

AN APPROACH TO DETECT HUMAN FACIAL SKIN IN THERMAL IMAGERY

by

Balvinder Kaur
A Thesis
Submitted to
Graduate Faculty
of
George Mason University
in Partial Fulfillment of
The Requirements for the Degree
of
Master of Science
Electrical Engineering

Committee:



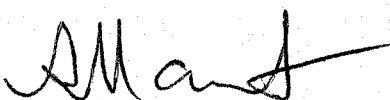
Dr. Jill K. Nelson, Thesis Director



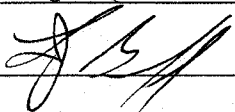
Dr. Andre Manitus, Committee Member



Dr. Vasiliki N. Ikonomidou, Committee Member



Dr. Andre Manitus, Department Chair



Dr. Lloyd J. Griffiths, Dean, Volgenau School of Engineering

Date: 6/6/12

Summer Semester 2012
George Mason University
Fairfax, VA

An Approach to Detect Human Facial Skin in Thermal Imagery

A Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science at George Mason University

By

Balvinder Kaur
Bachelor of Science
George Mason University, 1999

Advisor: Jill K. Nelson, Assistant Professor
Department of Electrical and Computer Engineering

Summer Semester 2012
George Mason University
Fairfax, VA

Copyright © 2012 by Balvinder Kaur
All Rights Reserved

Dedication

I dedicate this thesis work to my loving husband Harpreet Singh and our handsome son Taaren Veer Singh who encouraged and supported me throughout this journey of the master's degree program. Thanks are due to my father-in-law Mr. Jagjit Singh, mother-in-law, Mrs. Narinder Kaur, my father, Mr. Balbir Singh, and my mother, Mrs. Surjit Kaur, whose support and interest made the work much easier.

Acknowledgements

Dr. Jill Nelson, my advisor provided me invaluable help and guidance throughout the master's research. Her knowledge, flexibility, patience, kindness, mentoring, and friendship made me helpless to ask for more. I also gratefully thank and acknowledge my committee members, Dr. Andre Manitius and Dr. Vasiliki Ikonomidou, for their help, support, and guidance. Very special thanks are owed to my chain of command starting with Dr. A. Fenner Milton for encouraging higher education, Dr. Keith Krapels, Ms. Susan Harkrider, Dr. J. Andrew Hutchinson, Mr. Jonathan Hixson, my colleges and friends for guiding and supporting me throughout my master's program. Finally, I want to thank Dr. Barbara O'Kane for being my mentor, associate, colleague, friend, and an individual who seeded this research idea in my brain with a question, "Can you detect skin?"

Table of Contents

	Page
List of Tables _____	vii
List of Figures _____	viii
Abstract _____	xiii
1 Introduction and Background _____	1
1.1 Relevant Studies _____	3
1.2 Motivation _____	5
1.3 Proposed Methodology and Solution _____	7
1.4 Proposed Candidate Algorithms _____	9
2 Algorithms for Skin Detection _____	10
2.1 Pre-processing _____	11
2.1.1 Normalize Data _____	11
2.1.2 Smooth Data _____	12
3 Preparing Ground Truth (GT) Data _____	18
3.1 Automatic Skin-chip Extraction _____	18
3.2 GLCM Feature Generation _____	21
3.2.1 Contrast _____	24
3.2.2 Entropy _____	24
3.2.3 Homogeneity _____	25
3.2.4 Correlation _____	25
3.2.5 Mean _____	26
4 Applying PCA for Data Reduction _____	28
4.1 Feature Reduction _____	29
4.2 Fused Image Generation _____	37
5 Classification for Skin Detection _____	39
5.1 Classification with Adaptive Optimized Least-Mean-Square (LMS) _____	39
5.2 Classification with Principal Component Analysis (PCA) _____	46

5.3	Classification with Linear Discriminant Analysis (LDA)	50
5.4	Post-processing	56
6	Validation and Data Analysis	58
6.1	Algorithm Validation Procedure	58
6.2	Data Analysis	62
6.2.1	Further Analysis of the LDA performance	66
7	Conclusions	72
7.1	Open Questions	73
7.2	Productivity of the Research	74
	References	75

List of Tables

Table	Page
Table 1: Paired t-test results between LMS, PCA and LDA algorithms results.....	66
Table 2: Between-class and within-class scatter values for skin and non-skin classes ...	70

List of Figures

Figure	Page
Figure 1: Facial skin pore characteristics: 1) Subject with no response; 2) Subject with high response; 3) Digital zoom of active pore or sweat gland ; 4) X-axis thermal cross section ; 5) Y-axis thermal cross section.....	6
Figure 2: Facial non-skin and skin texture characteristics (Gray Level Co-occurrence Matrix) in 11x11 window: Non-skin locations (left) and Skin locations.....	8
Figure 3: Main algorithm flow chart.....	10
Figure 4: Pre-processing algorithm flow chart.....	11
Figure 5: Dilation process: 1) Element A; 2) Square structuring element B; 3) Dilation of A by B, shown shaded.....	14
Figure 6: Erosion process: 1) Element A; 2) Square structuring element B; 3) Erosion of A by B, shown shaded.....	15
Figure 7: Opening process: 1) Element A; 2) Disk structuring element B; 3) Opening of A by B, shown shaded light blue.....	16
Figure 8: Flow chart to prepare Ground Truth (GT).....	18

Figure 9: Auto skin detector process: 1) Original image; 2) Image after smoothing; 3) Image with skin and edges ; 4) All skin image; 5) Skin chip.....	19
Figure 10: GLCM kernel mapping.....	22
Figure 11: GLCM kernel: 1) Sample original chip; 2) Transformation process (N to 1 mapping) [15]; 3) GLCM kernel.....	22
Figure 12: Sample data representing signal and noise variability.....	28
Figure 13: For selected skin portion (top left, original), Gray Level Co-occurrence Matrix (GLCM) driven texture feature images	29
Figure 14: Mean image.....	31
Figure 15: Eigenvalue distribution.....	33
Figure 16: Eigenvectors plotted on top of original data for 3 features: 1) Overview of data and 2) Zoomed in three orthogonal components.....	33
Figure 17: Eigenimages corresponding to the K eigenvectors - 1) Eigenimage#1; 2) Eigenimage#2; 3) Eigenimage#3; 4) Eigenimage#4.....	34
Figure 18: Process to construct a feature image.....	35
Figure 19: PCA selected GLCM feature images- 1) Contrast; 2) Correlation; 3) Energy; 4) Homogeneity.....	36

Figure 20: Flow chart to generate a fused image.....	38
Figure 21: Fused image results: 1) Original image and 2) Fused image.....	38
Figure 22: An adaptive threshold algorithm based on LMS optimization	40
Figure 23: Adaptive optimized LMS algorithm parameters: 1) Learning curve for the MSE over k iterations and 2) Threshold weight convergence over k iterations.....	45
Figure 24: Optimized adaptive LMS results: 1) Original image, 2) Filtered image, and 3) Image after threshold (Skin only image).....	45
Figure 25: Manually selected skin (green) and non-skin samples (red) to construct two classes.....	47
Figure 26: PCA classification results; 1) Image, 2) PCA transformed image, and 3) PCA classified skin mask.....	50
Figure 27: Feature data: Sample test data (blue), Skin class (green), and Non-skin class (red)	52
Figure 28: LDA results: 1) LDA generated vector projection and 2) Classes data projection onto LDA generated vector.....	55
Figure 29: LDA Generated results: 1) Data pixels projection onto the LDA-generated vector and 2) Skin mask after LDA classification.....	56

Figure 30: Sample results after post-processing - 1) Original image; 2) Image after adaptive optimized LMS threshold; 3) Skin-only image after post-processing; 4) Image after PCA classification; 5) Skin-only image after post-processing; 6) Image after LDA classification; 7) Skin-only image after post-processing;57

Figure 31: Validation procedure flow diagram: 1) Algorithm to generate skin-only image; 2) Generate manual ground truth; 3) GT and algorithm output comparison.....59

Figure 32: Ground truth data: 1) Manual ground truth (non-skin in blue and skin in green stars); 2) Image with only non-skin locations; 3) Image with only skin locations..60

Figure 33: 1) GT skin image 2) Algorithm output non-skin image with non-skin in white; 3) Algorithm missed non-skin locations.....61

Figure 34: 1) GT non-skin image 2) Algorithm output skin-only image with skin in white; 3) Algorithm missed non-skin locations.....61

Figure 35: 1) GT non-skin image 2) Algorithm output skin-only image with skin in white; 3) Algorithm missed non-skin locations; 4) GT skin image 5) Algorithm output non-skin image with non-skin in white; 6) Algorithm missed non-skin locations.....62

Figure 36: Actual true detections with LMS, PCA and LDA algorithms.....63

Figure 37: Average true detections with LMS, PCA and LDA algorithms.....64

Figure 38: LDA data projection- original data scatter.....67

Figure 39: LDA data projection- original data and projection vector.....68

Figure 40: LDA data projection- data after projecting onto the projection vector.....68

Figure 41: Skin (green) and non-skin (red) classes: 1) Before LDA projection; 2) After LDA projection.....69

Abstract

AN APPROACH TO DETECT HUMAN FACIAL SKIN IN THERMAL IMAGERY

Balvinder Kaur, MS

George Mason University, 2012

Thesis Director: Dr. Jill K. Nelson

Image processing tools to detect human skin in visible band imagery have been well explored by many organizations, and approaches have been developed for many security and military applications. Visible cameras are limited to human skin detection during daylight or artificial illumination conditions, but the challenge of human skin detection during nighttime remains an ongoing research effort. The most challenging problems are to understand skin texture and to develop mathematical tools for discriminating skin texture from non-skin textures in images collected using a single thermal band. To solve this problem, a set of image processing algorithms have been designed and developed for generating the skin-texture feature set discriminating feature selection, and classification. First, Gray Level Co-occurrence Matrix (GLCM)-driven skin-texture features are generated based on the skin portions of the imagery. Principal Component Analysis (PCA) is then performed on the feature set to isolate the skin discriminating features.

Then, PCA-reported skin discriminating features are employed to construct a fused image. The purpose of this fused image is to represent the skin pixels in terms of the skin-discriminating features and use this image for skin discrimination. In the last process, this fused image is used for skin and non-skin classification at the local level. For classification, three image processing approaches are adopted: 1) Adaptive optimized threshold with Least Mean Square algorithm, 2) Principal Component Analysis (PCA), and 3) Linear Discriminant Analysis (LDA). Results from all three classification techniques are analyzed for accuracy confidence levels. This research provides a generalized approach for human skin detection in thermal images, providing a non-contact, remote, and passive method for human skin detection in day or night imagery for security and military applications.

1 Introduction and Background

Human skin is a biological organ that provides an observer the ability to understand and evaluate the instantaneous state of mind of an individual within the constraints of a scenario [8] [11] [32]. For many applications related to positive identification, a human face may be observed over a period of time during which facial imagery features are detected, tracked, and identified. For many security applications, human facial skin is observed closely by the analyst through remote visible (VIS) and Infrared (IR) sensors [20] [24]. To understand the stress level of an individual, statistical calculations are performed on human facial skin regions of the thermal data [13] [24]. An algorithm to discriminate human facial skin will aid in improving these statistical calculations and thereby provide a more accurate analysis. Further, a skin detection algorithm in a single IR band will allow detecting skin during day or night conditions.

Image processing tools to detect human skin in imagery within the visible range of the electromagnetic spectrum have been well explored by many organizations, and many approaches and applications have been developed [18] [27]. Based on the characteristics of the human skin, many researchers of human skin detection methods are focusing on the measure of skin reflectivity and skin color proportions in multiple bands [29] [34]. Many factors can potentially affect identification such as usage constraints and the state

of field tests during skin detection. The results of skin detection are affected significantly by the variety of illumination sources and diverse weather conditions, if observed in the visible range [29]. Although there are challenges related to heat signatures in the thermal imagery, weather and illumination sources do not affect the thermal imagery [20] [30]. Moreover, thermal sensors are widely used for security and military application, so we can take advantage of existing technology and add to their present functionality by modifying sensor parameters.

Under this research we wish to design a set of human skin detection algorithms for IR sensor systems, capable of performing well under various weather conditions such as rain, fog, clouds, etc. Since an IR sensor system is capable of operating during day and night conditions, use of this type of sensor system with skin detection capability will significantly add to existing technology. If a human is observed as a target in a high resolution scene, a human skin detection capability will add additional features of the human as a target. For medical applications, human skin detection would assist medics to conduct stand-off remote triage [8] [17] [32]. For high resolution imagery, skin detection can be used for human auto-tracking [27] [28]. Another potential application is dimensionality reduction by focusing on a Region of Interest (ROI- human/skin) within the scene. Focusing on the ROI will contribute to reducing the storage space requirements and increasing data processing speed [23][28] [32].

1.1 Relevant Studies

Recent research has focused on methods for skin detection in visible imagery, primarily for human detection in a scene. Many algorithms have been developed for human facial skin detection in visible images that take advantage of multiple bands (VIS/IR) [18] [27]. One research group has developed skin color detection algorithms in the visible range based on modified Red-Green-Blue (RGB), Cyan-Yellow-Magenta (CYM) and Hue Saturation Intensity/Value (HSI/HSV) techniques [23].

Some work has been done in the areas developing models for understanding human skin characteristics in multi-spectral and hyperspectral images. These models are based on human skin reflectance and multi-color information obtained from multiple wavebands [22] [29]. Along with the model development to understand human skin characteristics, Jing and his team have developed algorithms for skin detection in near infrared (NIR) and visible (VIS) bands based on skin color estimation [18].

A group of researchers have studied a pixel-based skin color detection technique for color images and have developed an algorithm for selecting the best color space. This group has found that smaller selection of color space can reduce the standard deviation of the skin and non-skin classes and therefore generates compact classes. For associating color image pixels with skin and non-skin classes, they have employed a Bayesian classifier [27].

For post-processing of human detection readings, some work has been performed employing hyperspectral sensors. Airborne Real-time Cueing Hyperspectral Enhanced Reconnaissance (ARCHER) is an example of a hyperspectral system used for collecting images during day light under minimal cloud cover conditions. It uses a set of skin detection algorithms based on the insight gained from the modeled skin reflectance and utilizing the knowledge of the normalized difference of vegetation index (used for vegetation segmentation in hyperspectral images) [29] [37].

For almost all the applications during processing or post-processing, the feature classification process is a key component of discriminating targets from anomalies. A group has developed an automated imaging system for the discrimination between normal and abnormal tissues in medical images obtained by chest radiography scans. They have utilized PCA, LDA, and non-linear Discriminant Analysis (NDA) for features reduction methods. In addition to standard classification algorithms, they have also explored the use of an Artificial Neural Network (ANN) classifier for classification [1].

