitioning between the AisaEAGLE and AisaHAWK. The increase in noise in the 1300 nm and 1800 nm regions are due to the water in the atmosphere.

each wavelength between 402-2455 nm and was accomplished using the *interp1* function with ‘*spline*’ method in MATLAB[®]. Wavelength from 350-402 nm and 2455-2500 nm ranges in the ASD data were removed to match the data from the HSI image. The standard deviation for each wavelength from 402 nm to 2455 nm are used to generate the Gaussian noise and is added to each ASD sample in the data set. Figure 4.5 shows the spectral data

of a non-noisy spectralon white panel with Gaussian noise added, and the spectral data of the spectralon white panel from the HSI image.



Figure 4.4: A layer in the HSI image at wave band 1007 nm. As can be seen in the figure, the ghosting effect indicates the transition from the AisaEAGLE to the AisaHAWK. This requires image registration to correctly align the two sensors.

Two different tests are performed with Gaussian noise. A test is performed by training the neural network with non-noisy ASD data and testing the network with noisy ASD data. Another test is accomplished by both training and testing the neural network with noisy ASD data. Table 4.14 shows the results of training with non-noisy data and Table 4.15 shows the results of training with noisy data for group one. Results for group two and three are in Appendix C. The results show that training with noisy data performs better than non-noisy data. The results show a small decrease in accuracy when training with noisy data. This is because of the increase of noise in the 1000 and 1300 nm regions. As previously mentioned, the 1000 nm region is where the sensor transitions from the AisaEAGLE to the AisaHAWK and the 1300 nm region is the water absorption region.

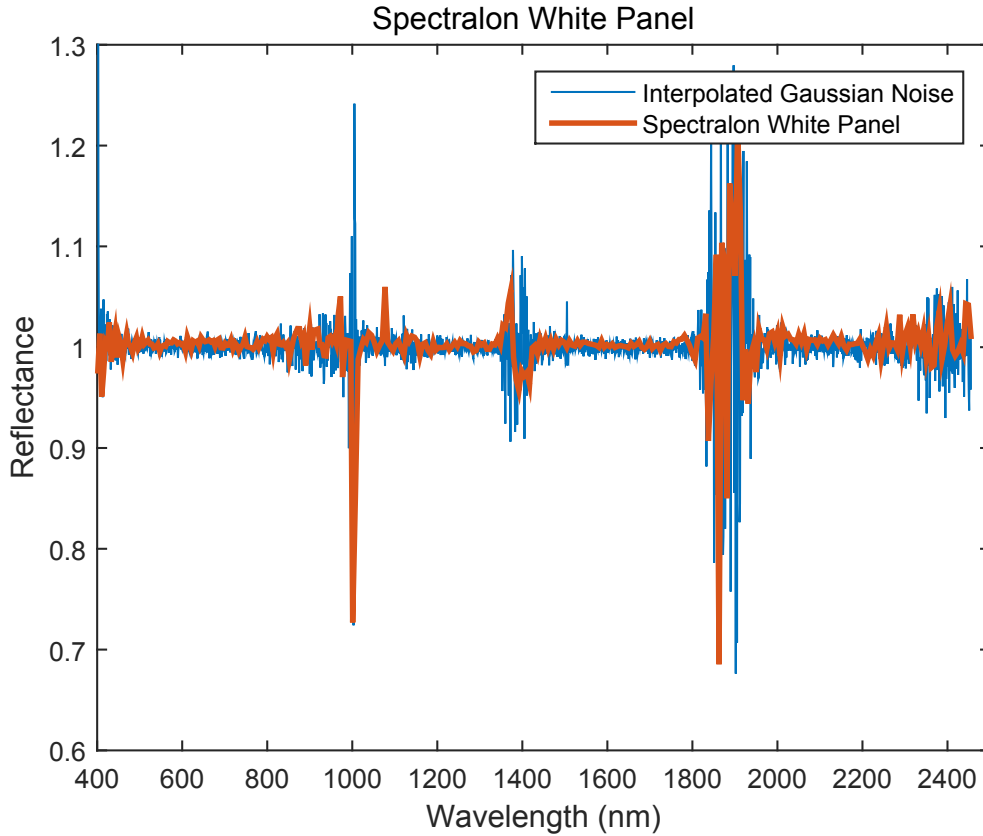


Figure 4.5: This shows the spectral data of a non-noisy spectralon white panel with Gaussian noise added, and the spectral data of the spectralon white panel from the HSI image. The red line is the spectralon white panel from the HSI image and the blue line is the non-noisy spectralon white panel with Gaussian noise added.

Three of the four features in the optimal feature set are in the 1000 nm range and the fourth feature is in the 1300 nm water absorption region.

4.5 HSI Testing

Three HSI images are used for HSI training and testing. The HSI images are collected using the AisaDUAL hyperspectral sensor system. The first two are used for training

Table 4.14: Classification results of training with non noisy data for Group 1

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	45.5%	55%	44.5%	50%	53%	49.5%
	POI 2	50%	49%	48.5%	47%	47.5%	50.5%
	POI 3	65%	66.5%	50%	60.5%	57%	65%
	POI 4	43%	41.5%	35.5%	48.5%	52.5%	50%

Table 4.15: Classification results of training with noisy data for Group 1

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	81%	77%	80.5%	82.5%	94%	86.5%
	POI 2	83.5%	89%	84.5%	82.5%	84.5%	93%
	POI 3	95%	85%	88.5%	85.5%	97.5%	94%
	POI 4	90%	91.5%	86.5%	90.5%	89.5%	91%

while the third is used for testing. In the first two images, the skin pixels are individually selected with the *ginput* function in MATLAB[®]. The selected pixels are labeled and normalized accordingly. The images are also registered prior to pixel extraction. Image registration is required to align the data from the two sensors in the AisaDUAL. As can be seen from Figure 4.4, the parallax causes a larger offset for objects closer to the AisaDUAL. This require each POI and non-POI to be registered separately and then combined to make the training set. Image registration is accomplished using edge detection courtesy of the code created by Lt James Arneal.

Figure 4.6 contains the training images used for the network. In both images, the POI is outlined in green and the non-POIs are outlined in red. Two non-POIs in both images

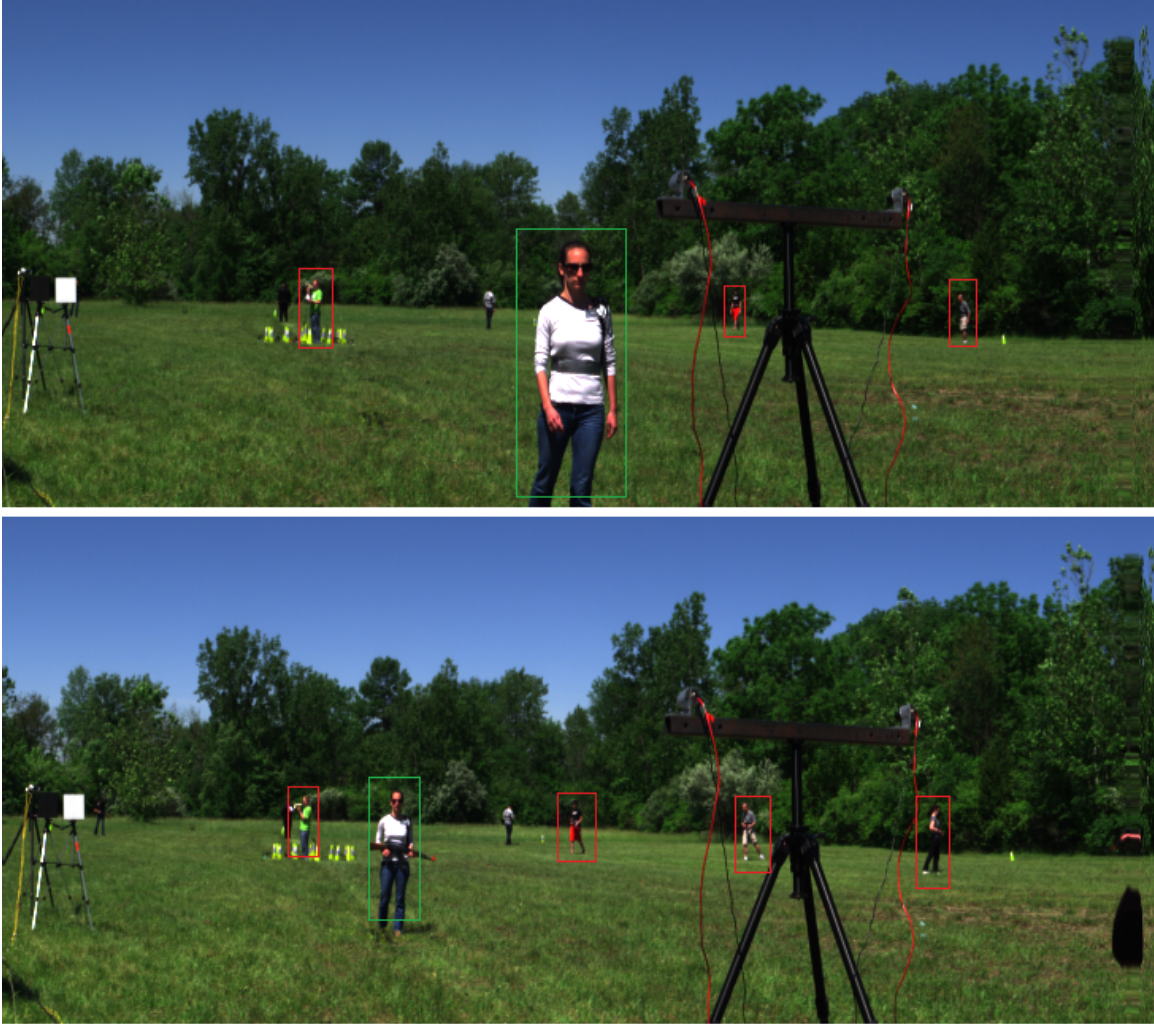


Figure 4.6: Training Images. The POI is outline in green and the non-POIs are outlined in red. Two non-POIs were not used due to the difficulty in distinguishing their skin pixels from their shirt or shadow pixels.

were not used due to the difficulty of discerning their skin pixels from their shirt and shadow pixels.