A study conducted in computing fingerprint features using texture based co-occurrence matrices was successful. The ANN classifier was employed for separating and associating the fingerprints with specified classes [41].

Through this interesting research for human skin modeling, detection, classification and recent threats to national security by individual human actors, we have recognized a gap that we must fill under this research, namely detecting human facial skin in a single thermal band.

1.2 Motivation

The image processing tools to detect human skin in imagery based on the visible range of the electromagnetic spectrum have been well explored and approaches have been developed for many applications. However, visible cameras are limited to human skin detection during daylight or under artificial illumination and the challenge of human skin detection during nighttime remains an ongoing research effort. Tools for human skin detection in thermal imagery add to the capability to detect human skin under both day and night conditions [20] [21] .

An additional motivation for skin detection is derived from recent research involving human signature understanding, specifically human facial pore detection and evaluation of the pore count signal to analyze stress level [20] [24]. Human skin pores are related to the Autonomic Nervous System (ANS) response system. Although not fully understood at a molecular level, Electrodermal Activity (EDA) or Skin Conductance (SC) has often been used in psychophysical experiments to monitor ANS response [20] [24]. Often, EDA is measured with electrodes attached to the tips of fingers, which is cumbersome and not oriented towards use for remote sensing or in a field environment [24]. Recent work with high resolution thermal imagers has shown that they provide an emerging capability for remotely and passively assessing the ANS response [24] [30][33]. This is done by assessing the activation of eccrine skin pores on the face in real time (see Figure 1) [21] [24].

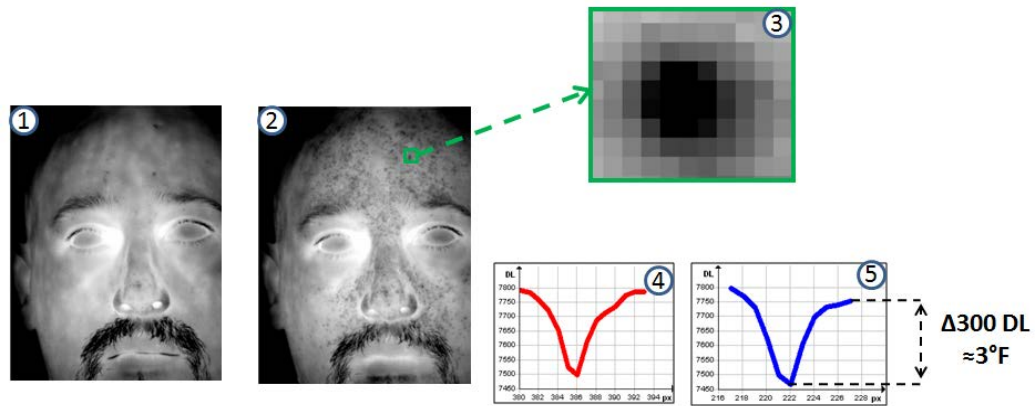


Figure 1: Facial skin pore characteristics: 1) Subject with no response; 2) Subject with high response; 3) Digital zoom of active pore or sweat gland ; 4) X-axis thermal cross section ; 5) Y-axis thermal cross section [20]

Detected eccrine sweat gland activation in thermal imagery has been shown to relate to EDA measurements from contact electrodes. By using passive thermal imagery at a standoff distance, this potentially provides a non-contact alternative to the widely used method of assessing ANS response with contact electrodes. Figure 1 shows an example of a facial image of a person in a low and high stress situation. Under stress conditions, activated pores expand and become larger. It can be seen by comparing the two conditions that a significant number of pores are activated when the subject becomes stressed, and these can clearly be seen in the thermal image [21] [24].

A real-time system was developed to provide pore counts for thermal imagery; however, there were significant image processing challenges. The existing algorithm used a static mask around the central portion of the face and relied on tracking to ensure the mask stayed in place. As a subject's head turned, this mask included areas of the face, such as hair, where the pore detection algorithms provided false counts. A study was conducted

to overcome these issues by designing new techniques to detect the facial skin, generate a skin-only mask, and update this skin-only mask from frame to frame [20]. The algorithm and the false pore mitigation results were published. Results demonstrated an improvement in the final pore count results by reducing false alarms by 48% [20]. For skin detection, the Gray Level Co-occurrence Matrix (GLCM) texture kernel was used for skin mask detection and all of possible texture features were considered. The skin detection algorithm produced favorable results, but the theory behind the concept was not understood completely. A lack of understanding of the theory behind skin detection algorithm motivated the work presented in this thesis. Now, the question was “What texture properties characterize skin and how can this be proven?”

1.3 Proposed Methodology and Solution

To study the texture characteristics, initially we looked at the local skin and non-skin portions within Mid-wave-thermal (MWIR) imagery and generated the Gray Level Co-occurrence Matrix (GLCM). Figure 2 illustrates a visual difference in skin and non-skin textures. The skin GLCM shows a gray level spread throughout the whole window whereas the non-skin GLCM shows a diagonal spread of gray levels. Further, because a human can visually tell the difference between skin and non-skin texture using this tool, we concluded that texture could be used to discriminate skin from non-skin portions in an image.

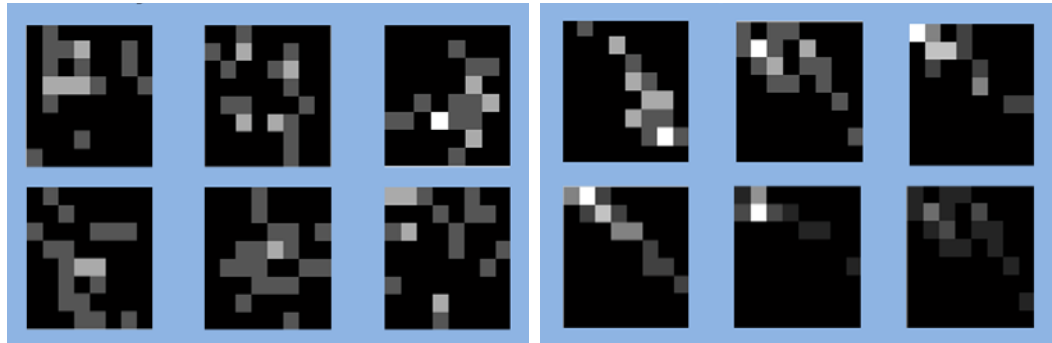


Figure 2: Facial non-skin and skin texture characteristics (Gray Level Co-occurrence Matrix) in 11x11 window: Non-skin locations (left) and Skin locations (right)

The most challenging problems were to understand skin texture and to develop a set of mathematical tools for discriminating skin textures from non-skin textures in a single thermal band. To solve this problem, a set of image processing algorithms were designed and developed for generating the skin-texture feature set for discriminating feature selection and classification. First, GLCM-driven skin-texture features were generated based on the skin portions of the imagery [16]. Principal Component Analysis was then performed on the feature set to isolate the skin discriminating features [36]. Then, PCA-reported skin discriminating features were employed to construct a filtered image. In the last process, this filtered image is used for skin and non-skin classification at the local level (within the GLCM kernel size). For classification, three image processing approaches were adopted: 1) Adaptive optimized threshold with Least Mean Square algorithm, 2) PCA, and 3) Linear Discriminant Analysis (LDA) [10] [25] [36] [40]. Results from all three classification algorithms were analyzed for validating the classification algorithms.

1.4 Proposed Candidate Algorithms

As discussed in the previous section, proposed candidate algorithms to solve the problem of human skin detection are as listed below.

1. Feature generation based on the Gray Level Co-occurrence Matrix
2. Feature reduction using PCA
3. Classification for skin segmentation with
 - a. Adaptive LMS optimized algorithm
 - b. PCA algorithm
 - c. LDA algorithm employing Fisher's criteria
4. Post-processing for cleaning the resulting image using morphological operations

These algorithms will be discussed in more detail in next few chapters.

2 Algorithms for Skin Detection

We applied a set of algorithms to a thermal image to separate human facial skin from the non-skin areas. This set of skin detection algorithms has three major parts 1) pre-processing the data to prepare for processing, 2) processing the data for feature selection and then classification for skin detection, and 3) post-processing to remove single hanging pixels from the generated skin-only images. Figure 3 illustrates a top level flow chart for the skin detection algorithm.

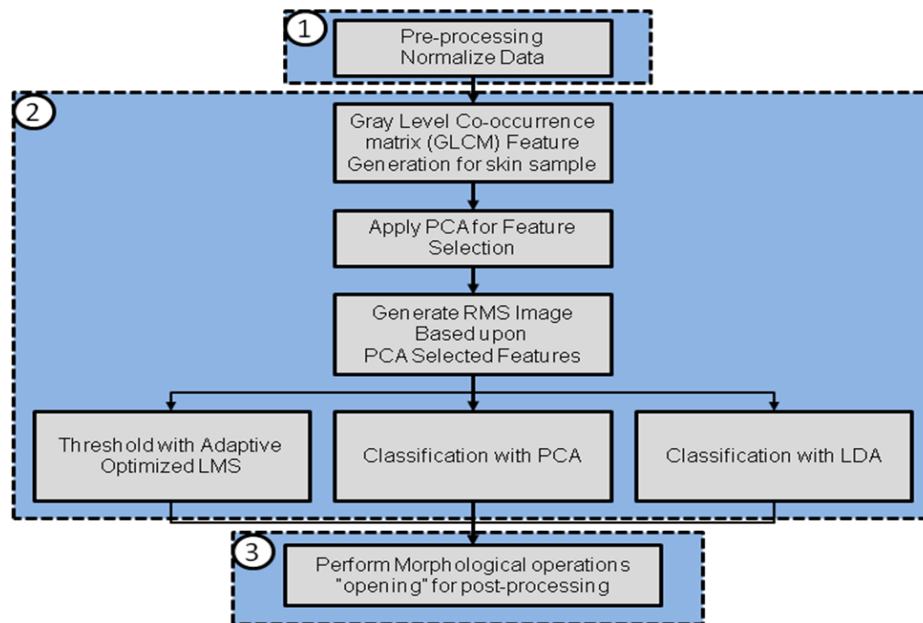


Figure 3: Main algorithm flow chart

This chapter covers the pre-processing section of the algorithm whereas the processing and post-processing parts of the algorithm covered in the remaining chapters.

2.1 Pre-processing

Data preparations include correction for lighting effects, non-uniformity, gray-level adjustment, dead-pixels, etc [15]. For our case, a set of operations performed for preprocessing include 1) data normalization to bring the gray level of the imagery to a common ground ($0 \leq \text{range} \leq 1$), and 2) data smoothing to reduce the dynamic range of the facial skin.

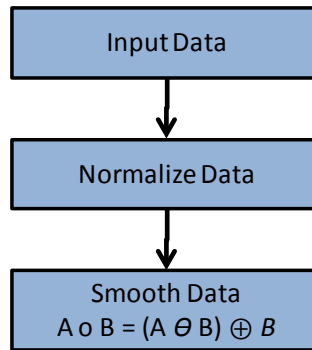


Figure 4: Pre-processing algorithm flow chart

2.1.1 Normalize Data

Under the normalization process, gray levels of all the images are brought to a common baseline. This process assures a valid comparison between two data sets that are different due to illumination conditions or dynamic range variations. During the normalization

process two steps are followed. In step one, we compute an initial minimum intensity value of the image (I_{min}) and use this value to compute each pixel's new intensity value following mathematics in Equation 1. This step baselines the minimum value of the image to zero and therefore defines the minimum range of the data. In step two (Equation 2), we divide the new image generated following Equation 1 by the maximum value of the new image itself. This process defines the maximum range of the data to value one. This process is applied to every frame of the video sequence before computing statistics on each frame.

$$\begin{aligned} \text{Let } I_{min} &= \text{Minimum intensity value of Image } I \\ I_{new} &= I_{ij} - I_{min} \end{aligned} \quad \text{(Eq. 1)}$$

$$\begin{aligned} \text{Let } I_{max} &= \text{Maximum intensity value of Image } I_{new} \\ I_{norm,ij} &= \frac{I_{new,ij}}{I_{max}} \end{aligned} \quad \text{(Eq. 2)}$$

Where: I = original image

I_{norm} : Normalized image

2.1.2 Smooth Data

To automatically extract a sample skin-chip from the scene, a data smoothing operation is performed utilizing a morphological operation called opening. An opening is defined as erosion followed by dilation. The dilation process adds pixels to the boundaries of objects

in an image whereas the erosion process removes pixels from the boundaries of objects in an image [9] [15] [16].

In general, morphological operations apply a structuring element to an input image and produce an output image of the original image size or smaller (based on the operation). After performing morphological operation, the value of each pixel in the resultant image is based on computed values of the original image within the structuring element window. A morphological operation is sensitive to the size of the structuring element since the output results are based on the overlapping of structuring element with the valid target pixels. [15].

2.1.2.1 Dilation

Assume that A is an original object onto which we operate with a structuring element B. In theory, dilation on A is performed employing the structuring element B and is denoted by $A \oplus B$ (Equation 3). Equation 3 shows that a set of pixels represented by z is the output of the dilation function. Moreover, resultant z pixels are the common pixels among original set A and calculated set $(\hat{B})_z$ [9] [15].

$$\begin{aligned}
 A \oplus B &= \{z \mid (\hat{B})_z \cap A \neq null\} \\
 \hat{B} &= \{w \mid w = -b, \text{ for } b \in B\} \\
 (\hat{B})_z &= \{c \mid c = w + z, \text{ for } w \in B\}
 \end{aligned}
 \tag{Eq. 3}$$

where \hat{B} : Reflection of a set B

$(\hat{B})_z$: Reflection of a set B translated by point z

Figure 5 visually demonstrates the dilation operation more clearly. Here, the structuring element B of size $d/4$ is applied to the object A. After performing the dilation operation, size of the dilated object A increases by $d/4$ in both horizontal and vertical directions [9] [15].