Figure 4.7 is the test image used for evaluating the network. The bottom image in Figure 4.7 is the truth image used to calculate the classification accuracy. The white pixels in the truth image are the POI skin pixels and the green pixels are the non-POIs' skin

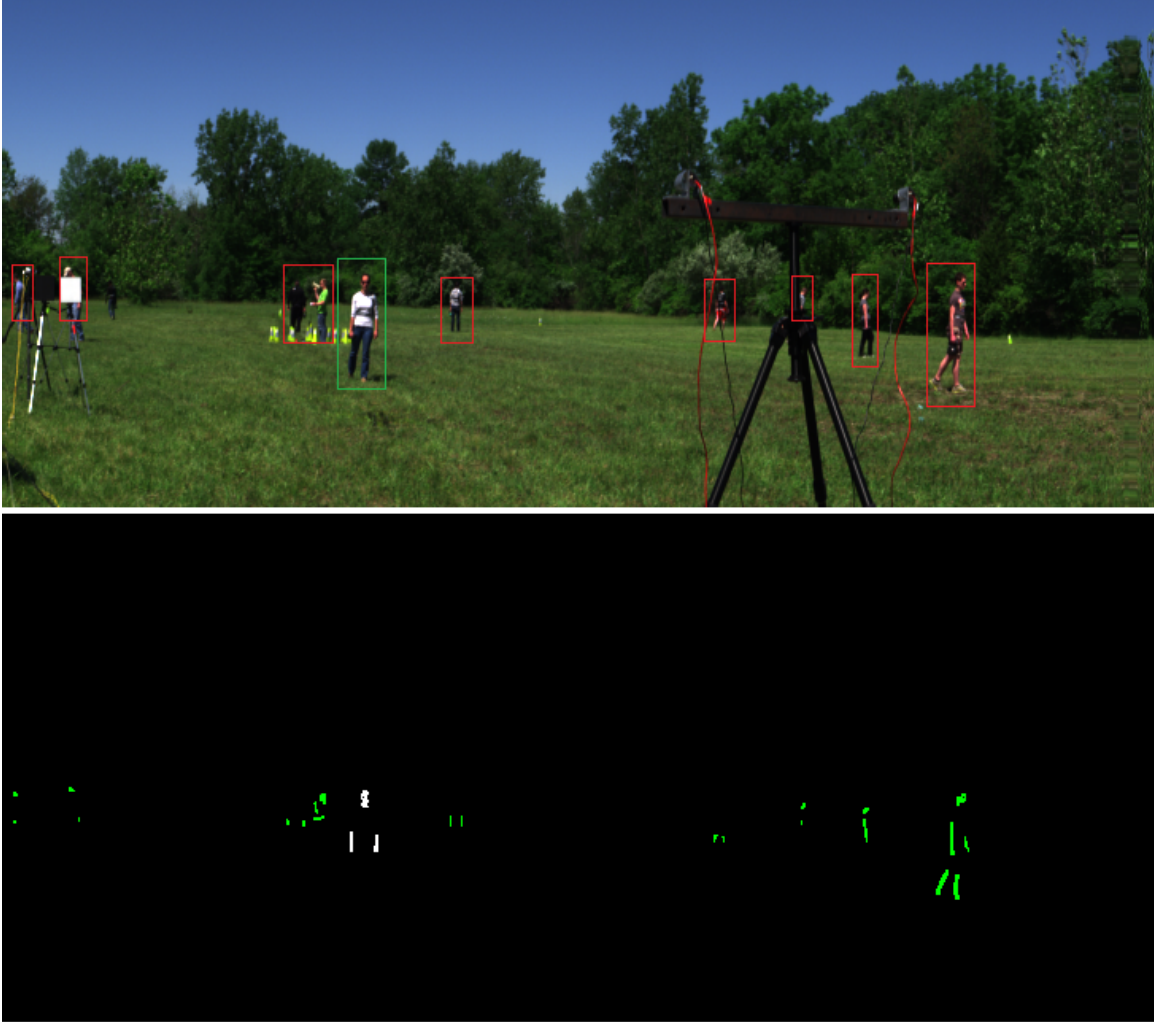


Figure 4.7: The test image is at the top and the truth image is the bottom. In the test image, the POI is outline in green and the non-POIs are outlined in red. In the truth image, the white pixels are the POI and the green pixels are the non-POIs. There are 91 POI pixels and 357 non-POI pixels in the truth image.

pixels. There are a total of 91 POI and 357 non-POI pixels in the truth image. Figure 4.8 is the result from the neural network. The neural network was able to correctly classify four of the POI skin pixels. However, the network also misclassify 79 of the non-POI pixels. The classification accuracy of the network is roughly around 41%.



Figure 4.8: Result from using the optimal feature set (1014 nm, 1024 nm, 1033 nm, and 1348nm) with the neural network. Identified POI pixels are colored in white and non-POI pixels are colored in light blue. Most of the POI pixels are correctly identified, however, most of the non-POIs are also incorrectly identified as the POI.

Fast Correlation-Based Filter (FCBF) was reaccomplished to determine if a new feature set can alleviate the disparity between the HSI result and the results from the noise simulation tests. Certain wavelength ranges were omitted from FCBF in order to avoid obtaining the same feature set. Ranges between 927-1017 nm, 1329-1454 nm, and 1797-1991 nm were omitted because they are in the water absorption spectrum and the sensor transitioning range. Wavelength range of 390-800 nm was also omitted because they are the color spectrum and are highly dependent upon lighting. FCBF was performed on the simulated noisy ASD data with the aforementioned wavelengths removed. The resulting wavelengths from FCBF are 804 nm, 1137 nm, 891 nm, 1343 nm, 1285 nm, 1255 nm, 1564 nm, and 1695 nm. Figure 4.9 is the result from using the top four features in the new feature set.

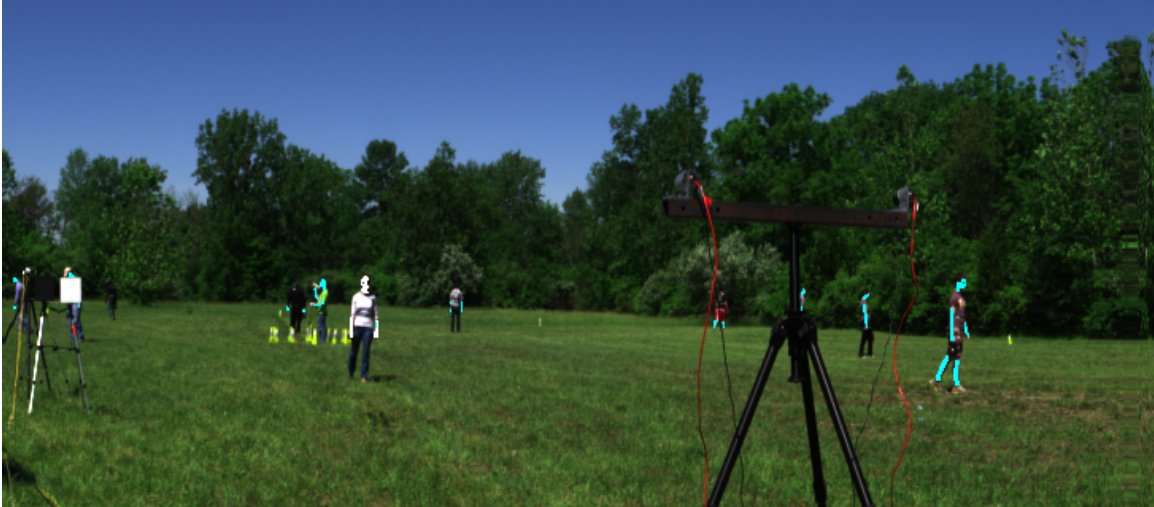


Figure 4.9: Result of neural network with the top four features (804 nm, 1137 nm, 891 nm, 1343 nm) in the new feature set.

The Equal Weighted Average (EWA) of the new feature set is roughly 54%. Tests are ran with the new feature set using l^2 norm and bigger topology. The EWA for these tests are between 50% - 60% and their results are in Appendix D. A closer examination at the pixels used to train the network helped determined the cause of the poor performance. Figure 4.10 shows some of the samples of the POI and non-POIs used for training the network. Both the POI and non-POI samples exhibit a large overlapping range. Even after normalization, the samples still showing large overlap as can be seen in Figure 4.10.

The large variation in spectral data for the POI can be caused by the different angles of incidence. As can be seen in Figure 4.11, the different angles of incidence have a large effect in the intensity level. Even though the samples are taken from the same person, the different intensity due to the different angles of incidence can cause the samples to be construed as samples from a different person. The large variation can also be due to the different area of the body. Cain demonstrated that different areas of the body have slightly

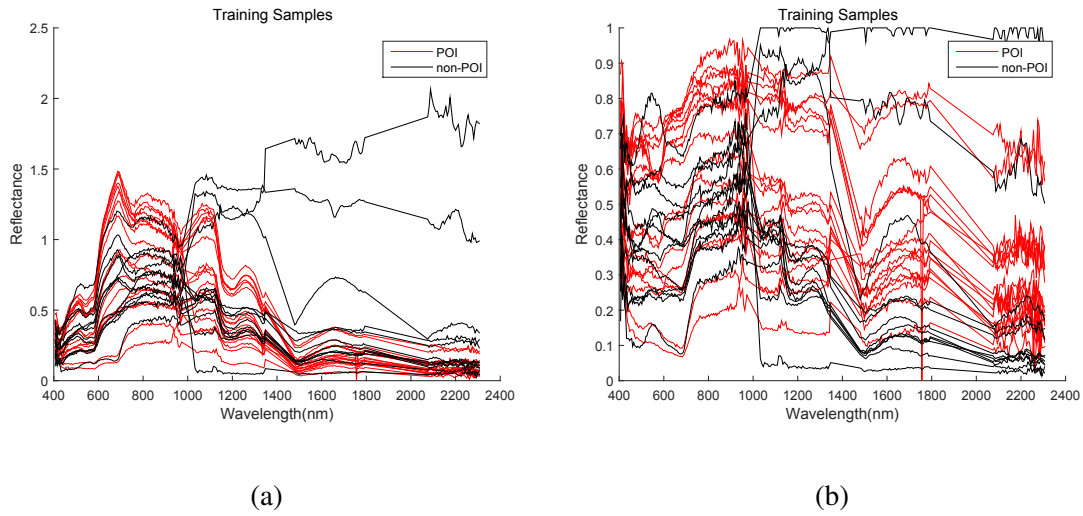


Figure 4.10: (a) shows the samples of the POI and non-POIs used for training the network. The samples shown are not normalized. (b) shows samples of the POI and non-POIs after normalization using normalization method number 1.

different spectral readings [2]. The different area of the body and the different angles of incidence could have contributed to the large variation in the POI and non-POIs samples.

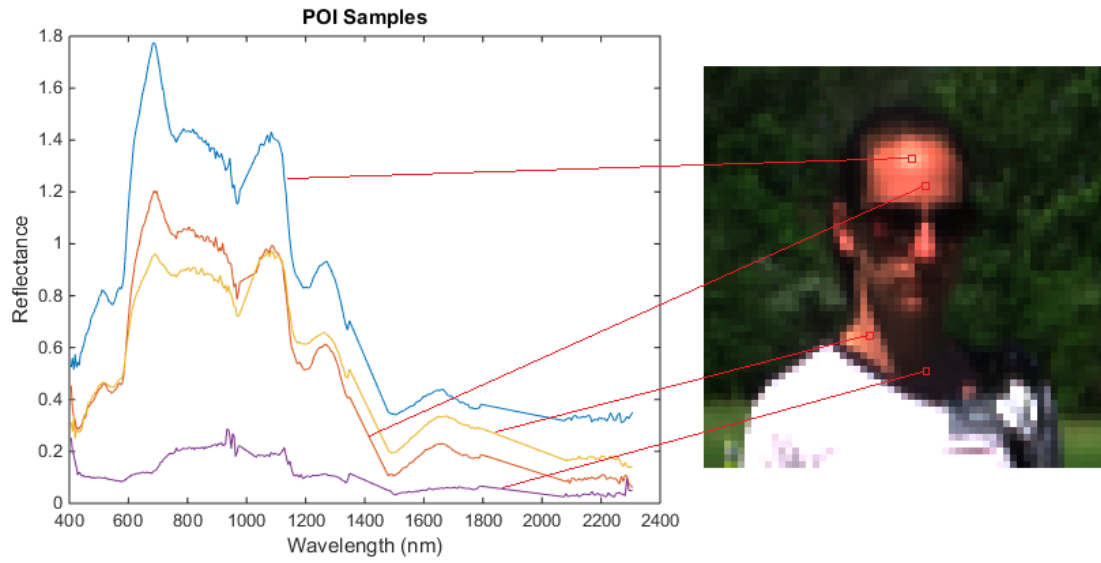


Figure 4.11: Samples of the POI and non-POIs after normalization using normalization 1.

V. Conclusion

5.1 Introduction

This thesis tested and determined if a baseline neural network model was sufficient in identifying a person of interest (POI) within his/her skin tone group and assessed the practical application of using an artificial neural network (ANN) for identification with real world hyperspectral images. A baseline model of the neural network was created in MATLAB[®] using the Neural Network Toolbox and was determined to be sufficient at identifying the POI down to 2-4 people. However, to obtain better classification accuracy, a higher fidelity model is required. The normalization method, activation function, training function, and topology were investigated and optimized to determine a better classification model. Based on the results from investigation, a new optimized neural network model was determined and compared to a support vector machine (SVM) model with different kernels to determine which classifier is better at skin identification. Gaussian noise was added to test the model to simulate atmospheric noise in an Hyperspectral imaging (HSI) image. The new neural network model was tested with real HSI images to determine real world application.

5.2 Summary of Results

By optimizing the multilayer perceptron (MLP) model, the training function provided the highest improvement compared to the normalization, activation function and topology. The classification accuracy increased from 69-90% to 80-100% when the training function was changed from gradient descent with momentum to the Levenberg-Marquardt algorithm. Changing the topology showed minimal improvement to the classification accuracy. However, the minimal improvement was because after adjusting the normalization, activation and training functions, the classification accuracy was already in

the 90-99% range. A second topology test was accomplished with the normalization, activation function, and training function set to the same configuration as the baseline model. The results from this test showed that varying the topology provided stability of the ANN rather than improving the classification. The new ANN model showed large improvements in identifying the POIs within their respective skin tone groups over the baseline model.

When comparing the new ANN model with a SVM, the new ANN model outperformed the SVM model in all three skin tone groups. Both SVM and the new ANN model used the same data and the SVM results were inconsistent. SVM showed large differences in classification accuracy between different POIs in the same class as compared to the neural network. The neural network was consistently able to classify with Equal Weighted Average (EWA) in the 90-100% range for POIs within the same skin tone group. SVM classification accuracy showed a range of 60-94%. Using different kernels in the SVM, the differences in the accuracy are negligible. The inconsistency and large standard deviation of classification accuracy with the SVM results make it unsuitable for POI identification.

The neural network showed high classification accuracies for noisy data. The noise was generated from the standard deviation calculated from each wave band of the spectralon white panel in the HSI image and modeled by a Gaussian noise model. The high accuracy is only achievable when the network was trained with noisy data. When training with non-noisy data and testing with noisy data, the network shows a dramatic decrease in classification accuracy. When training with non-noisy data and testing with noisy data, the accuracy dropped to 40-65%, however, training with noisy data increases the accuracy to 75%-94%. The decrease in accuracy compared to training and testing with non-noisy data is due to the high level of noise in the 1000 nm and 1300 nm range. The noise increase around 1000 nm can be attributed to the transition between the AisaEAGLE

and the AisaHAWK, and the noise at the 1300 nm bandwidth can be attributed to the water absorption phenomenon.

Based on the results from testing with noisy data, it was assumed that there would be a decrease in accuracy when testing with real HSI images. The results in Section 4.5 verified that the classification accuracy did decrease. The classification accuracy was lower compared to the simulated noise results. This decrease from simulated data to real data is attributed to the fact that the feature set that was identified as the optimal features were in highly noisy regions. The optimal features that were identified, 1024 nm, 1014 nm, 1033 nm, and 1348 nm, were in the water absorption spectrum and the ranges where the AisaDual transitions from the AisaEagle to the AisaHawk. This required a new set of features, requiring Fast Correlation-Based Filter (FCBF) to be reaccomplished. When FCBF was reaccomplished, the sensors transitioning wavelength, water absorption wavelength, and other certain wavelength ranges were omitted from consideration due to similar issues. However, even with a new feature set, the highest classification accuracy the network was able to obtain was 60%. A second possible reason is the angle of incidence of solar radiation. Different angles of incidence produce different intensities which can be seen in Figure 4.11. Another possible reason is what Cain pointed out when she measured the standard deviation within and between classes [2]: the deviation between the different areas of the body with atmospheric noise might have been too much to correctly train the network.

When testing with data collected from the FieldSpec 3, the ANN was able to perform extremely well for both non-noisy and noisy conditions. However, when using real HSI images, the neural network performance is extremely poor. The ANN misclassified most of the POI and non-POI skin pixels. This could be due to the standard deviation of the different areas of the body and the different angles of incidence.