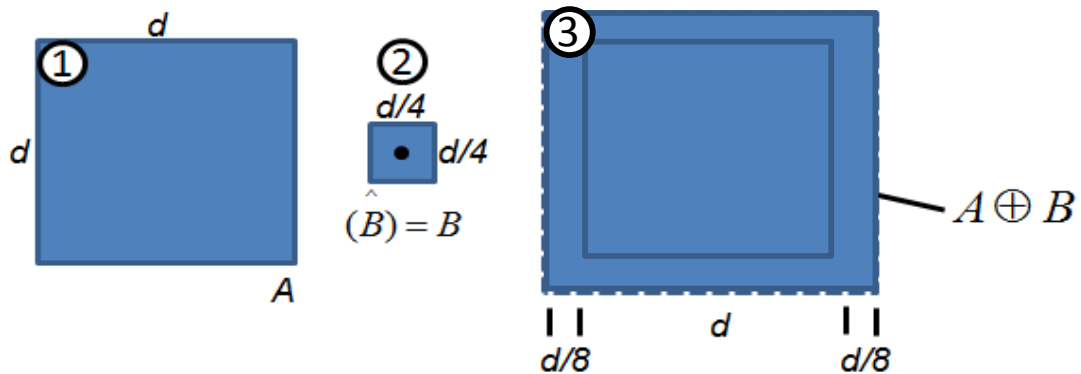


Figure 5: Dilation process: 1) Element A; 2) Square structuring element B; 3) Dilation of A by B, shown shaded [15]

2.1.2.2 Erosion

Erosion on A is performed employing the structuring element B and is denoted by $A \ominus B$ (Equation 4). A simplified version of Equation 5 is expressed as Equation 6 [9] [15].

$$A \ominus B = \{z \mid (B)_z \subseteq A\}$$

$$(B)_z = \{c \mid c = b + z, \text{ for } b \in B\} \quad (\text{Eq. 4})$$

where, $(B)_z$: set B translated by point z

Equation 4 shows that a set of pixels represented by z is the output of the erosion function. The set z is selected such that resulted eroded output $(B)_z$ is within the set A. Figure 6 demonstrates a visual interpretation of the erosion operation. When structuring element B of size $d/4$ is applied to an object A of size d, the size of eroded object A is decreased to $3d/4$ in both horizontal and vertical directions. Erosion operation removes edges of object A by half of the size of the structuring element from each side.

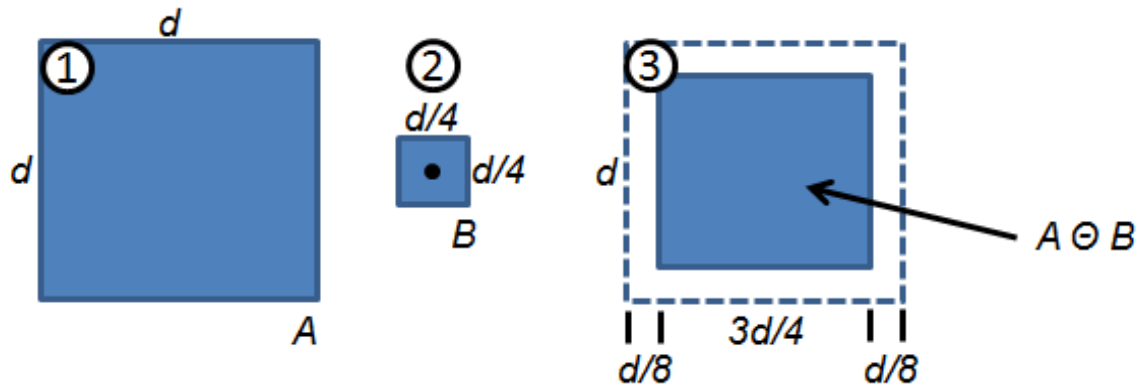


Figure 6: Erosion process: 1) Element A; 2) Square structuring element B; 3) Erosion of A by B, shown shaded [15]

2.1.2.3 Opening

An opening is defined as erosion of element A followed by dilation of the eroded element A. The structuring element size for both operations may be similar or may differ based on the expected results. Although, opening removes some of the object pixels from the object's boundaries, it preserves the object regions that have a similar shape to the kernel.

Equation 5 shows the mathematical equation for the opening operation. The equation states that the erosion of A with the structuring element B is performed first, and then the dilation operation is performed on the eroded output [9] [15].

$$A \circ B = (A \ominus B) \oplus B \quad (\text{Eq. 5})$$

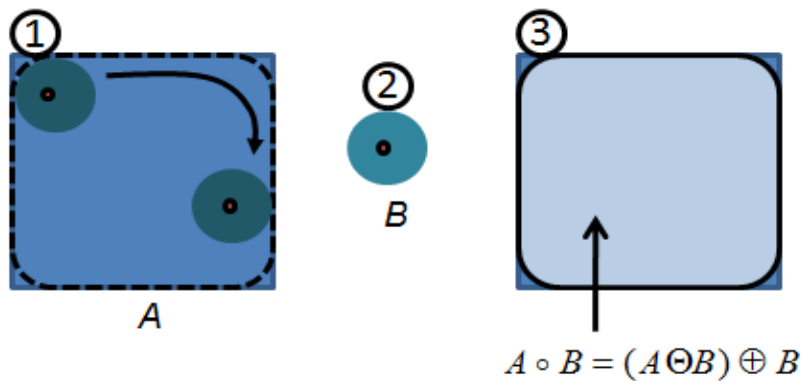


Figure 7: Opening process: 1) Element A; 2) Disk structuring element B; 3) Opening of A by B, shown shaded light blue [15]

Figure 7 shows a visual interpretation of the output object after performing the opening operation. Notice that object A after the opening operation has round edges because the

structuring element B has a disc shape. During an erosion operation, the outer boundaries of the object are trimmed by half of the structuring element size, and during the dilation operation boundaries are increased by half of the structuring element size [9][15].

3 Preparing Ground Truth (GT) Data

A Ground Truth (GT) sample skin chip is a key component of our skin detection algorithm. GT preparation includes two major steps 1) skin chip extraction and 2) GLCM feature generation for the extracted skin chip (Figure 8). This chapter discusses a method for automatic skin-chip extraction and a set of algorithms for calculating GLCM features from the GT.

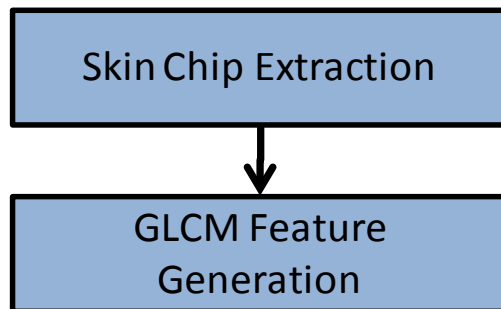


Figure 8: Flow chart to prepare Ground Truth (GT)

3.1 Automatic Skin-chip Extraction

A series of operations are performed on the original image to select a sample skin-chip automatically (Figure 9). First, the data is normalized and then a smoothing operation is

performed on the image. These operations are performed on images to smooth contours, replace low intensity areas (i.e., pores) with local neighborhood intensities, and eliminate thin lines [20]. After the smoothing operation, the image has a high contrast difference between overall skin and non-skin regions.

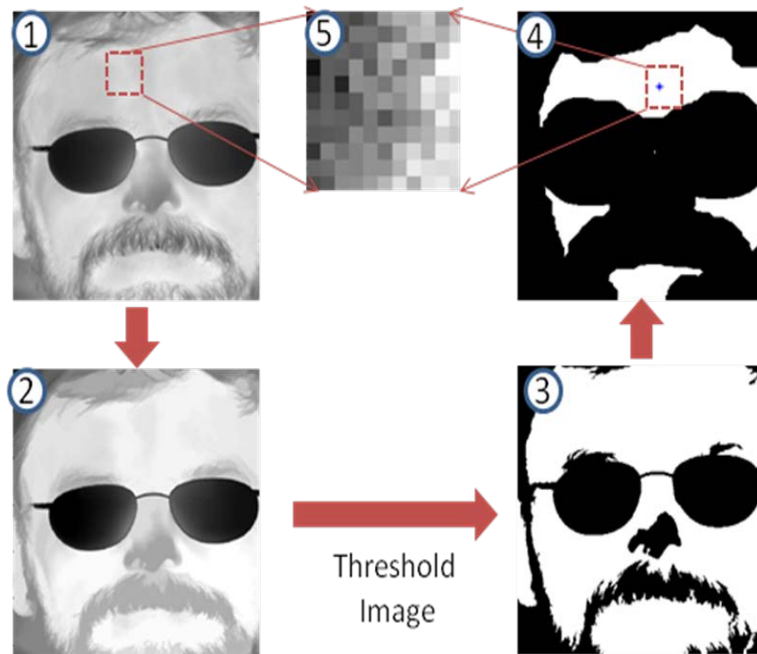


Figure 9: Auto skin detector process: 1) Original image; 2) Image after smoothing; 3) Image with skin and edges ; 4) All skin image; 5) Skin chip

An average intensity value of the smoothed image is calculated and applied as a threshold to the smoothed image itself. Output of this process generates an image with minimal non-skin regions along with skin regions. Minimal non-skin regions are close to the edges of valid skin pixels and thin hair regions. Since this process of determining the

skin is contrast-based, we can distinguish hot spots from the cold spots (high blood flow regions have high intensity values and can be called hot spots) within the image, but we cannot detect all the skin regions. Further, because we are experimenting with high resolution imagery with small backgrounds, hot spots in the imagery are from human facial skin. Therefore, we use this knowledge obtained from thermal imagery to automatically generate a skin ground truth chip.

Figure 9 , part 3 shows a sample output of the thresholded, smoothed image. As stated previously, this image has skin and non-skin portions; therefore, to derive a skin-only image, we apply the morphological operation erosion with a structuring element of size 60x60 pixels. This operation removes 30 pixels from the edges of the object and ensures the skin pixels in the resultant image. To ensure that the resultant image has skin-only pixels, this algorithm was applied to images from five subjects. Then three operators visually verified that the resultant images had skin-only pixels. For this particular example, the resultant image is shown in Figure 9 part 4 where pixels in white represent skin regions and pixels in black represent non-skin regions. We have designed an algorithm to randomly select a skin location to ensure the selected skin location is surrounded by all valid pixels within a window size of 11 x 11. Figure 9 part 4 shows a selected sample skin location (blue star). Then from the original image (Figure 9 part 1), centered at the selected skin location, we extract the skin-chip.

The next section discusses a process to generate the Gray Level Co-occurrence Matrix (GLCM) and texture-driven features computations.

3.2 GLCM Feature Generation

To process data for skin detection after the skin-chip extraction, a GLCM matrix is generated, and a set of texture-driven features are calculated. GLCM is an image-texture-driven matrix that is widely used for many applications to perform texture based classification analysis and to estimate image properties related to second-order statistics [15] [16] [31]. The GLCM kernel is derived from sample texture chip by following N-to-1 mapping process for mapping intensity pairs. Each entry (i, j) in the GLCM corresponds to the number of occurrences of the pair of gray levels i and j that are a distance d apart in the original chip [15]. The GLCM matrix, denoted by $C_{\Delta x, \Delta y}$ is defined mathematically in Equation 6. Visually, Figure 10 represents a process of constructing a GLCM matrix. For example, an entry for pair (1, 1) is made as 1 at the (1, 1) location in the predefined GLCM template because there was only one (1, 1) pair in the original image. However, there are two pairs for (1, 2), so an entry of 2 is made in the (1, 2) location in the predefined GLCM template. Following the horizontal pattern counting procedure, the GLCM matrix of the skin sample is constructed as follows.

$$C_{\Delta x, \Delta y}(i, j) = \sum_{p=1}^n \sum_{q=1}^m \begin{cases} 1, & \text{if } I(p, q) = i \text{ and } I(p + \Delta x, q + \Delta y) = j \\ 0, & \text{otherwise} \end{cases} \quad \text{(Eq. 6)}$$

Where: I = Skin Chip

$m = n = 1, 2, \dots$, # of rows/columns of I

i and j are used for data binning

$$\Delta x \equiv 1; \quad \Delta y \equiv 0$$

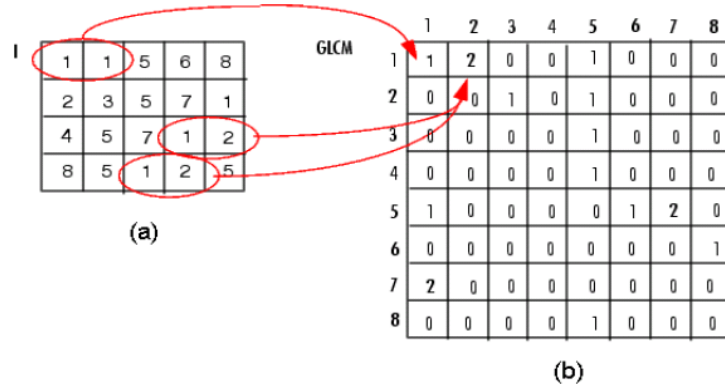


Figure 10: GLCM kernel mapping
(Extracted from [15])

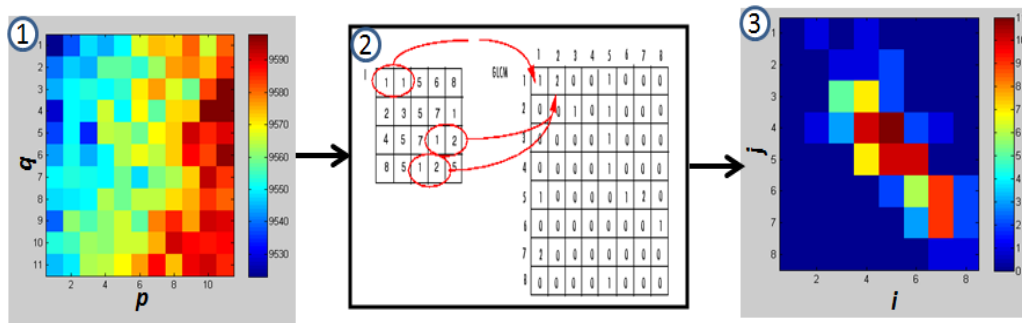


Figure 11: GLCM kernel: 1) Sample original chip; 2) Transformation process (N to 1 mapping) [15]; 3) GLCM kernel

The GLCM matrix is square and the rows and columns of the matrix represent the image data range. The test image in Figure 10 has gray levels up to 8, but the employed thermal data for our experiments has 256 possible gray levels. We can construct a 256 x 256

square GLCM matrix, but data processing at each pixel will be computationally very intense. Therefore, data binning in a set of 8 bits is adapted to generate a GLCM kernel of 8 x 8 pixels. Figure 11 illustrates a sample GLCM kernel derived from original data. Haralick suggested that texture-driven features can be calculated on the GLCM kernel for further processing [16].

Haralick has proposed fourteen statistical features that can be derived from GLCM. However our analysis only explores use for skin discrimination [20]. Equation 7 normalizes the GLCM matrix and this normalized GLCM is used to derive texture-driven features listed, as described in this section.

$$P_{i,j} = \frac{C_{i,j}}{\sum_{i,j=0}^{N-1} C_{i,j}} \quad (\text{Eq. 7})$$

Where: i : the row number; j : the column number

$C_{i,j}$: the value in the pixel (i, j) of the GLCM matrix

$P_{i,j}$: the normalized pixel value at (i, j)

N : the number of rows or columns.