5.3 Future Work

There are a few possible works that can be accomplished to expand upon skin identification. One possible future work is to retest the neural network with images taken with a different HSI sensor. One reason for the low classification accuracy that the network was experiencing was because the optimal feature set that was used was in the range where the two sensors were in transition. The transition introduced a high level of noise that made it difficult for the neural network to classify correctly using those optimal features.

Another possible direction to take is to test the HSI image using radiance instead of reflectance. One reason that the network failed to correctly identify the POI in the image was the different angle of incidence of solar reflection. The different angle of incidence caused the skin reading to have different intensity which in turn created a wide range of possible POI skin reflectance. Beisley's work on radiance showed that he was able to classify skin pixels within an HSI image in both shadow and highly lit areas with high accuracy [53].

Appendix A: Model Optimization Results: Normalization

Table A.1: Normalization methods investigated in this thesis.

Normalization Method	Formula	Description
Max Samples	$\frac{X}{X_{max}}$	Normalization by dividing the sample X by the maximum value of the sample.
Area Normalization	$\frac{X}{X_{area}}$	Normalization by dividing the sample X by the area under the curve of the sample.
Unary Features	$\frac{X_f - f_{min}}{f_{max} - f_{min}}$	Normalization by making everything 0 to 1 in each feature. X_f is the feature f in sample X and f_{min} and f_{max} are the minimum and maximum values of that feature in the data set respectively.
Max Features	$\frac{X_f}{f_{max}}$	Normalization by dividing each feature in sample X_f by the maximum value of that feature from all the samples.
Euclidean	$\frac{X}{\ X\ }$	Normalizes by dividing each sample by its L-2 norm.
Unary Samples	$\frac{X - X_{max}}{X_{max} - X_{min}}$	Normalization by making everything 0 to 1 in each sample. X_{min} and X_{max} are the minimum and maximum values of sample X respectively.

Table A.2: MLP results from testing different normalizations with one and two hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology					
		[4 1 2]	[4 4 2]	[4 5 2]	[4 3 2 2]	[4 3 4 2]	[4 5 3 2]
Norm Method	No Normalization	85%	78%	78%	83%	89.5%	78.5%
	Max Samples	40%	61.5%	78%	50%	65.5%	67.5%
	Area Normalization	30%	80%	83%	75%	82%	73%
	Unary Features	63%	80%	80%	22%	82.5%	80.5%
	Max Features	79%	68%	69.5%	64%	68.5%	72.5%
	Euclidean	74%	79.5%	77%	35%	76%	87%
	Unary Samples	74.5%	78.5%	79%	40.5%	84%	83.5%

Table A.3: MLP results from testing different normalizations with three hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 3 1 1 2]	[4 5 3 5 2]	[4 1 2 1 2]
Norm Method	No Normalization	50%	83%	83%
	Max Samples	93%	85.5%	83%
	Area Normalization	50%	81.5%	50%
	Unary Features	27%	83%	85%
	Max Features	31.5%	83%	69.5%
	Euclidean	73.5%	75%	50%
	Unary Samples	82.5%	93.5%	45%

Table A.4: MLP results from testing different normalizations with four hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 4 3 1 3 2]	[4 5 4 2 5 2]	[4 2 2 3 5 2]
Norm Method	No Normalization	50%	84.5%	50%
	Max Samples	82.5%	50%	83%
	Area Normalization	81%	88%	50%
	Unary Features	50%	65.5%	41%
	Max Features	73%	83%	50%
	Euclidean	80%	79.5%	88%
	Unary Samples	83%	56.5%	97%

Table A.5: MLP results from testing different normalizations with five hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology		
		[4 2 1 4 1 2 2]	[4 4 4 4 4 5 2]	[4 5 4 4 1 4 2]
Norm Method	No Normalization	50%	83%	55%
	Max Samples	78%	85%	83%
	Area Normalization	80%	89%	70.5%
	Unary Features	26.5%	82%	55%
	Max Features	50%	73%	85%
	Euclidean	50%	65%	77%
	Unary Samples	50%	75%	85%

Appendix B: Model Optimization Results: Activation Function

Table B.1: MLP results for testing different activation functions with the top three normalization methods and one hidden layer

		Normalization Method					
		Max Samps	Area	Unary Samps	Max Samps	Area	Unary Samps
		[4 5 2]	[4 5 2]	[4 5 2]	[4 4 2]	[4 4 2]	[4 4 2]
Activation Function	compet	50%	50%	50%	50%	50%	50%
	elliotsig	94.5%	75.5%	65.5%	91.5%	84%	78%
	hardlim	83%	62%	83%	78%	60.5%	60%
	hardlims	71%	66.5%	78.5%	78%	61%	74.5%
	logsig	69.5%	80%	85.5%	60.5%	83%	82%
	netinv	50%	50%	73.5%	43%	50%	50%
	poslin	40%	50%	35%	50%	50%	45%
	purelin	45%	50%	45%	35%	50%	50%
	radbas	88%	88%	84%	91.5%	88%	83.5%
	radbasn	88.5%	83%	84.5%	93%	83%	71.5%
	satlin	86%	88%	78%	78%	83.5%	75%
	satlins	88%	84.5%	94%	85%	77%	80%
	softmax	50%	50%	50%	50%	50%	45%
	tansig	88.5%	81%	88%	91%	74.5%	81.5%
	tribas	97.5%	89.5%	84%	80.5%	84.5%	85.5%

Table B.2: MLP results for testing different activation functions with the top three normalization methods and one hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 1 2]	[4 1 2]	[4 1 2]
Activation Function	compet	50%	50%	50%
	elliotsig	77.5%	71.5%	40%
	hardlim	67%	75%	69.5%
	hardlims	67%	75%	55%
	logsig	25%	30%	67%
	netinv	40%	50%	50%
	poslin	50%	50%	50%
	purelin	40%	50%	46%
	radbas	74%	72.5%	69.5%
	radbasn	50%	50%	50%
	satlin	52%	26.5%	45%
	satlins	82.5%	40%	73%
	softmax	50%	50%	50%
	tansig	65%	75%	79%
	tribas	60%	70%	79.5%

Table B.3: MLP results for testing different activation functions with the top three normalization methods and two hidden layers

		Normalization Method					
		Max Samps	Area	Unary Samps	Max Samps	Area	Unary Samps
		[4 2 1 2]	[4 2 1 2]	[4 2 1 2]	[4 1 5 2]	[4 1 5 2]	[4 1 5 2]
Activation Function	compet	50%	50%	50%	50%	50%	50%
	elliotsig	50%	45.5%	50%	93.5%	92.5%	67%
	hardlim	50%	50%	50%	61.5%	62%	72%
	hardlims	53%	50%	47.5%	57.5%	70%	50%
	logsig	60%	35.5%	50%	76.5%	80.5%	78.5%
	netinv	50%	50%	50%	48%	50%	50%
	poslin	50%	50%	50%	45%	50%	50%
	purelin	50%	50%	42%	40%	50%	50%
	radbas	93%	94%	70.5%	78%	88.5%	82.5%
	radbasn	50%	50%	50%	50%	50%	50%
	satlin	67%	80%	73%	50%	78%	50%
	satlins	88%	95%	44.5%	88%	85.5%	78%
	softmax	50%	50%	50%	50%	50%	50%
	tansig	80%	75%	50%	95%	78%	83.5%
	tribas	88%	69.5%	75%	78%	75%	88.5%

Table B.4: MLP results for testing different activation functions with the top three normalization methods and two hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 1 2 2]	[4 1 2 2]	[4 1 2 2]
Activation Function	compet	50%	50%	50%
	elliotsig	77%	27%	65%
	hardlim	57.5%	74.5%	50%
	hardlims	55%	50%	59.5%
	logsig	77%	50%	50%
	netinv	50%	50%	50%
	poslin	50%	50%	50%
	purelin	50%	50%	50%
	radbas	72%	86%	83%
	radbasn	50%	50%	50%
	satlin	69.5%	69.5%	41%
	satlins	65.5%	75%	80.5%
	softmax	50%	50%	50%
	tansig	75%	71%	98.5%
	tribas	75%	74%	76%

Table B.5: MLP results for testing different activation functions with the top three normalization methods and three hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 1 3 4 2]	[4 1 3 4 2]	[4 1 3 4 2]
Activation Function	compet	50%	50%	50%
	elliotsig	95%	70%	73%
	hardlim	70.5%	75%	64.5%
	hardlims	50%	75%	65%
	logsig	71%	75%	79%
	netinv	50%	50%	50%
	poslin	50%	50%	50%
	purelin	40.5%	50%	45%
	radbas	86.5%	97.5%	81%
	radbasn	50%	50%	50%
	satlin	75%	86%	67%
	satlins	72%	37%	67%
	softmax	50%	50%	50%
	tansig	83%	84%	82%
	tribas	89%	69.5%	50%