3.2.1 Contrast

The contrast is a difference between the neighborhood pixels of a normalized GLCM $P(i,j)$ and measures the amount of local variation in an image. The contrast is a spatial dependent metric because the gray level difference is measured between the neighboring pixels. This difference is also referred to as a difference between the moments [16]. As per Equation 8, contrast is a sum of squared variances at each pixel of the GLCM.

$$\sum_{i,j=0}^{N-1} P_{i,j} (i - j)^2 \quad (\text{Eq. 8})$$

Here the contrast value is zero if i and j are equal, but contrast difference is introduced if i and j are not equal. The goal is to add all those differences to report a contrast of the GLCM matrix. A high value of contrast is achieved when the GLCM pixels have wide range of intensity variations.

3.2.2 Entropy

Entropy measures the disorder of intensity variations within an image and is independent of the spatial information. A difference between the contrast and entropy is the spatial dependency. As illustrated previously, a contrast is a spatially dependent metric and difference between the pixels is added to generate a metric whereas the entropy is independent of the spatial information. When the intensity variation between the image pixels is small or negligible, entropy of the image is large. Also, when the image has

non-uniform texture, there are not that many repeating intensity pairs, which results in small values for the GLCM elements. Further, small element values over the whole image results in a very large entropy [16]. A mathematical Equation 9 is used to compute entropy of an image.

$$\sum_{i,j=0}^{N-1} P_{i,j} (-\ln(P_{i,j})) \quad (\text{Eq. 9})$$

3.2.3 Homogeneity

Homogeneity of the image is measured by the inverse difference between moments (IDM) where IDM is inversely proportional to GLCM contrast. Equation 10 returns a value that represents the closeness of the GLCM non-zero pixels with respect to the GLCM diagonal. The range of this parameter is between 0 and 1, and it achieves the largest value when non-zero pixels are centered near the main diagonal [16].

$$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1 + P_{i,j}(i - j)^2} \quad (\text{Eq. 10})$$

3.2.4 Correlation

The correlation operation performed on the GLCM measures the relationship of gray levels in the neighborhood pixels. Equation 11 returns a measure of how correlated a pixel is to its neighbors over the whole image. The range of this parameter is between 1

and -1; any value less than 0 indicates negative correlation and greater than 0 indicates positive correlation [2] [16] .

$$\sum_{i,j=0}^{N-1} P_{i,j} \left[\frac{(i - \mu_x)(j - \mu_y)}{\sqrt{\sigma_x^2 \sigma_y^2}} \right] \quad (\text{Eq. 11})$$

where the parameters $\mu_x, \mu_y, \sigma_x, \sigma_y$ are calculated employing the equations 12 and 13.

$$\begin{aligned} P_x(i) &= \sum_{j=0}^{N-1} P_{i,j}; & P_y(j) &= \sum_{i=0}^{N-1} P_{i,j} \\ \mu_x &= \sum_{i=0}^{N-1} i \sum_{j=0}^{N-1} P_{i,j} = \sum_{i=0}^{N-1} iP_x(i) \\ \mu_y &= \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} jP_{i,j} = \sum_{j=0}^{N-1} jP_y(j) \end{aligned} \quad (\text{Eq. 12})$$

$$\begin{aligned} \sigma_x^2 &= \sum_{i=0}^{N-1} (i - \mu_x)^2 \sum_{j=0}^{N-1} P_{i,j} = \sum_{i=0}^{N-1} (P_x(i) - \mu_x(i))^2 \\ \sigma_y^2 &= \sum_{j=0}^{N-1} (j - \mu_y)^2 \sum_{i=0}^{N-1} P_{i,j} = \sum_{j=0}^{N-1} (P_y(j) - \mu_y(j))^2 \end{aligned} \quad (\text{Eq. 13})$$

3.2.5 Mean

The mean operation computes an average gray value within a window size $k \times k$ and replaces the center pixel value with the computed gray value. Employing Equation 14,

this operation is repeated across the whole image and an average GLCM image is computed.

$$Mean(i, j) = \frac{1}{(k-1)^2} \sum_{i,j=0}^{k-1} P(i, j) \quad \text{(Eq. 14)}$$

Although, features calculated in this chapter are derived from the skin-chip, not all the features represent variability in the skin data equally. We are interested using a limited set of features; therefore, in the next chapter, we explore a feature reduction process employing PCA.

4 Applying PCA for Data Reduction

Principal Component Analysis (PCA) represents a signal that lies in the direction of the principal component. PCA has been called one of the most valuable results from the applied linear algebra. PCA computes a set of orthogonal components (vectors) passing through the data. Figure 12 shows a sample data where the σ_{signal}^2 component is in the direction of the maximum data variability and therefore is called the principal component. However the σ_{noise}^2 component is an orthogonal component to the σ_{signal}^2 component and represents the direction with less variability. Thus, PCA provides a method for reducing the data's dimensionality and representing the data in terms of the data variability [19] [25] [36].

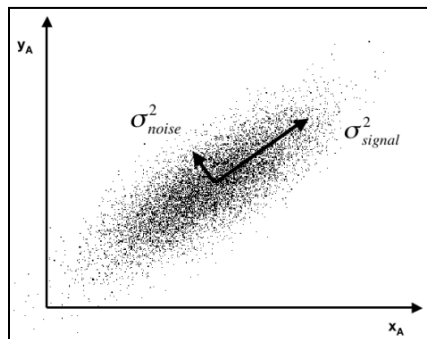


Figure 12: Sample data representing signal and noise variability
(Extracted from [36])

4.1 Feature Reduction

We apply PCA analysis for understanding the most contributing texture features for skin characterization. We normalize the extracted skin-chip to have a common mean of 0 and a standard deviation of 1 following Equation 15.

$$\bar{x}_{ij} = \frac{x_{ij} - \mu}{\sigma} \quad (\text{Eq. 15})$$

where: μ is the mean value of the original image x

σ is the standard deviation of image

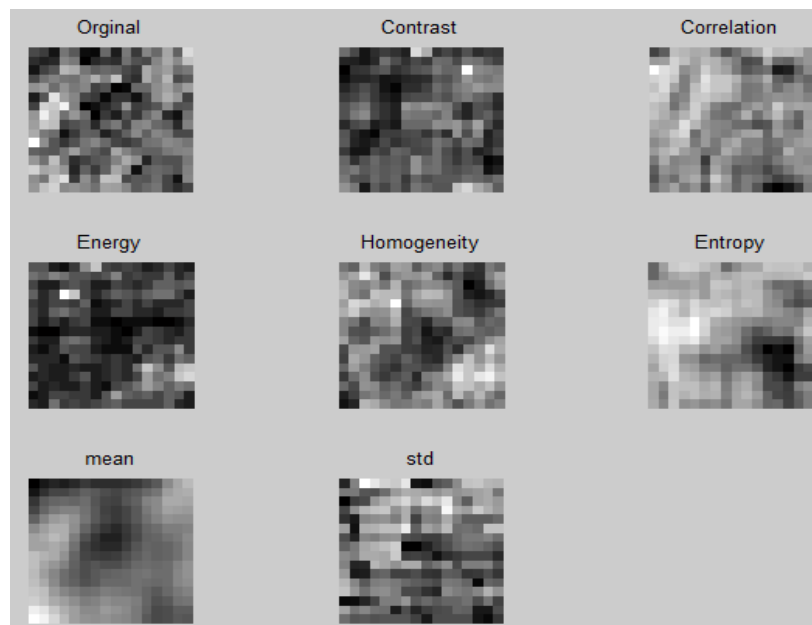


Figure 13: For selected skin portion (top left, original), Gray Level Co-occurrence Matrix (GLCM) driven texture feature images

After normalizing the skin-chip, GLCM-driven feature images are constructed. Figure 13 demonstrates calculated feature images.

These 2-dimensional feature images are arranged in a set of 1-dimensional vectors and a matrix is created for PCA analysis.

$$X = \{FeatureImg_1, FeatureImg_2, \dots, FeatureImg_M\} \quad (\text{Eq. 16})$$

where: M is the number of feature images

Each feature image has N pixels

Each feature image is $N \times 1$ in the matrix 'X'

The goal is to reduce the number of features from M to K where $K < M$. K represents the most contributing eigenvectors, and eigenimages are reconstructed using K eigenvectors. The algorithm for feature reduction from M images to K images involves the steps listed below.

Step 1: Calculate a mean image of matrix X. Figure 14 demonstrates a sample mean image.

$$\bar{x} = \frac{1}{M} \sum_{i=1}^M X_i \quad (\text{Eq. 17})$$

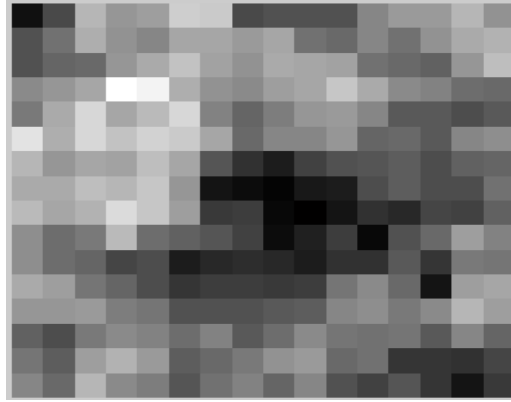


Figure 14: Mean image

Step 2: Subtract the mean image from each feature image to adjust the data. For a valid comparison between the two or more data sets, we prepare the data with a common baseline of zero-mean value [36].

$$\Phi_i = x_i - \bar{x} \quad (\text{Eq. 18})$$

Step 3: Form the matrix A

$$A = [\Phi_1 \Phi_2 \dots \Phi_M]; \quad (N \times M \text{ matrix}) \quad (\text{Eq. 19})$$

Step 4: We compute the covariance matrix C following Equation 20. The covariance can be used to describe relationships between all pairs of measurements in a data set. Diagonal values are the variances between the data pairs, and the top and bottom triangles in the covariance matrix represent correlation between the set of measurements.

$$C = \frac{1}{M} \sum_{i=1}^M \Phi_i \Phi_i^T = AA^T; \quad (N \times N \text{ matrix,}) \quad (\text{Eq. 20})$$

where $i = 1: M$ features of the PCA input matrix X .

Step 5: Then, we compute the eigenvalues and associated eigenvectors of the covariance matrix C [25] [36]. The PCA-generated eigenvectors are orthogonal to each other and show the variability in the direction of the vectors. After calculating the eigenvectors and eigenvalues, we sort the eigenvalues and associated eigenvectors from highest to lowest. The data variability for the arranged eigenvectors varies from maximum to minimum. Step 6 demonstrates a procedure to compute a reduced set (K) of eigenvalues and associated eigenvectors for further processing.

Step 6: The K is a number that specifies the dimensionality of the subspace of M that preserves the variability of the skin features. In other words, the K eigenvectors are selected such that they demonstrate the maximum variability of the data. Equation 28 demonstrates the constraint used for defining K . The expected threshold can be set to any number between zero and one. For example, a threshold of 0.9 is selected and compared with the ratio specified in Equation 21 for a possible K value. The first K number (from 1 to N) that satisfies the condition in Equation 21 is selected as the K value.

$$\text{Threshold} < \frac{\sum_{i=1}^K \lambda_i}{\sum_{i=1}^M \lambda_i} \quad (\text{Eq. 21})$$

Figure 15 demonstrates the eigenvalues rank plot for the sample skin-chip extracted in the third chapter. As clearly shown in Figure 15, first 4 eigenvalues are the most contributing and are noticeably higher than the rest of the eigenvalues. Based on the expected threshold value of 0.90, K was determined to be 4 eigenvectors/eigenvalues.

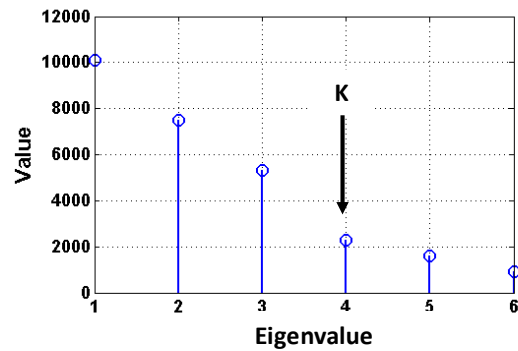


Figure 15: Eigenvalue distribution

Figure 16 demonstrates a sample image data with three features and the top four eigenvectors passing through the data.

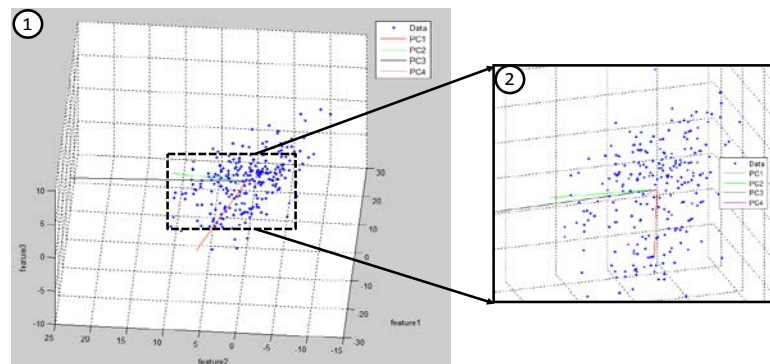


Figure 16: Eigenvectors plotted on top of original data for 3 features: 1) Overview of data and 2) Zoomed in three orthogonal components

Step 7: Using the K eigenvectors, M eigenimages are generated. For the previously selected skin-chip sample, Figure 17 shows a partial sample set of eigenimages that were created employing eigenvectors associated with top 4 eigenvalues.

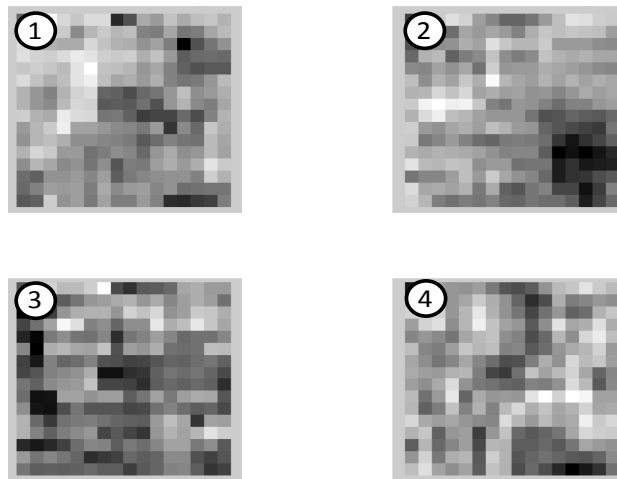


Figure 17: Eigenimages corresponding to the K eigenvectors - 1) Eigenimage#1; 2) Eigenimage#2; 3) Eigenimage#3; 4) Eigenimage#4

Step 8: For many data dimensionality reduction applications, step 7 is the last step. However, for our case, the goal is to understand which features contribute the most in referencing features to represent the skin data. Therefore after obtaining the skin contributing feature sets in step 7, each eigenimage is compared with each feature image and the correlation value is computed. A high correlation value between an eigenimage and a feature image represents an association. Output of this step reports the name of the K features associated with the selected K eigenimages. In other words, output of this step is the number and the type of feature images that should be computed for skin detection.