Table B.6: MLP results for testing different activation functions with the top three normalization methods and three hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 4 1 3 2]	[4 4 1 3 2]	[4 4 1 3 2]
Activation Function	compet	50%	50%	50%
	elliotsig	83%	76%	43%
	hardlim	88%	55%	82%
	hardlims	76%	50%	50%
	logsig	78%	76.5%	67%
	netinv	39%	50%	50%
	poslin	50%	50%	50%
	purelin	50%	50%	45.5%
	radbas	93%	87.5%	100%
	radbasn	50%	50%	50%
	satlin	78%	95.5%	100%
	satlins	86%	95%	98%
	softmax	50%	50%	50%
	tansig	93%	89.5%	100%
	tribas	88%	90%	85%

Table B.7: MLP results for testing different activation functions with the top three normalization methods and three hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 5 4 3 2]	[4 5 4 3 2]	[4 5 4 3 2]
Activation Function	compet	50%	50%	50%
	elliotsig	95%	91%	100%
	hardlim	68%	55%	55%
	hardlims	60%	73%	50%
	logsig	84.5%	75%	79.5%
	netinv	50%	72.5%	68%
	poslin	40%	50%	50%
	purelin	35.5%	50%	64.5%
	radbas	93%	100%	86.5%
	radbasn	88.5%	99.5%	100%
	satlin	93%	89%	90%
	satlins	90.5%	97%	87%
	softmax	50%	50%	50%
	tansig	97.5%	97%	100%
	tribas	98.5%	97.5%	96.5%

Table B.8: MLP results for testing different activation functions with the top three normalization methods and four hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 2 4 5 2 2]	[4 2 4 5 2 2]	[4 2 4 5 2 2]
Activation Function	compet	50%	50%	50%
	elliotsig	50%	71%	63%
	hardlim	50%	50%	60%
	hardlims	62%	50%	55%
	logsig	45.5%	45%	83%
	netinv	50%	50%	50%
	poslin	50%	50%	50%
	purelin	49.5%	50%	55.5%
	radbas	86.5%	100%	98.5%
	radbasn	91%	75%	86%
	satlin	65.5%	92.5%	85.5%
	satlins	75%	50%	82.5%
	softmax	50%	50%	50%
	tansig	50%	86%	67%
	tribas	88%	98%	93%

Table B.9: MLP results for testing different activation functions with the top three normalization methods and four hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 3 1 2 2 2]	[4 3 1 2 2 2]	[4 3 1 2 2 2]
Activation Function	compet	50%	50%	50%
	elliotsig	45%	45%	35%
	hardlim	50%	50%	50%
	hardlims	50%	50%	50%
	logsig	65%	75%	50%
	netinv	50%	50%	79.5%
	poslin	50%	50%	50%
	purelin	50%	50%	50%
	radbas	88%	87%	84.5%
	radbasn	50%	50%	50%
	satlin	50%	50%	55%
	satlins	93.5%	55%	55%
	softmax	50%	50%	50%
	tansig	67%	73%	55%
	tribas	83%	82%	95%

Table B.10: MLP results for testing different activation functions with the top three normalization methods and four hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 5 5 4 1 2]	[4 5 5 4 1 2]	[4 5 5 4 1 2]
Activation Function	compet	50%	50%	50%
	elliotsig	50%	80.5%	67%
	ehardlim	50%	58%	57%
	ehardlims	75%	50%	55%
	elogsig	50%	85%	50%
	enetinv	50%	50%	35%
	eposlin	50%	50%	50%
	epurelin	37%	50%	49.5%
	eradbass	90%	91%	89.5%
	eradbassn	50%	50%	50%
	esatlin	92.5%	66.5%	95%
	esatlins	98%	87.5%	65%
	esoftmax	50%	50%	50%
	etansig	100%	90%	99%
	etribas	77%	81.5%	90%

Table B.11: MLP results for testing different activation functions with the top three normalization methods and five hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 5 4 4 1 5 2]	[4 5 4 4 1 5 2]	[4 5 4 4 1 5 2]
Activation Function	compet	50%	50%	50%
	elliotsig	97.5%	60%	64%
	hardlim	65%	50%	95%
	hardlims	50%	50%	72.5%
	logsig	65%	95%	100%
	netinv	45%	50%	56%
	poslin	50%	50%	50%
	purelin	45%	50%	45%
	radbas	90%	90%	82.5%
	radbasn	50%	50%	50%
	satlin	78%	50%	86.5%
	satlins	76%	68%	86.5%
	softmax	50%	50%	50%
	tansig	100%	100%	99.5%
	tribas	81.5%	92.5%	70%

Table B.12: MLP results for testing different activation functions with the top three normalization methods and five hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 4 2 5 3 5 2]	[4 4 2 5 3 5 2]	[4 4 2 5 3 5 2]
Activation Function	compet	50%	50%	50%
	elliotsig	65%	75.5%	50%
	hardlim	50%	50%	65%
	hardlims	59.5%	60%	50%
	logsig	89%	97.5%	70%
	netinv	50%	50%	50%
	poslin	50%	50%	50%
	purelin	50%	50%	63%
	radbas	90%	89.5%	78%
	radbasn	50%	98.5%	50%
	satlin	69.5%	94.5%	50%
	satlins	88%	80%	98%
	softmax	50%	50%	50%
	tansig	50%	92.5%	97%
	tribas	87.5%	50%	82.5%

Table B.13: MLP results for testing different activation functions with the top three normalization methods and five hidden layers

		Normalization Method		
		Max Samps	Area	Unary Samps
		[4 5 5 4 3 4 2]	[4 5 5 4 3 4 2]	[4 5 5 4 3 4 2]
Activation Function	compet	50%	50%	50%
	elliotsig	67%	42%	99.5%
	hardlim	69%	72.5%	55.5%
	hardlims	70%	80%	50%
	logsig	87.5%	75%	97%
	netinv	50%	50%	50%
	poslin	50%	50%	50%
	purelin	62%	50%	70%
	radbas	95%	95%	99%
	radbasn	93%	50%	100%
	satlin	88%	99%	93.5%
	satlins	92.5%	97.5%	100%
	softmax	50%	50%	50%
	tansig	93%	97.5%	50%
	tribas	93%	92.5%	87%

Appendix C: Model Optimization Results: Training Function

Table C.1: MLP results from testing different training functions with one and two hidden layers. All results are calculated using Equal Weighted Accuracy.

		Topology					
		[4 1 2]	[4 3 2]	[4 5 2]	[4 4 2 2]	[4 4 5 2]	[4 5 4 2]
Training Function	traingdm	83%	92.5%	90%	85%	97%	89.5%
	traingdx	74.5%	100%	88%	100%	99.5%	100%
	trainlm	95.5%	100%	88%	95%	97%	98.5%

Table C.2: MLP results from testing different training functions with three hidden layers. All results are calculated using Equal Weighted Accuracy.

		Normalization Method		
		[4 1 1 5 2]	[4 5 2 2 2]	[4 5 4 2 2]
Training Function	traingdm	70%	95.5%	94%
	traingdx	72.5%	90.5%	95%
	trainlm	95%	96.5%	95.5%

Table C.3: MLP results from testing different training functions with four hidden layers.

All results are calculated using Equal Weighted Accuracy.

		Normalization Method		
		[4 4 2 5 5 2]	[4 4 4 3 5 2]	[4 4 4 5 4 2]
Training Function	traingdm	96.5%	70%	95%
	traingdx	95%	97%	88%
	trainlm	98%	100%	100%

Table C.4: MLP results from testing different training functions with five hidden layers.

All results are calculated using Equal Weighted Accuracy.

		Normalization Method		
		[4 1 5 2 2 5 2]	[4 2 5 5 1 1 2]	[4 4 2 2 5 4 2]
Training Function	traingdm	67.5%	92%	57%
	traingdx	87.5%	90%	98%
	trainlm	78%	97%	99.5%

Appendix D: MLP and SVM Results for Testing All Three Groups

Table D.1: Neural network results for the first four POI of group one with the optimized configuration.

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	96%	100%	100%	100%	100%	100%
	POI 2	96%	92%	100%	97.5%	100%	91.5%
	POI 3	90%	83.5%	100%	81.5%	99.5%	99.5%
	POI 4	85%	100%	100%	95%	73.5%	94.5%

Table D.2: Neural network results for the first four POI of group two with the optimized configuration.

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	93.8%	95%	88.1%	91.3%	98.1%	93.1%
	POI 2	100%	100%	93.8%	98.1%	100%	100%
	POI 3	100%	100%	100%	100%	100%	100%
	POI 4	94.4%	85.6%	80.6%	81.3%	94.4%	91.3%

Table D.3: Neural network results for the first four POI of group three with the optimized configuration.