The process to construct a feature image is to start with a filter window size of the original skin-chip. As demonstrated in Figure 18, the sample image has M rows and N columns. We move the filter window from the left side to the right side of the image starting from location one, in steps of a pixel, by following the arrows. These arrows are only one pixel apart in the horizontal and vertical directions. The purpose of this process is to move the filter window on top of each pixel of the original image and calculate the GLCM kernel within that window. From the pixel's calculated GLCM kernel, calculate the K set of features recognized in step 8 by following equations in the third chapter. At each pixel, we calculate the features values within the filter window. To construct feature images, we create K , $M \times N$ blank images that we use to place the features values and construct feature images. For each pixel location, we place the calculated feature value in the blank feature images. At the end of the process, we have moved our filter window from point 1 to point 4 by moving at a step of one pixel at a time in horizontal and vertical directions.

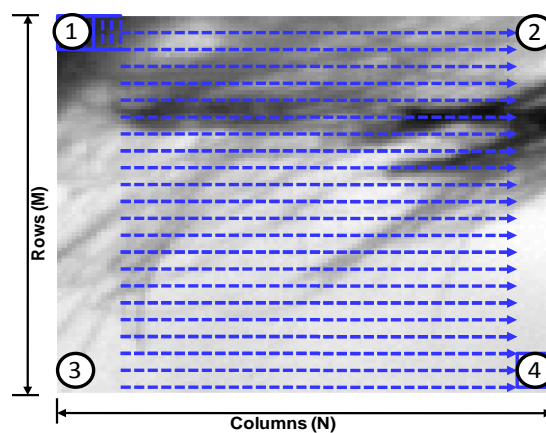


Figure 18: Process to construct a feature image

Following the feature image generation process, K feature images are constructed as shown in Figure 19. For this particular data, we have identified that features 1, 3, 6 and 7 (contrast, energy, mean, and standard deviation, respectively) are the most contributing features for skin detection. Therefore for further analysis, we need to compute only these four features for the test images for the skin detection.

In conclusion, the PCA allows us to recognize a set of features that characterizes the skin. As a result, only a limited set of features are computed for the sequence of video frames for discriminating skin from non-skin regions. For this particular data set, four skin discriminating features were recognized. Figure 19 shows four feature images computed over a small region of face with hair and skin portion.

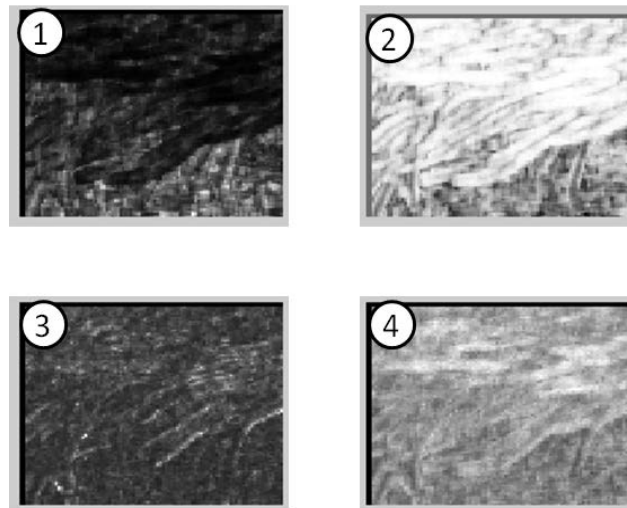


Figure 19: PCA selected GLCM feature images- 1) Contrast; 2) Correlation; 3) Energy; 4) Homogeneity

4.2 Fused Image Generation

After performing PCA on a set of features and obtaining a reduced set of skin discriminating features, the set of PCA-reported features are calculated for a test image as explained in the previous section (see Figure 19).

We had two options for discriminating skin from non-skin regions- 1) operate on each individual feature image or 2) operate on a fused feature image. For the first option, we apply three classification algorithms one at a time on each feature image. After computing K classified images with skin and non-skin regions, we merge results from each classifier and construct an image with skin and non-skin regions. Since we have three classification algorithms, this operation is repeated three times for generating skin and non-skin regions. However, for the second option, we combine feature images to construct a fused image and then apply classification algorithms on the fused image. After running both options, we have learned that the second option is computationally less intense whereas results are similar. Under the scope of this thesis, we have adopted the second option and have explored the skin discrimination algorithms listed in the fifth chapter.

Many options are available to combine multiple images to construct a fused image [15]. Since we are interested in pixel intensity, we choose to weight all features equally and compute a root-mean-square (RMS) intensity value at each pixel. Equation 22 and Figure 20 demonstrate a procedure for constructing a fused image.

$$F_Im(i, j) = \sqrt{(PCA_{img,1}(i, j))^2 + (PCA_{img,2}(i, j))^2 + \dots + (PCA_{img,K}(i, j))^2} \quad (\text{Eq. 22})$$

Where: $PCA_{img,K}$: PCA reported feature images

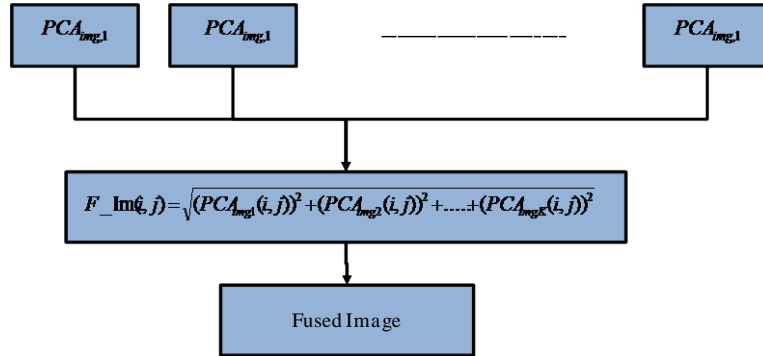


Figure 20: Flow chart to generate a fused image

Figure 21, part 1 demonstrates an example of a portion of a test image. First, PCA-reported features are calculated for this entire portion of the test image and then employing the image fusion process explained in this section, a fused image is generated (Figure 21, part 2).

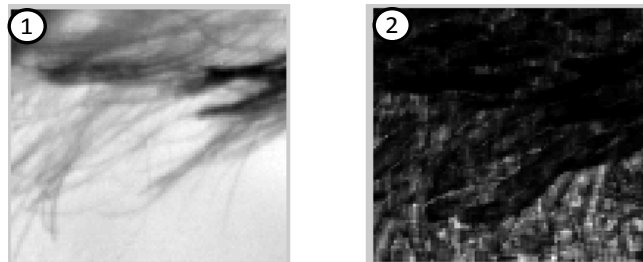


Figure 21: Fused image results: 1) Original image and 2) Fused image

5 Classification for Skin Detection

Now that we have derived a fused image where each pixel's gray level is defined by a combination of PCA selected features, the next task is to classify these pixels based on their gray levels and fused texture characteristics. Since we have prior knowledge of features that were employed to construct a gray level of each pixel, we also have knowledge of the skin (ground truth) in terms of texture and the gray levels.

We utilize this information for differentiating the skin from non-skin portions following three classification algorithms: 1) adaptive optimized LMS, 2) PCA, and 3) LDA. The subsequent sections will explain each classification algorithm in more detail.

5.1 Classification with Adaptive Optimized Least-Mean-Square (LMS)

This classification algorithm is based on thresholding. Thresholding is a very challenging operation because of the presence of noise in the data. Moreover, this is an application-specific operation and many studies have been performed to find an optimal threshold algorithm [40]. Since we understand the properties of the actual skin (D_k), and a fused image (X_k), we can design an adaptive optimized threshold algorithm based on the Least-Mean-Square (LMS) estimation of the error. Figure 22 shows a flow diagram

of the algorithm that we have designed for segmenting/classifying skin from non-skin portions within an image [40].

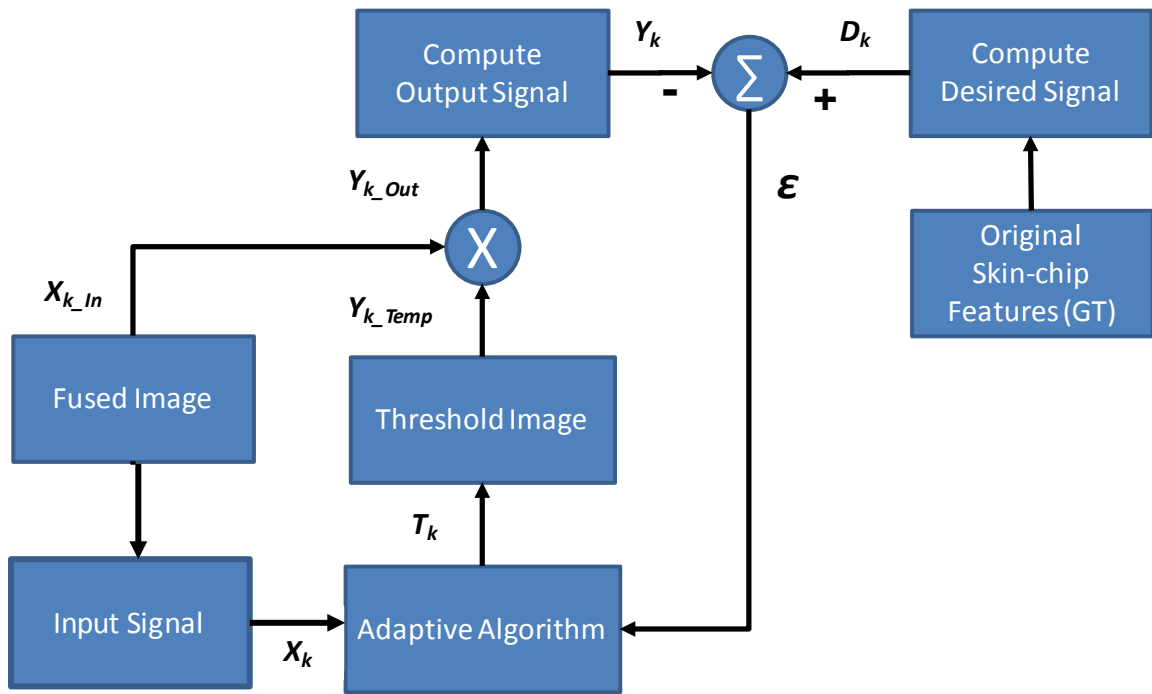


Figure 22: An adaptive threshold algorithm based on LMS optimization [40]

where the variables are defined as follows:

X_{k_In} : Fused image

X_k : Input signal

D_k : Desired signal computed from the original skin-ship features

T_k : Threshold value derived from the adaptive algorithm

Y_{k_Temp} : Image after applying threshold T_k

Y_{k_Out} : Image after multiplying the X_{k_In} with Y_{k_Temp}

Y_k : Output image signal computed from Y_{k_Temp}

ε : Error between Y_k and D_k

As explained in the previous section, the fused image X_{k_In} is constructed following Equation 22. To construct X_k signal, we follow the Equation 23.

$$X_k = \left(\frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N X_{k_In}(i, j) \right) \quad \text{(Eq. 23)}$$

where,

$$i = 1, 2, \dots, M; M = \text{number of rows of } X_{k_In}$$

$$j = 1, 2, \dots, N; N = \text{number of columns of } X_{k_In}$$

To calculate the desired signal D_k (GT signal) we used the skin chip that was automatically selected in the third chapter. For this skin chip, we calculated all the PCA-reported GLCM features and calculated the desired signal D_k using Equation 24 and Equation 25. A small random noise w_k is added to the D_k signal since in the real-world conditions it is not possible to have an exact statistical feature value of a target.

$$S_k(i, j) = \sqrt{I_{GT,fl}(i, j)^2 + I_{GT,f2}(i, j)^2 + \dots + I_{GT,fn}(i, j)^2} \quad \text{(Eq. 24)}$$

$$D_k = \left(\frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N S_k(i, j) \right) + w_k \quad (\text{Eq. 25})$$

where the variables are defined as follows:

S_k : Fused Ground Truth (GT, i.e., skin image)

$I_{GT,f\#}$: GT image of the GLCM feature value for PCA selected feature f# (#: 1,2, 3,...)

D_k : Composite feature signal value at the center of the skin chip

Initially, the threshold value (T_k) is set to an average value of the fused image (X_k) as defined by Equation 26.

$$T_k = \left(\frac{1}{MN} \right) \sum_{i=1}^M \sum_{j=1}^N X_{k_In}(i, j) \quad (\text{Eq. 26})$$

A few other parameters should be defined for the LMS adaptive threshold algorithm: 1) the number of iterations to reach an optimal threshold value and 2) the step size μ , a gain constant that regulates the speed and stability of adaption. In the first iteration, the very first skin-only image (Y_{k_Temp}) is calculated following Equation 27. To compute the Y_{k_Out} , we multiply the Y_{k_Temp} image with X_{k_In} image or mathematically, we follow Equation 28. Finally, the Y_k signal is computed by following the Equation 29

$$Y_{k_Temp}(i, j) = \begin{cases} 1, & X_k(i, j) > T_k \\ 0, & \text{otherwise} \end{cases} \quad (\text{Eq. 27})$$

$$Y_{k_Out}(i, j) = \begin{cases} X_k(i, j), & Y_{k_Temp}(i, j) = 1 \\ 0, & otherwise \end{cases} \quad (\text{Eq. 28})$$

$$Y_k = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N Y_{k_Out}(i, j) \quad (\text{Eq. 29})$$

Since Y_k is the resultant signal after applying T_k constraints onto the X_{k_In} , mathematically, Y_k is expressed as Equation 30 [40].

$$Y_k = T_k X_{k_In} \quad (\text{Eq. 30})$$

The error (ε) between the generated signal Y_k and desired signal D_k is calculated by following Equation 31.

$$\begin{aligned} \varepsilon_k &= D_k - Y_k \\ \varepsilon_k &= D_k - T_k X_{k_In} \end{aligned} \quad (\text{Eq. 31})$$

Employing the steepest descent type of adaptive algorithm illustrated in Equation 32, we compute the next threshold value using Equation 34. At the first iteration, the threshold (T_k) is computed employing the Equation 26 whereas starting from the second iteration, Equation 34 is employed to compute the threshold (T_{k+1}). We estimate the gradient at each iteration step following the Equation 33 [40].

$$T_{k+1} = T_k - \mu \hat{\nabla}_k \quad (\text{Eq. 32})$$

$$\hat{\nabla}_k = \frac{\partial \varepsilon_k^2}{\partial T_k} = 2\varepsilon_k \frac{\partial \varepsilon_k}{\partial T_k} = 2\varepsilon_k (-X_k) = -2\varepsilon_k X_k \quad (\text{Eq. 33})$$

$$T_{k+1} = T_k + 2\mu \varepsilon_k X_k \quad (\text{Eq. 34})$$

where the variables are defined as follows:

T_{k+1} : Next threshold value

μ : Step size

$\hat{\nabla}_k$: Gradient estimate of the squared error

Figure 23, part 1 illustrates a learning curve to approach an optimal mean-square error value and part 2 illustrates a weight convergence plot to reach an optimal weight (threshold) which is close to the GT value of the skin (D_k). Notice that the algorithm converges to an optimal weight at about iteration 2000. The MSE plot shows an effect of the quantization error close to the iteration 2000. At this iteration level, a step-size of 0.001 and a MSE of <0.001 impact the threshold value in the least significant digits only. Therefore, for the next iteration, a small change in the threshold value does not impact the MSE in the higher significant digits. However, when the threshold value increases at the higher significant bits level, a sudden change in the MSE is noticed (i.e., close to

iteration 2000 in Figure 23, part 1). The outcome of the algorithm is an image mask with skin represented in white (see Figure 24).