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	100%	100%	83.3%	68.3%	93.3%	100%
	POI 2	95%	83.3%	65%	50%	50%	93.3%
	POI 3	88.3%	90%	53.3%	80%	98.3%	80%
	POI 4	100%	70%	50%	83.3%	86.7%	100%

Table D.4: SVM Results for Group 1 with Linear Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	63%	63%	63%	63%	63%	63%
	POI 2	50%	50%	51%	50%	51%	51%
	POI 3	67%	67%	67%	67%	67%	72.5%
	POI 4	54.5%	54.5%	54.5%	55%	54.5%	54.5%

Table D.5: SVM Results for Group 2 with Linear Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 2	POI 1	62.5%	62.5%	62.5%	62.5%	62.5%	62.5%
	POI 2	56.3%	56.3%	56.3%	56.3%	56.3%	56.3%
	POI 3	87.5%	87.5%	87.5%	87.5%	87.5%	87.5%
	POI 4	56.3%	56.3%	56.3%	56.3%	56.3%	56.3%

Table D.6: SVM Results for Group 3 with Linear Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 3	POI 1	68.3%	71.7%	71.7%	73.3%	73.3%	71.7%
	POI 2	83.3%	83.3%	83.3%	83.3%	83.3%	83.3%
	POI 3	66.7%	66.7%	66.7%	66.7%	66.7%	66.7%
	POI 4	68.3%	66.7%	66.7%	68.3%	66.7%	66.7%

Table D.7: SVM Results for Group 1 with Radial Basis Function Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	63%	63%	63%	63%	63%	63%
	POI 2	50%	51%	50%	51%	51%	50%
	POI 3	67%	67%	67%	67%	67%	72.5%
	POI 4	54.5%	54.5%	54.5%	54.5%	54.5%	54.5%

Table D.8: SVM Results for Group 2 with Radial Basis Function Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 2	POI 1	62.5%	62.5%	62.5%	62.5%	62.5%	62.5%
	POI 2	56.3%	56.3%	56.3%	56.3%	56.3%	56.3%
	POI 3	87.5%	87.5%	87.5%	87.5%	87.5%	87.5%
	POI 4	56.3%	56.3%	56.3%	56.3%	56.3%	56.3%

Table D.9: SVM Results for Group 3 with Radial Basis Function Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 3	POI 1	73.3%	71.7%	70%	70%	73.3%	71.7%
	POI 2	83.3%	83.3%	83.3%	83.3%	83.3%	83.3%
	POI 3	66.7%	66.7%	66.7%	66.7%	66.7%	66.7%
	POI 4	68.3%	66.7%	66.7%	68.3%	66.7%	66.7%

Table D.10: SVM Results for Group 1 with Polynomial Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	73%	73%	73%	66%	72.5%	72%
	POI 2	55%	55%	55%	55%	55.5%	59.5%
	POI 3	90%	90%	90%	90%	90%	90%
	POI 4	51.5%	56%	61.5%	61.5%	56.5%	62.5%

Table D.11: SVM Results for Group 2 with Polynomial Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 2	POI 1	91.9%	91.3%	91.9%	85%	91.3%	90.6%
	POI 2	87.5%	87.5%	87.5%	87.5%	87.5%	87.5%
	POI 3	93.8%	93.8%	93.8%	93.8%	93.8%	93.8%
	POI 4	81.3%	81.3%	81.3%	81.3%	81.3%	81.3%

Table D.12: SVM Results for Group 3 with Polynomial Kernel

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 3	POI 1	68.3%	68.3%	68.3%	68.3%	65%	68.3%
	POI 2	83.3%	83.3%	83.3%	83.3%	83.3%	83.3%
	POI 3	66.7%	66.7%	66.7%	66.7%	66.7%	66.7%
	POI 4	66.7%	66.7%	66.7%	68.3%	66.7%	66.7%

Appendix E: MLP Results of Testing with Gaussian Noise

Figure E.1: MLP results of training with non noisy data and testing with noisy data for Group 1

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1	POI 1	45.5%	55%	44.5%	50%	53%	49.5%
	POI 2	50%	49%	48.5%	47%	47.5%	50.5%
	POI 3	65%	66.5%	50%	60.5%	57%	65%
	POI 4	43%	41.5%	35.5%	48.5%	52.5%	50%

Figure E.2: MLP results of training with non noisy data and testing with noisy data for Group 2

		Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 2	POI 1	66.9%	36.9%	42.5%	51.9%	45%	63.1%
	POI 2	34.4%	41.3%	48.8%	49.4%	49.4%	47.5%
	POI 3	60.6%	46.3%	60.6%	44.4%	37.5%	47.5%
	POI 4	49.4%	54.4%	52.5%	51.2%	56.3%	39.4%

Figure E.3: MLP results of training with non noisy and testing with noisy data data for Group 3

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 3 POI 1	45%	50%	53.3%	35%	46.7%	50%
POI 2	61.7%	53.3%	51.7%	48.3%	61.7%	46.7%
POI 3	48.3%	53.3%	55%	50%	35%	45%
POI 4	63.3%	45%	45%	60%	61.7%	53.3%

Figure E.4: MLP results of training and testing with noisy data for Group 1

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 1 POI 1	81%	77%	80.5%	82.5%	94%	86.5%
POI 2	83.5%	89%	84.5%	82.5%	84.5%	93%
POI 3	95%	85%	88.5%	85.5%	97.5%	94%
POI 4	90%	91.5%	86.5%	90.5%	89.5%	91%

Figure E.5: MLP results of training and testing with noisy data for Group 2

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
Group 2 POI 1	86.9%	95%	86.3%	91.9%	75.6%	91.3%
POI 2	85.6%	87.5%	73.1%	78.8%	79.4%	91.9%
POI 3	94.4%	94.4%	71.9%	93.8%	91.9%	93.8%
POI 4	92.5%	69.4%	79.4%	91.9%	66.9%	86.9%

Figure E.6: MLP results of training and testing with noisy data for Group 3

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	General
POI 1	80%	88.3%	61.7%	66.7%	63.3%	75%
POI 2	83.3%	61.7%	83.3%	76.7%	80%	90%
POI 3	55%	75%	81.7%	78.3%	98.3%	88.3%
POI 4	91.7%	96.7%	83.3%	73.3%	85%	91.7%

Group 3

Appendix F: HSI Results



Figure F.1: Result of using the optimal feature set with l^2 norm.

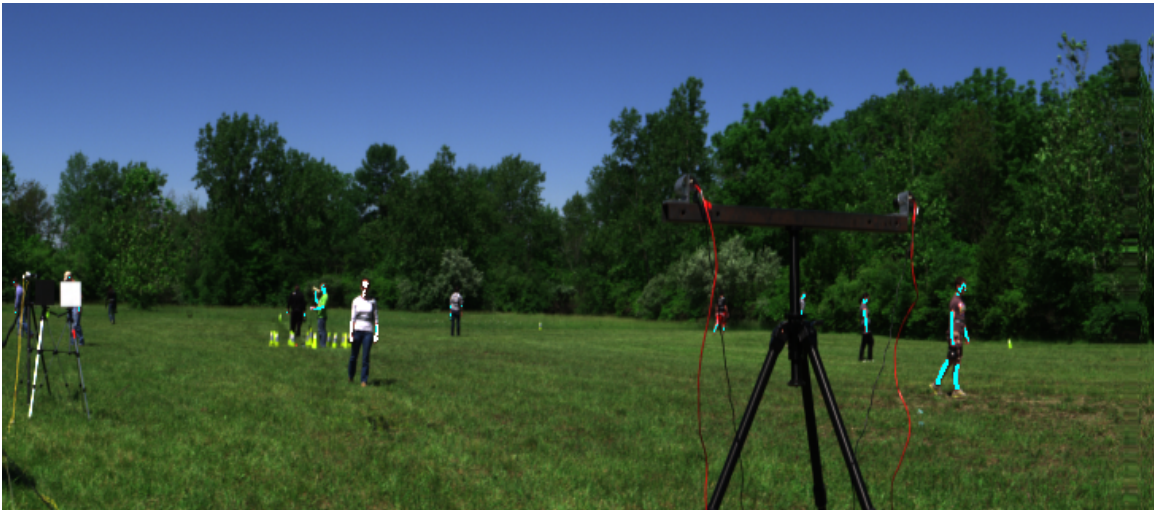


Figure F.2: Result of using the optimal feature set with normalization method 6

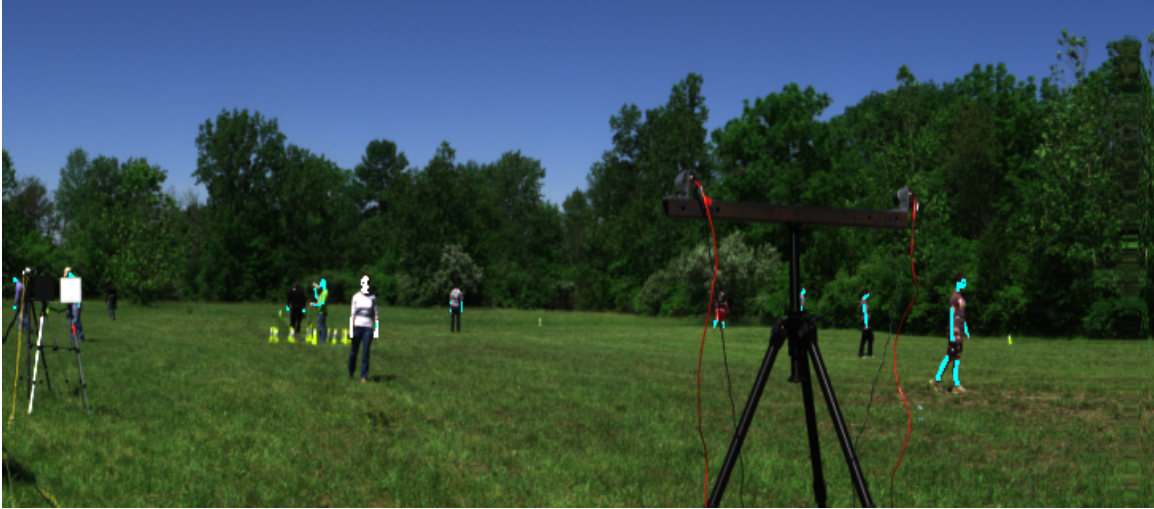


Figure F.3: Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with normalization method 1.



Figure F.4: Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with l^2 norm.

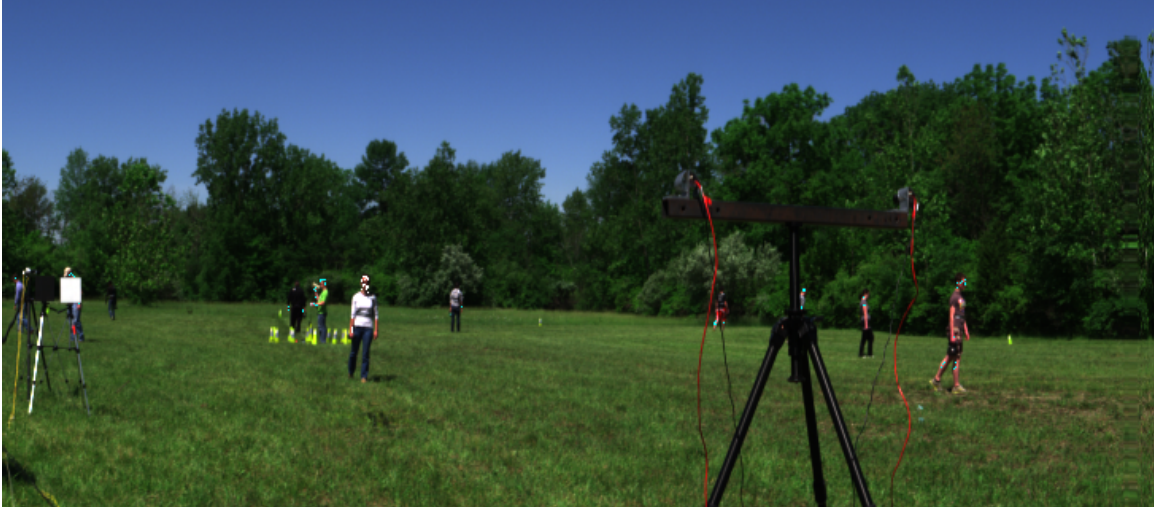


Figure F.5: Result of using the optimal feature set with normalization 1 and a bigger network size of one hidden layer and ten hidden nodes.

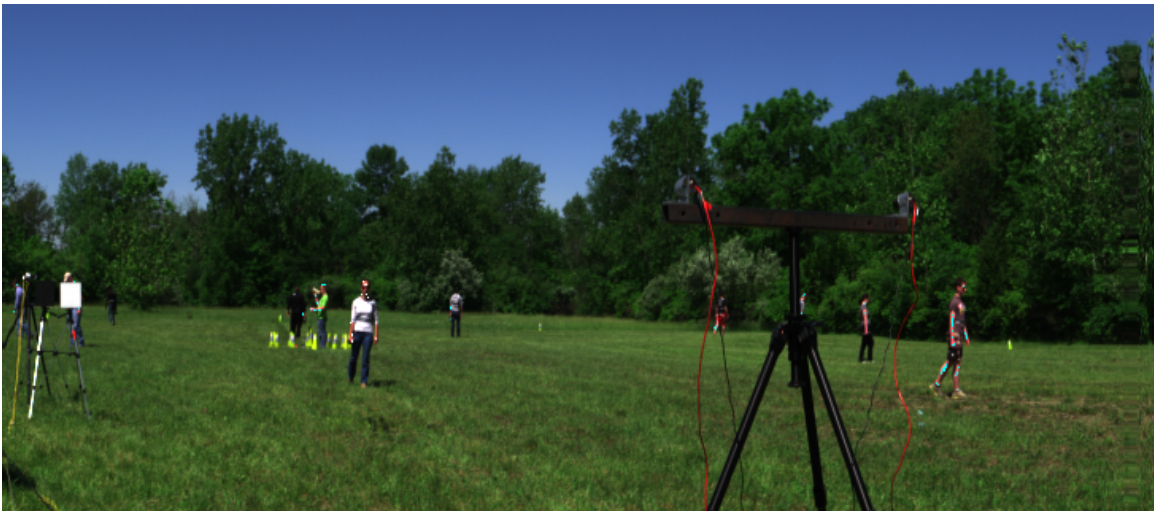


Figure F.6: Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm) with normalization method 1 network size of one hidden layer and ten hidden nodes.

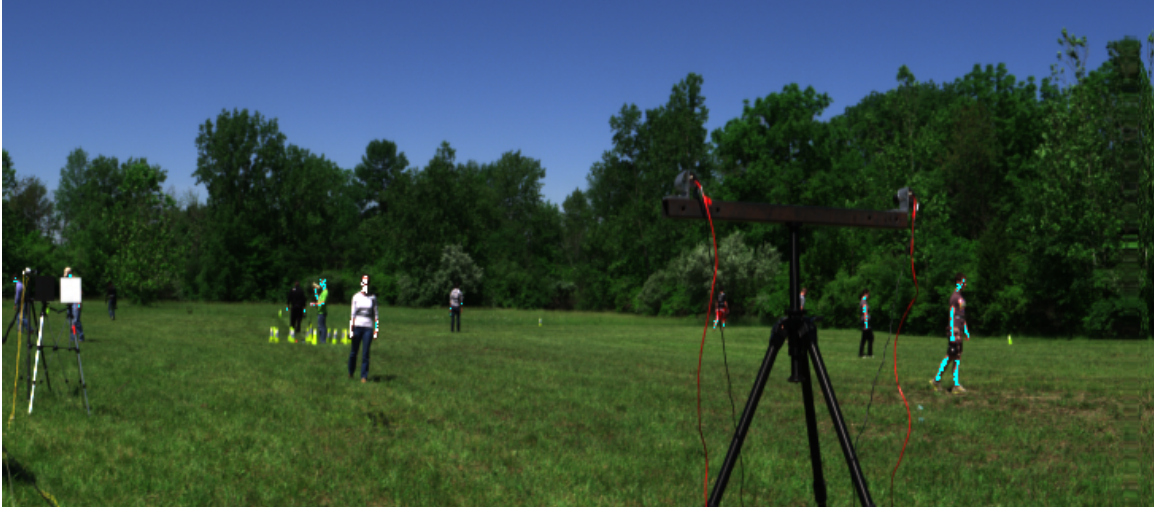


Figure F.7: Result of using the new feature set (804 nm, 1137 nm, 891 nm, 1343 nm, 1285 nm, 1255 nm, 1564 nm, and 1695 nm) with normalization method 1 network size of one hidden layer and ten hidden nodes.

Bibliography

- [1] Zhihong Pan, G.E. Healey, M. Prasad, and B.J. Tromberg, “Face recognition in hyperspectral images,” in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, June 2003, vol. 1, pp. I-334–I-339 vol.1.
- [2] Lindsay Cain, “Feature selection on hyperspectral data for dismount detection,” M.S. thesis, Air Force Institute of Technology, 2014.
- [3] Pathak, Madhu A. and Jimbow, Kowichi and Szabo, George and Fitzpatrick, Thomas B., “Sunlight and melanin pigmentation,” *Photochemical and Photobiological Reviews*, pp. 211–239, 1976.
- [4] James T. Enns, *The Thinking Eye, the Seeing Brain*, W. W. Norton & Company, Inc., 2004.
- [5] N. Hirose and M. Fukuda, “Numerical wind tunnel (nwt) and cfd research at national aerospace laboratory,” in *High Performance Computing on the Information Superhighway, 1997. HPC Asia '97*, Apr 1997, pp. 99–103.
- [6] S. Varadarajan, “System x: building the virginia tech supercomputer,” in *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on*, Oct 2004, pp. 2–.
- [7] Anil Jain, Lin Hong, and Sharath Pankanti, “Biometric identification,” in *COMMUNICATIONS OF THE ACM*, Feb 2000, pp. 91–98.
- [8] A.S. Nunez and Michael J. Mendenhall, “Detection of human skin in near infrared hyperspectral imagery,” in *Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International*, July 2008, vol. 2, pp. II-621–II-624.
- [9] A.S. Nunez, Michael J. Mendenhall, and K. Gross, “Melanosome level estimation in human skin from hyperspectral imagery,” in *Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, 2009. WHISPERS '09. First Workshop on*, Aug 2009, pp. 1–4.
- [10] Gerard J. Tortora and Bryan Derrickson, *Principles of Anatomy and Physiology*, chapter 5, John Wiley & Sons, Inc, 2009.
- [11] Elaine N. Marieb and Katja Hoehn, *Human Anatomy & Physiology*, chapter 5, Pearson, 2013.
- [12] M.T. Eismann, A.D. Stocker, and N.M. Nasrabadi, “Automated hyperspectral cueing for civilian search and rescue,” *Proceedings of the IEEE*, vol. 97, no. 6, pp. 1031–1055, June 2009.