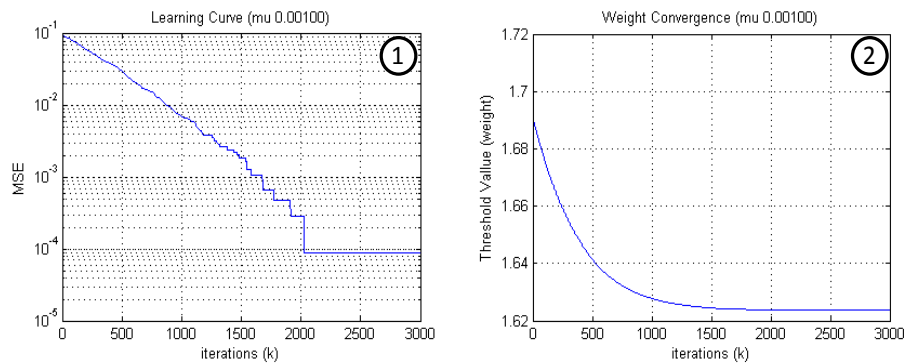


Figure 23: Adaptive optimized LMS algorithm parameters: 1) Learning curve for the MSE over k iterations and 2) Threshold weight convergence over k iterations

Figure 24, part 3 shows an output after applying an optimized adaptive LMS threshold algorithm to the fused image. Here the pixels in white represent skin pixels.

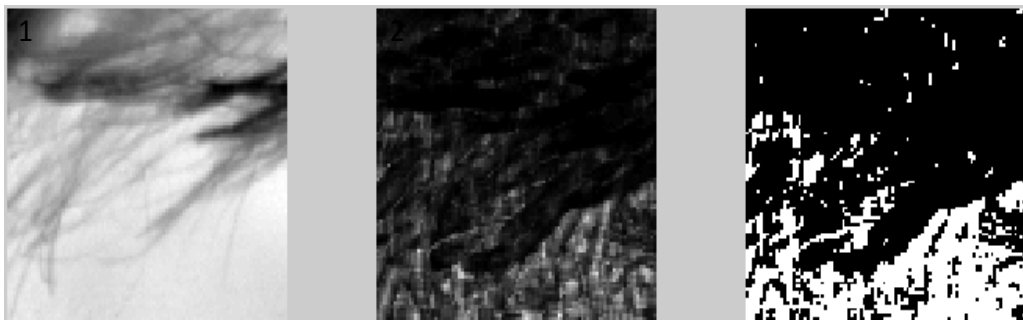


Figure 24: Optimized adaptive LMS results: 1) Original image, 2) Filtered image, and 3) Image after threshold (Skin only image)

This algorithm generates a skin-only image, but many skin pixels are not reported as such. Although resultant image statistics are closest to the statistics of our ground truth sample, not all skin regions in the original image have the same statistics. For example, statistics of regions close to the edges and hair are biased by the neighboring statistics. So instead of reporting these regions as skin regions, they are reported as non-skin regions. To overcome this challenge, in the next section, we explored classification with PCA algorithm for discriminating skin from non-skin pixels.

5.2 Classification with Principal Component Analysis (PCA)

PCA for classification is widely used for many applications; therefore, by taking advantage of the developed theory, we can apply PCA for skin classification [36]. The first task is to prepare a set of skin and non-skin samples by manually clicking on imagery and extracting $N \times N$ chips (see Figure 25). As an input to the PCA classification algorithm, set of extracted chips are used to formulate the skin and the non-skin classes. The robustness of the algorithm is achieved by defining multiple samples for skin and non-skin classes.

Figure 25 demonstrates intensity variations for both skin and non-skin classes. As shown in the Figure 25, the intensity range of the skin and non-skin pixels is very close. Therefore, classification is a very challenging problem. After randomly selecting five sample skin and five sample non-skin locations, we constructed a matrix as defined in Equation 35. Each column of this matrix is associated with either skin or non-skin class.

Multiple skin and non-skin samples are defined to take advantage of the skin variability in various subjects or data collected at different times.

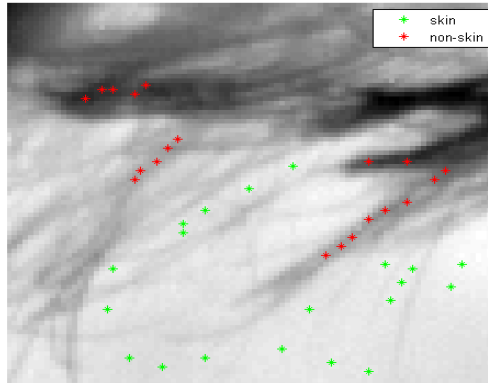


Figure 25: Manually selected skin (green) and non-skin samples (red) to construct two classes

$$X = \{SkinSpl_1, \dots, SkinSpl_L, NonSkinSpl_1, \dots, NonSkinSpl_L\} \quad (\text{Eq. 35})$$

where the variables are defined as follows:

$SkinSpl_L$ are the skin samples from the fused image; $L = 5$

$NonSkinSpl_L$ are the non-skin samples from the fused image; $L = 5$

A series of steps performed to classify skin and non-skin regions are listed below:

Steps 1 through 6 are the same as presented in the PCA feature selection section.

Step 7: Each sample Φ_i of the set A (calculate in step 3) can be represented as a linear combination of the top K eigenvectors. Following Equation 36, we compute $\hat{\Phi}_i$ eigenimages of the skin and non-skin samples.

$$\hat{\Phi}_i = \sum_{j=1}^K w_j u_j, \quad (w_j = u_j^T \Phi_i) \quad \text{(Eq. 36)}$$

where: i = total number of skin and non-skin samples

j = 1,2,..., K; K is total number of eigenvectors $\ll M$

w_j = weight of each class in the set A

Step 8: Each normalized skin and non-skin sample (Φ_i) of the set A is represented as Ω_i following Equation 37.

$$\Omega_i = [w_1 \quad w_2 \quad \dots \quad w_K]^T; \quad (w_j = u_j^T \Phi_i) \quad \text{(Eq. 37)}$$

Step 9: For each pixel in the fused image, we extract a $N \times N$ chip (Γ). As illustrated in Equation 38, prepare the sample chip Φ .

$$\Phi = \Gamma - \bar{x} \quad \text{Eq. 38)}$$

where, \bar{x} : mean sample (as per following equation in step 1)

Step 10: We project this sample chip onto the eigenspace to construct an eigenimages following Equation 39.

$$\hat{\Phi} = \sum_{j=1}^K w_j u_j, \quad (w_j = u_j^T \Phi) \quad (\text{Eq. 39})$$

Step 11: Then we represent Φ as Ω following Equation 40 .

$$\Omega = [w_1 \quad w_2 \quad \dots \quad w_k]^T, \quad (w_j = u_j^T \Phi) \quad (\text{Eq. 40})$$

Step 12: Following Equation 41, we compute error between the Ω_i and Ω .

$$e_i = \sqrt{(\Omega(1) - \Omega_i(1))^2 + (\Omega(2) - \Omega_i(2))^2 + \dots + (\Omega(k) - \Omega_i(k))^2} \quad (\text{Eq. 41})$$

Step 13: As discussed earlier, multiple skin and non-skin samples for two classes were constructed to take advantage of data variation in each class. We defined the first five samples as skin class and the last five samples as non-skin class. Therefore, following Equation 42, we determine each pixel's association with the skin or non-skin class.

$$S(i, j) = \begin{cases} 1, & \text{if } (e_p)_{i,j} < (e_q)_{i,j}; \\ 0, & \text{otherwise} \end{cases} \quad (\text{Eq. 42})$$

where, $p = 1,2,3,4,5$; $q = 6,7,8,9,10$

Following steps 9 and 13, we construct a skin-only image with skin pixels in white and non-skin pixels in black as demonstrated in the Figure 26. But we notice that there are remaining skin pixels that are not reported as such. After studying that the skin and non-skin classes overlap, we explored another classification algorithm called Linear Discriminant Analysis (LDA) that we discuss in the next section.

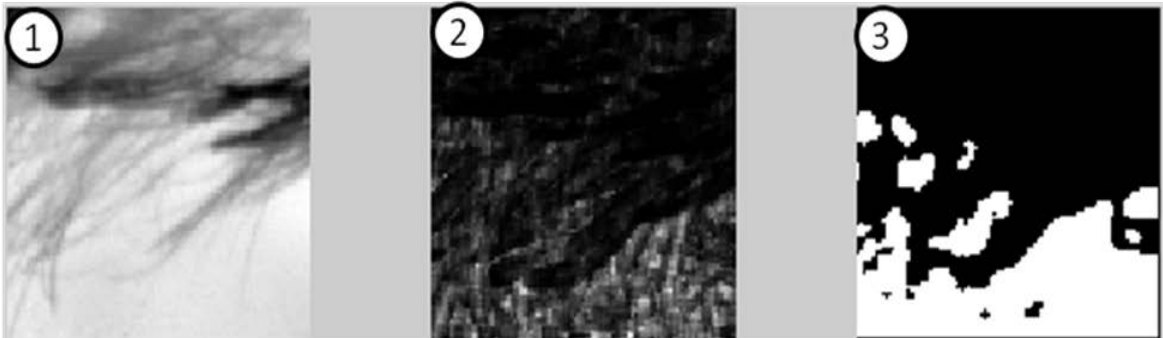


Figure 26: PCA classification results; 1) Image, 2) PCA transformed image, and 3) PCA classified skin mask

5.3 Classification with Linear Discriminant Analysis (LDA)

LDA operation is applied to the fused image data to maximize discriminatory power. Unlike PCA which represents information in an orthogonal linear space, LDA represents discriminatory information in a linearly separable space whose bases are not necessarily orthogonal [10] [25].

The LDA algorithm finds a linear transformation that is used for transforming the skin and non-skin classes to increase separation between these classes. This can be achieved

through scatter matrix analysis [25]. For the scatter analysis, we have adopted Fisher's criterion, defined as the ratio of the determinant of the between-class scatter matrix of the projected samples to the within-class scatter matrix of the projected samples [10]

To apply the LDA algorithm to the data for skin discrimination, the first step is to construct skin and non-skin classes by manually clicking on the imagery, as demonstrated in the previous section. Equation 43 demonstrates the format of these two classes.

$$\begin{aligned}
 X &= \{ [Class_1] \ [Class_2] \} \\
 [Class_1] &= [Class_{1,Feature_1} \ Class_{1,Feature_2} \ \dots \ Class_{1,Feature_K}] \\
 [Class_2] &= [Class_{2,Feature_1} \ Class_{2,Feature_2} \ \dots \ Class_{2,Feature_K}]
 \end{aligned} \tag{Eq. 43}$$

where: $Class_1$: Skin class

$Class_2$: Non-skin class

K = total number of PCA selected features

Our goal is to construct a simplified X matrix with two classes, with each class having only two features sets. Equation 44 demonstrates a simplified version of the class matrix.

$$X = \{ [Class_{1,Feature_a} \ Class_{1,Feature_b}] [Class_{2,Feature_a} \ Class_{2,Feature_b}] \} \tag{Eq. 44}$$

where $Feature_a$ and $Feature_b$ are combined features. For this particular example,

four features were selected by PCA; therefore, $Class_{p,Feature_a}$ and $Class_{p,Feature_b}$ are

computed employing Equation 45.

$$\begin{aligned}
Class_{p,Feature_a}(i, j) &= \sqrt{Class_{p,Feature1}(i, j)^2 + Class_{p,Feature2}(i, j)^2} \\
Class_{p,Feature_b}(i, j) &= \sqrt{Class_{p,Feature2}(i, j)^2 + Class_{p,Feature3}(i, j)^2}
\end{aligned}
\tag{Eq. 45}$$

where: $p = 1, 2$

The procedure adopted to combine classes was also applied for combining feature images. These combined feature images are later mapped onto the LDA-generated space for comparison with the skin and non-skin classes.

Figure 27 shows a data spread for a sample test image in blue, skin class in green, and non-skin class in red stars. Both skin and non-skin classes overlap; therefore, a confusion region exists between classes. In that confusion region, the PCA algorithm was not able to clearly distinguish skin and non-skin pixels. Therefore, we needed an algorithm that could separate these two classes with maximized inter-class distance.