- [13] Per Nostell, Arne Roos, and D. Ronnow, “Single-beam integrating sphere spectrophotometer for reflectance and transmittance measurements versus angle of incidence in the solar wavelength range on diffuse and specular samples,” *Review of Scientific Instruments*, vol. 70, no. 5, pp. 2481–2494, May 1999.
- [14] Teri M. Giacomoni P., Mammone T., “Gender-linked differences in human skin,” *Journal of Dermatological Science*, vol. 55, no. 3, pp. 144–149, Sept 2009.
- [15] Abel S. Nunez, *A Physical Model of Human Skin and Its Application for Search and Rescue*, Ph.D. thesis, Air Force Institute of Technology, 2009.
- [16] “Feature selection,” <http://featureselection.asu.edu/>, Accessed: 2015-01-03.
- [17] J.M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N.M. Nasrabadi, and J. Chanussot, “Hyperspectral remote sensing data analysis and future challenges,” *Geoscience and Remote Sensing Magazine, IEEE*, vol. 1, no. 2, pp. 6–36, June 2013.
- [18] Randall B. Smith, “Introduction to hyperspectral imaging,” <http://www.microimages.com/documentation/Tutorials/hyrspec.pdf>, January 2012.
- [19] Marcus Borengasser, William S. Hungate, and Russell Watkins, *Hyperspectral Remote Sensing Principles and Applications*, CRC Press, 2008.
- [20] Hans Georg Schaathun, *Machine Learning in Image Steganalysis*, chapter 13, John Wiley & Sons, Ltd, 2012.
- [21] Isabelle Guyon and Andr Elisseeff, “An introduction to variable and feature selection,” *Journal of Machine Learning Research* 3.
- [22] L. Hanzo, R. Maunder, J. Wang, and L. Yang, *Near-Capacity Variable-Length Coding: Regular and EXIT-Chart-Aided Irregular Designs*, chapter 1, Wiley-IEEE Press, 2011.
- [23] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.
- [24] Kenneth Ward Church and Patrick Hanks, “Word association norms, mutual information, and lexicography,” *Computational Linguistics*, vol. 16, no. 1, March 1990.
- [25] Lei Yu and Huan Liu, “Feature selection for high-dimensional data: A fast correlation-based filter solution,” in *Twentieth International Conference on Machine Learning*, 2003.
- [26] I.T. Jolliffe, *Principal Component Analysis*, Springer, 1986.
- [27] D. Hudson and M. Cohen, *Neural Networks and Artificial Intelligence for Biomedical Engineering*, chapter 4, Wiley-IEEE Press, 2000.

- [28] D. Hudson and M. Cohen, *Neural Networks and Artificial Intelligence for Biomedical Engineering*, chapter Supervised Learning, pp. 59–77, Wiley-IEEE Press, 2000.
- [29] Simon Haykin, *Neural Networks and Learning Machines*, Pearson Education, Inc., Upper Saddle River, New Jersey, 2009.
- [30] Simon Haykin, *Neural Networks and Learning Machines*, chapter 4, Pearson Prentice Hall, 2009.
- [31] Léon Bottou, “Online algorithms and stochastic approximations,” in *Online Learning and Neural Networks*, David Saad, Ed. Cambridge University Press, Cambridge, UK, 1998, revised, oct 2012.
- [32] Hao Yu and Bogdan M Wilamowski, “Levenberg-marquardt training,” *The Industrial Electronics Handbook*, vol. 5, pp. 1–15, 2011.
- [33] M.T. Hagan and M.B. Menhaj, “Training feedforward networks with the marquardt algorithm,” *Neural Networks, IEEE Transactions on*, vol. 5, no. 6, pp. 989–993, Nov 1994.
- [34] John Shawe-Taylor Nello Cristianini, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, chapter 6, Cambridge University Press, 2000.
- [35] William H. Press, Saul A. Teukolsky, William T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, chapter Section 16.5. Support Vector Machines, pp. 883–888, New York: Cambridge University Press, 2007.
- [36] Nina G. Jablonski and George Chaplin, “The evolution of human skin coloration,” *Journal of Human Evolution*, 2000.
- [37] Esteban J. Parra, “Human pigmentation variation: Evolution, genetic basis, and implications for public health,” *Yearbook of Physical Anthropology*, pp. 85–105, 2007.
- [38] Engin Karabudak, Wendel Wohlleben, and Helmut Cölfen, “Investigation of β -carotene-gelatin composite particles with a multiwavelength uv/vis detector for the analytical ultracentrifuge,” *European Biophysics Journal*, 2009.
- [39] “Optical absorption of hemoglobin,”
<http://modis.gsfc.nasa.gov/about/specifications.php>, Accessed: 2015-01-03.
- [40] “Fitzpatrick skin quiz,”
<http://www.skincancer.org/prevention/are-you-at-risk/fitzpatrick-skin-quiz>,
 Accessed: 2015-01-03.

- [41] Mark A. Haidekker, *Advanced Biomedical Image Analysis*, chapter 11, John Wiley & Sons, Inc., 2011.
- [42] Atam P. Dhawan, *Medical Image Analysis*, chapter 12, Institute of Electrical and Electronics Engineers, Inc., 2011.
- [43] Cong Phuoc Huynh and A. Robles-Kelly, "Hyperspectral imaging for skin recognition and biometrics," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*, Sept 2010, pp. 2325–2328.
- [44] "Asd fieldspec spectroradiometer,"
<http://www.asdi.com/products/fieldspec-spectroradiometers>, Accessed: 2015-01-03.
- [45] "Dual datasheet,"
http://www.specim.fi/files/pdf/aisa/datasheets/Dual_datasheet_ver1-2012.pdf,
Accessed: 2015-01-03.
- [46] "*rs*³ spectral acquisition software,"
<http://www.asdi.com/products/spectroscopy-software/rs3>, Accessed: 2015-01-03.
- [47] "Fieldspec 3 support central," <http://support.asdi.com/Products/Products.aspx>,
Accessed: 2015-01-03.
- [48] "Envi 5,"
<http://www.exelisvis.com/Default.aspx?alias=www.exelisvis.com/envi-5&>,
Accessed: 2015-01-03.
- [49] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten, "The weka data mining software: An update," *SIGKDD Explorations*, vol. 11, 2009.
- [50] Robert A. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 1, pp. 295–307, 1988.
- [51] "Skin optics summary," <http://omlc.org/news/jan98/skinoptics.html>, Accessed: 2015-01-03.
- [52] "Train and apply multilayer neural networks," <http://www.mathworks.com/help/nnet/ug/train-and-apply-multilayer-neural-networks.html>, Accessed: 2015-01-03.
- [53] Andrew P. Beisley, "Spectral detection of human skin in vis-swir hyperspectral imagery without radiometric calibration," M.S. thesis, Air Force Institute of Technology, 2012.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY) 26-03-2015		2. REPORT TYPE Master's Thesis		3. DATES COVERED (From — To) Oct 2013–Mar 2015			
4. TITLE AND SUBTITLE Hyperspectral Based Skin Detection for Person of Interest Identification				5a. CONTRACT NUMBER			
				5b. GRANT NUMBER			
				5c. PROGRAM ELEMENT NUMBER			
				5d. PROJECT NUMBER 15G258			
				5e. TASK NUMBER			
6. AUTHOR(S) Tang, Khoa A., Captain, USAF				5f. WORK UNIT NUMBER			
				7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB, OH 45433-7765			
				8. PERFORMING ORGANIZATION REPORT NUMBER AFIT-ENG-MS-15-M-031			
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Research Laboratories 711th Human Performance Wing Lead Scientist, Dr. Darrell F. Lochtefeld 2800 Q street, Bldg 824 Wright-Patterson AFB, OH 45433 (937) 255-2570 darrell.lochtefeld@us.af.mil				10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/RHXBA			
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)			
12. DISTRIBUTION / AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A: APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED							
13. SUPPLEMENTARY NOTES This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.							
14. ABSTRACT An optimal skin identification system can have great impact in areas such as security and surveillance. In the field of security, skin identification can augment the effectiveness of other biometric security systems such as facial recognition and fingerprint identification [1]. In the case of surveillance, a real-time skin identification system will be extremely useful at tracking POI. Detecting and tracking using a skin identification technique provides positive, real time acquisition of people as they exit a building and prevents inadvertent track loss in a large crowd. This thesis presents the results of an ANN that is created in MATLAB [®] using the Neural Network Toolbox to identify a POI based on their skin spectral data. A baseline model is used with the optimal feature set identified by Cain [2]. The baseline model is then modified and optimized to maximize a ANN capability to identify a POI. Gaussian noise is calculated and added to the data sets to simulate atmospheric noise of a real world scene. The simulated atmospheric noise reduced the neural network's accuracy by 14%. The neural network model is tested with real HSI images to verify the applicability of the ANN to identify a POI. The results for HSI testing are between 40-60% due to the illumination sources' angle of incidence and the standard deviation of skin reflectance associated with differing skin locations on the body.							
15. SUBJECT TERMS hyperspectral, skin, identification, neural network							
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON		
a. REPORT	b. ABSTRACT	c. THIS PAGE			Lt Col Jeffrey D. Clark, PhD (ENG)		
U	U	U	UU	113	19b. TELEPHONE NUMBER (include area code) (937) 255-3636 x4614 Jeffrey.Clark@afit.edu		