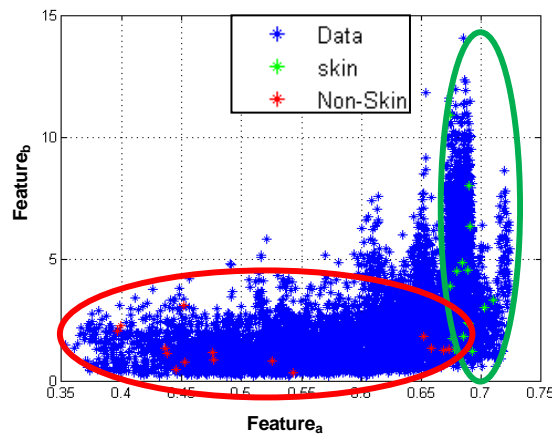


Figure 27: Feature data: Sample test data (blue), Skin class (green), and Non-skin class (red)

A set of steps followed to employ the LDA algorithm for skin classification are listed below:

Step 1: First we calculate the mean of the classes following the procedure discussed in PCA for classification section. This process generates two mean values, μ_1 and μ_2 . Arrange the mean values as per Equation 46.

$$\mu_p = \begin{bmatrix} \mu_{Class_{p,a}} \\ \mu_{Class_{p,b}} \end{bmatrix} \quad (\text{Eq. 46})$$

where:

$p = 1, 2$. (i.e., Class1, Class 2); $a = \text{Feature}_a$; $b = \text{Feature}_b$ (See Figure 27)

Step 2: Using Equation 47, we compute the covariance matrix for each class.

$$S_p = \sum_{p=1}^2 (X_p - \mu_p)(X_p - \mu_p)^T \quad (\text{Eq. 47})$$

Step 3: We calculate a within-class matrix (S_w) following Equation 48, where the covariance matrices are combined to calculate the within-class scatter for both classes. Within-class scatter represents the distance between a sample and other samples in that class.

$$S_w = \sum_{p=1}^2 S_p \quad (\text{Eq. 48})$$

Step 4: We determine a between-class scatter (S_B) employing Equation 49. Here the covariance is calculated between the means of two classes to represent the closeness between the two classes. Smaller between-scatter value represents a potential overlap between the two classes.

$$S_B = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T \quad \text{(Eq. 49)}$$

Step 5: An optimal, LDA driven projection vector w^* is obtained following Equation 50. This projection vector is calculated such that after projecting onto this vector, separation between the skin and non-skin classes statistically should increase. For an improved separation between the two classes after projecting both classes onto the LDA driven vector 1) the Euclidean distance between the mean locations of the two classes should increase and 2) the standard deviation of each class should reduce.

$$w^* = S_w^{-1}(\mu_1 - \mu_2) \quad \text{(Eq. 50)}$$

Finally, after calculating the scatter between classes and within classes, Fisher's criterion can be expressed in terms of S_W and S_B as demonstrated in Equation 51. This implies that $J(w)$ is a measure of the difference between class means normalized by a measure of the within-class scatter matrix.

$$J(w) = \frac{w^T S_B w}{w^T S_w w} \quad \text{(Eq. 51)}$$

To find the maximum value of $J(w)$, differentiate and equate the resulting equation to zero. This metric can be used for analyzing the statistics of both classes after projecting them onto the projection vector.

Step 6: The skin and non-skin class data is projected onto the derived projection vector by following Equation 52. Figure 28 illustrates the projected vector within the feature domain (part 1) and the classes projected onto the vector domain (part 2). In these plots, red and green arrows show the mean locations for the non-skin class and skin classes, respectively.

$$Xp_i = X_i(w^*)^T \quad (\text{Eq. 52})$$

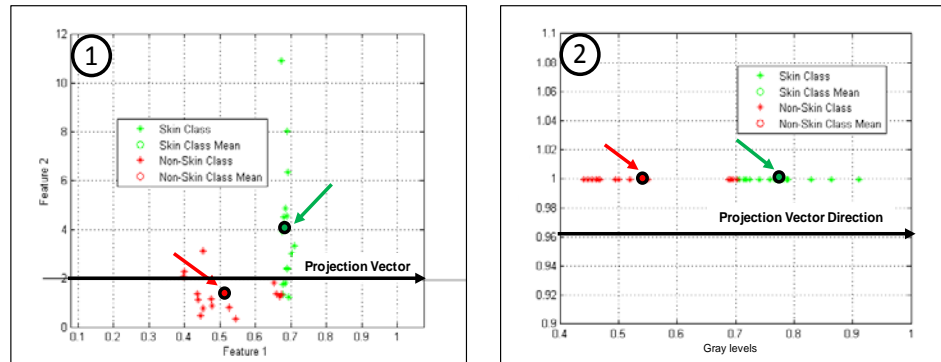


Figure 28: LDA results: 1) LDA generated vector projection and 2) Classes data projection onto LDA generated vector

Following Equation 51, we project every single pixel of the PCA-selected feature data onto the LDA space domain. Figure 29 Part 1 shows each data pixel projection onto the LDA-generated vector and the resultant skin-only image mask.

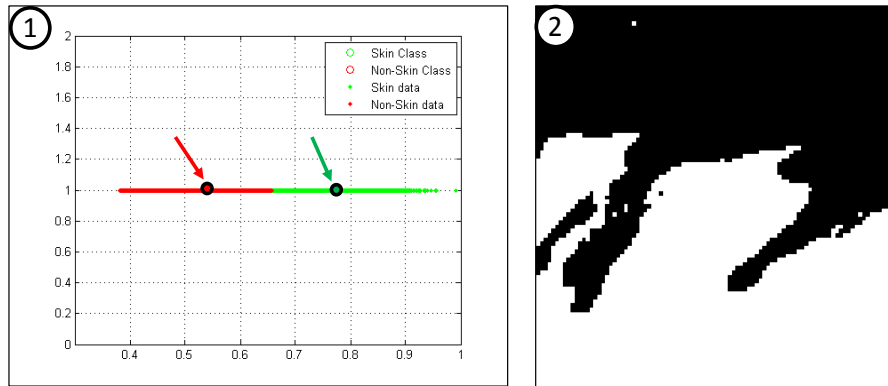


Figure 29: LDA Generated results: 1) Data pixels projection onto the LDA-generated vector and 2) Skin mask after LDA classification

After obtaining the skin-only images as an output from all three classification algorithms, a post-processing step is employed to remove artifacts. The next section discusses an adopted post-processing procedure.

5.4 Post-processing

In the post-processing step, the goal is to clean the final results by removing any outliers (i.e., single or small groups of hanging pixels). Post-processing can be a very crucial step depending on the application. For our application, we want to identify the maximal skin area within the image, so we want to remove hanging pixels that can possibly be thin

hair. These regions represent false detections and are highly dependent upon the original size of the GLCM kernel. For example, if a kernel size is bigger than a thin hair, depending upon the neighboring pixel, the thin hair region will be declared skin or non-skin region.

For this part, we have applied a morphological operation called “opening” which is discussed in the second chapter. Figure 30 demonstrates the results after performing opening operations on skin-only images generated by each of the classification algorithms. A structuring element size of 3x3 pixels was employed for the opening process that helped with removing small hanging pixels.

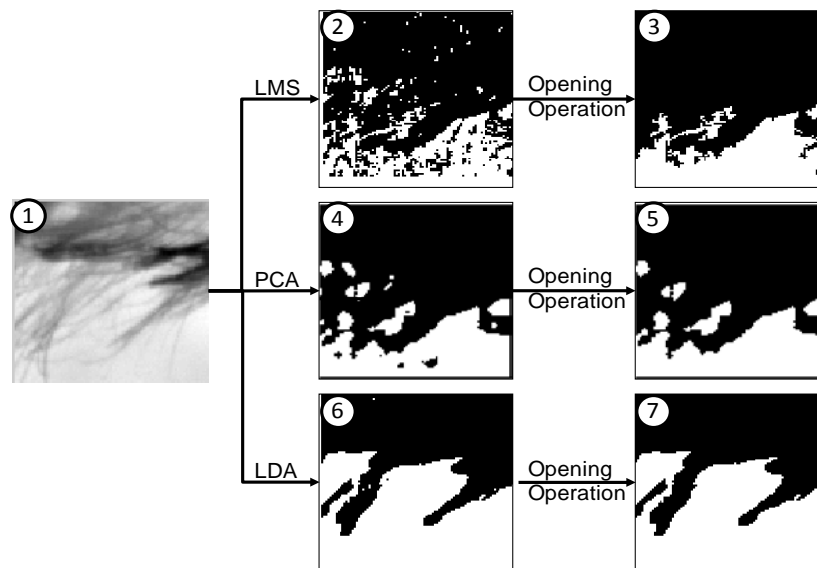


Figure 30: Sample results after post-processing - 1) Original image; 2) Image after adaptive optimized LMS threshold; 3) Skin-only image after post-processing; 4) Image after PCA classification; 5) Skin-only image after post-processing; 6) Image after LDA classification; 7) Skin-only image after post-processing;

6 Validation and Data Analysis

For testing purposes, we have used a small set of thermal data that was collected under an IRB-approved protocol for human testing using a MWIR (320×240 resolution) imager under a controlled environment. This thermal data includes skin portions of the face and non-skin portions such as background, hair, glasses, etc. Because there are no methods available to establish ground truth skin data in thermal imagery, 16 frames were used for manually selecting skin and non-skin locations.

6.1 Algorithm Validation Procedure

Figure 31 demonstrates an algorithm validation procedure for results generated by three skin classification algorithms. There are three major parts of the process: 1) Algorithm generated output skin-only mask, 2) manually-selected GT locations, and 3) comparison between the GT, skin, and non-skin locations.

Figure 31, part 1 is described in the previous chapters in detail. The first step of the algorithm reduces the feature dimensionality by applying PCA. The second step is to apply one of the three classification approaches (LMS or PCA or LDA) to generate a skin-only image. After the skin-only mask is generated, we prepare an inverted skin-only image to be compared with the non-skin ground truth locations.

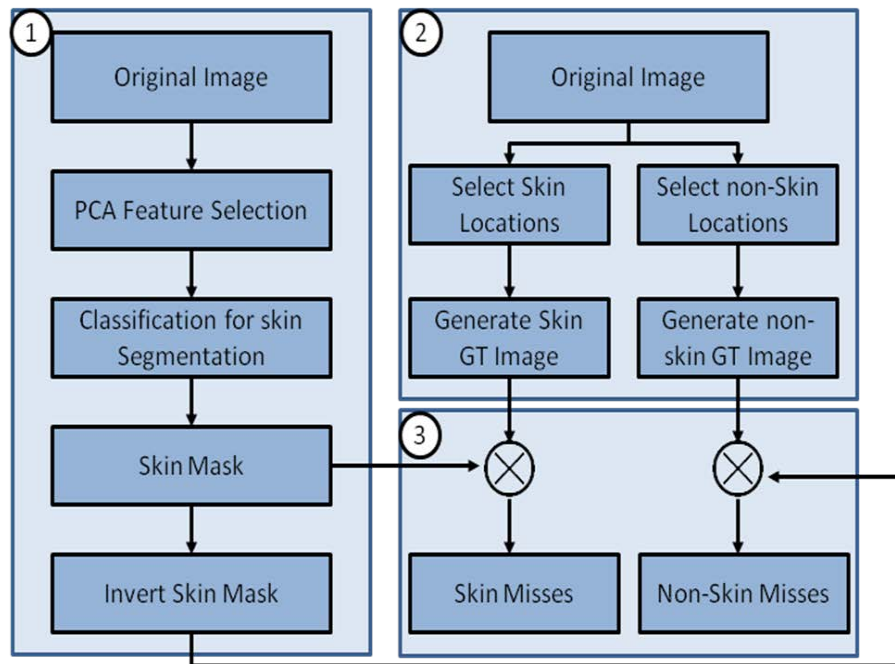


Figure 31: Validation procedure flow diagram: 1) Algorithm to generate skin-only image; 2) Generate manual ground truth; 3) GT and algorithm output comparison

Figure 31, part 2, shows a flow chart for generating ground truth (GT) locations and GT images with skin and non-skin locations. For validation process, we have selected 16 subjects and about 50 skin and 50 non-skin GT locations on each subject using 3 operators. As shown in Figure 32, after selecting the GT points from the original image, the algorithm locates and generates skin and non-skin GT images using zeros and ones at the skin and non-skin GT locations, respectively.

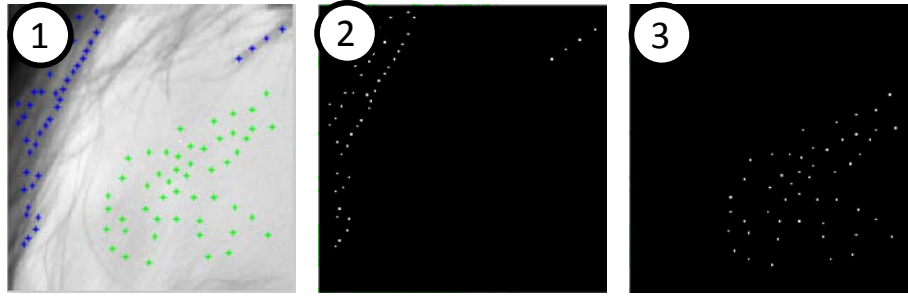


Figure 32: Ground truth data: 1) Manual ground truth (non-skin in blue and skin in green stars); 2) Image with only non-skin locations; 3) Image with only skin locations

Figure 31, part 3 compares the algorithm generated (LMS, PCA, and LDA) skin and non-skin locations, and manually-selected skin and non-skin locations. The outcome of this section is a list with number of correctly classified and incorrectly classified pixels. A correctly classified is a number of true detections that includes two cases where: 1) a skin pixel is declared as a skin pixel and 2) a non-skin pixel is declared as a non-skin pixel. Whereas the incorrectly classified number represents false detections and includes two cases where: 1) a skin pixel is declared as a non-skin pixel and 2) a non-skin pixel is declared as a skin pixel.

To compare the GT skin locations with the algorithm-generated skin mask, we multiply the inverted skin mask image with the skin GT image. As demonstrated in Figure 33, the output of this process should have ones at the locations where the algorithm failed to detect skin. These ones are the total number of incorrectly classified skin locations.



Figure 33: 1) GT skin image 2) Algorithm output non-skin image with non-skin in white; 3) Algorithm missed non-skin locations

To test the incorrectly classified non-skin locations, the skin mask is multiplied with the non-skin GT image as demonstrated in Figure 34. The output of this process should have ones at the incorrectly classified non-skin locations. Finally, the total incorrectly classified skin and non-skin pixels are added, and then subtracted from the total GT points to construct a list with correctly and incorrectly classified pixels as demonstrated in Figure 35.



Figure 34: 1) GT non-skin image 2) Algorithm output skin-only image with skin in white; 3) Algorithm missed non-skin locations

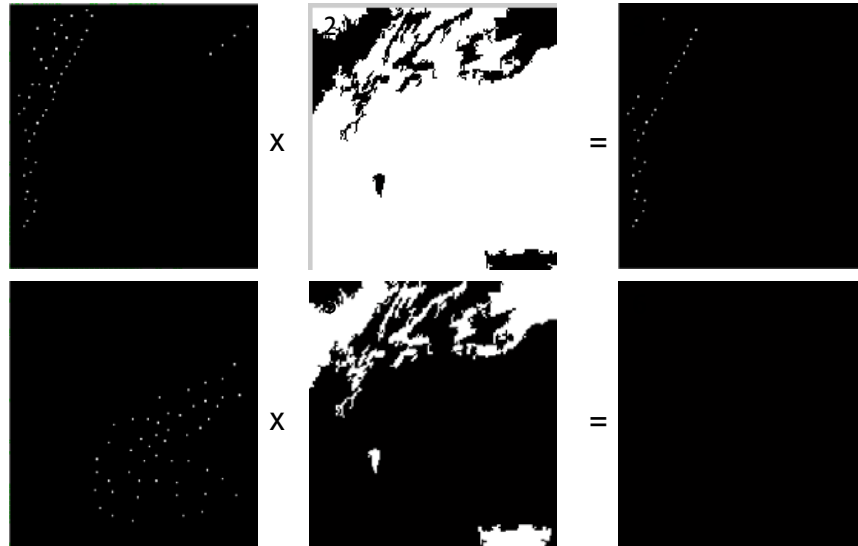


Figure 35: 1) GT non-skin image 2) Algorithm output skin-only image with skin in white; 3) Algorithm missed non-skin locations; 4) GT skin image 5) Algorithm output non-skin image with non-skin in white; 6) Algorithm missed non-skin locations

After comparing algorithm outputs with the GT skin and non-skin locations, we notice a significant improvement in the results with the LDA algorithm. To confirm our hypothesis that LDA has the highest correct pixel classification rate (true detections) as compare to the LMS and the PCA generated results, we perform a set of operations listed above on all 16 thermal images and generate a list of correctly and incorrectly classified pixels to perform further analysis.

6.2 Data Analysis

To perform correctly and incorrectly classified pixels analysis on a set of sixteen subjects, we use the list of correctly and incorrectly classified pixels that we have obtained

performing the operations listed in the previous section. Figure 36 demonstrates the percentage of correctly classified pixels by the LMS, PCA, and LDA algorithms. As we notice from the bar chart, the PCA algorithm has the least number of correctly classified pixels and the LDA algorithm has the maximum number of correctly classified pixels. We thus conclude that the LDA algorithm is the best algorithm for skin detection of the ones tested.

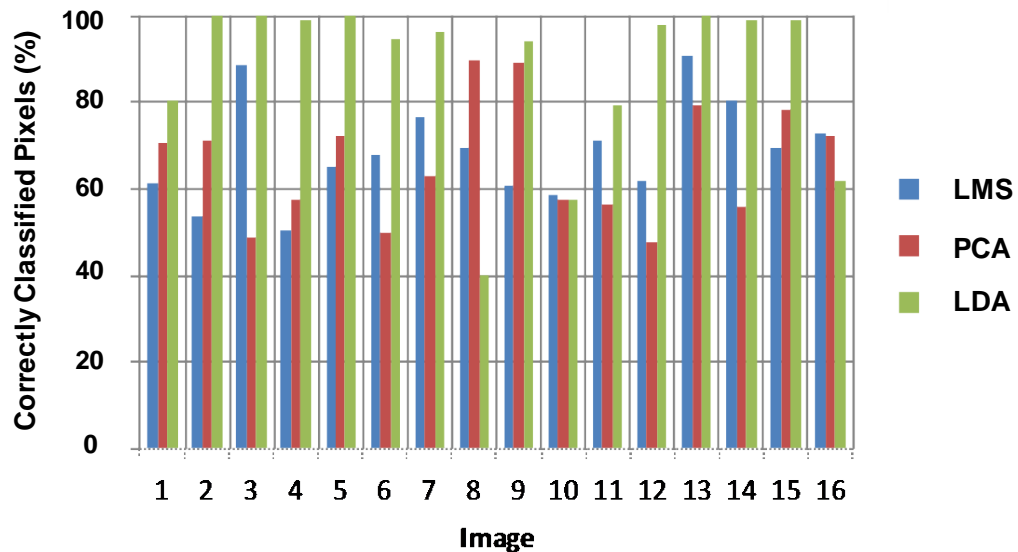


Figure 36: Actual true detections with LMS, PCA and LDA algorithms

Data points for each algorithm shown in Figure 36 are averaged and presented in a simpler form as demonstrated in Figure 37. Figure 37 demonstrates that the LDA algorithm outperforms the LMS and PCA algorithms. Notice that correctly classified

pixels for the LDA algorithm are about 80%, whereas the LMS and PCA correctly classified pixels are 68% and 65%, respectively. Also note that the error bars for the LMS and PCA algorithms are overlapping which implies non-significant difference between two data sets. Whereas the error bar for the LDA algorithm is standing alone which implies a significant difference between the LDA data and the other two data sets (PCA and LMS).

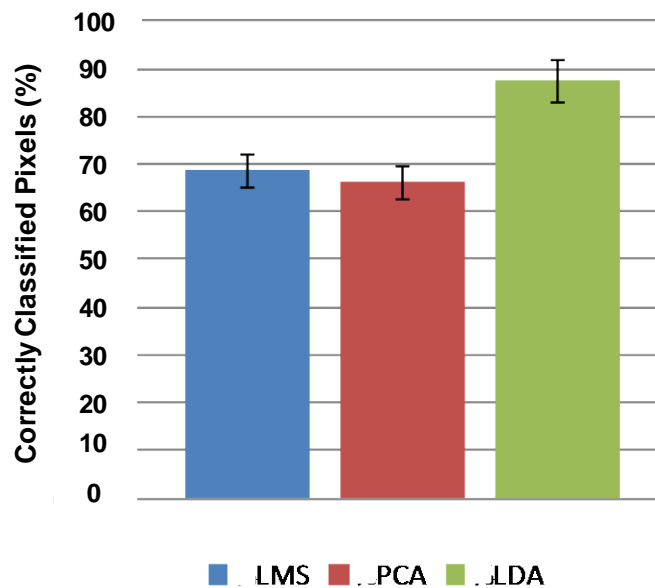


Figure 37: Average true detections with LMS, PCA and LDA algorithms (After comparing GT with each algorithm output)

For further verification, we perform a well known and widely used statistical test called the paired t-test on the data from all three algorithms. The paired t-test is used when there is one observation for each experiment such as a number of correctly classified

pixels for each of the classification algorithms [14]. To run a paired t-test, we need two inputs, 1) the number of correctly classified pixels for each classification algorithm, and 2) a set of employed algorithms, (LMS, PCA and LDA). The number of correctly classified pixels varies from subject to subject among the three classification algorithms [14].

By running a repeated measures Analysis of Variance (ANOVA) test on more than two sets and a t-test on two data sets, a p-value is computed which is a probability measure between 0 and 1. Moreover, the p-value is a probability measure against our null hypothesis. For our case, null hypothesis is that results from the LMS, and the PCA are as good as from the LDA. A higher p-value shows not a significant difference between the two data sets and an agreement with our hypothesis whereas a smaller p-value is a probability measure against our null hypothesis [14].

In order to investigate the significance of differences in classification accuracy between the three algorithms, a repeated measures Analysis of Variance (ANOVA) test was performed. The resulting p-value is 0.0008, which indicates the presence of significant differences. Follow-up paired t-test resulted in p-values shown in table 1. Based on Bonferroni corrected threshold of $0.05/3$, the difference between LDA and other two algorithms is significant [14].

Table 1 presents a computed p-value between the algorithms' outputs. As clearly demonstrated in Table 1, the p-value between the PCA and the LMS show a high/not-significant (~ 0.20) value whereas the values are much smaller when the LDA is

compared with the LMS or the PCA. Normally, significantly different results with a p value $\ll 0.01$ are presented as $<0.01^{**}$ to show the importance of the results [14]. A lower p-value indicates that the results generated by a limited data set are as good as the results generated by a large data set.

Table 1: Paired t-test results between LMS, PCA and LDA algorithms results

	LMS	PCA	LDA
LMS	1	.20 NS	$<0.01^{**}$
PCA		1	$<0.01^{**}$
LDA			1

Since the LDA algorithm outperforms the LMS and the PCA for skin discrimination, we hypothesize that the Fisher's criteria helps with a better separation of the skin and non-skin classes. Discussion in the next section proves our hypothesis and explains why and how the LDA algorithm takes advantage of the Fisher's criteria.

6.2.1 Further Analysis of the LDA performance

An LDA algorithm employing Fisher's criteria provides a projection vector for the original data transformation onto a vector space. The projection vector is calculated such that the scatter between the classes is maximized and scatter within each class is minimized [10]. This property of the LDA algorithm introduces an increased Euclidean

distance between two classes; therefore, it produces an improved skin-only image as compared to the LMS or PCA algorithms.

Figure 38 demonstrates the data spread for both skin and non-skin classes. As clearly demonstrated in Figure 38, both skin and non-skin samples and classes overlap. This implies that the skin and the non-skin classes share some common gray levels and common texture patterns. The pixels in those common gray levels and common texture patterns can be interpreted as skin or non-skin pixels.

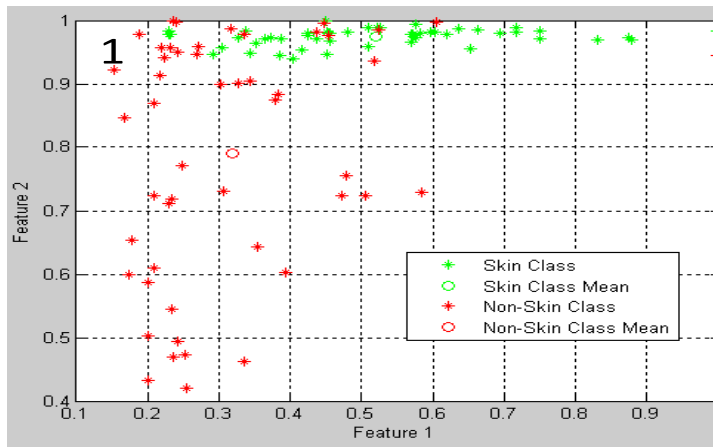


Figure 38: LDA data projection- original data scatter

Further, the LDA algorithm-generated projection vector provides a vector space onto which both skin and non-skin classes are projected. Figure 39 shows the projection vector along the two classes before projection, whereas Figure 40 demonstrates an improved separation between the two classes after projecting onto the LDA-driven projection vector.

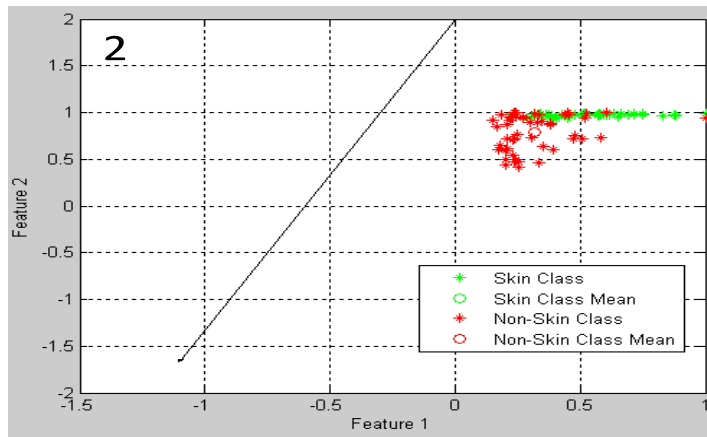


Figure 39: LDA data projection- original data and projection vector

Although an improved separation is observed in the data (Figure 40), an analytical metric is needed to confirm the improvement. This improvement is measured in terms of scatter between classes and within each class before and after applying the LDA algorithm.

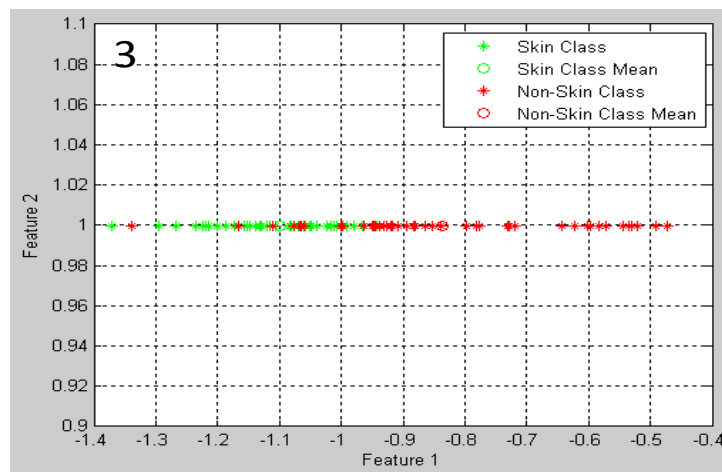


Figure 40: LDA data projection- data after projecting onto the projection vector

As stated in the fifth chapter, within-class scatter is a measure of data points gray level variability in terms of their means and standard deviations. The between-class scatter is a measure of variance between the two clusters (skin and non-skin), which is calculated employing the covariance matrix (described in the LDA classification algorithm section). Figure 41, part 1 demonstrates a scatter between two classes (S_B) and within each class (S_{W1} & S_{W2}). For both skin and non-skin classes, a convex hull is drawn capturing all the associated pixels within a boundary (red for non-skin class and green for skin class) to represent the boundary of two classes/clusters. Approximately half of the skin cluster overlaps with the non-skin cluster. Moreover, between-class scatter is very small and their mean values for both clusters reside in the region of non-skin class

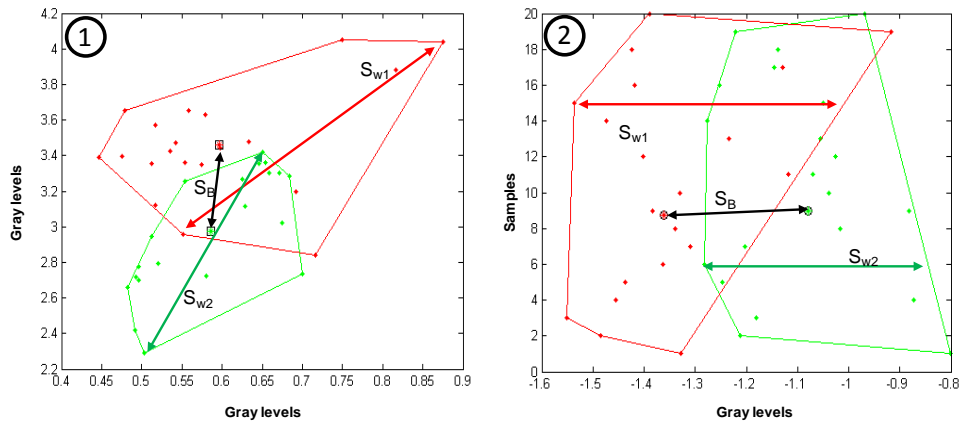


Figure 41: Skin (green) and non-skin (red) classes: 1) Before LDA projection; 2) After LDA projection

The problem of incorrectly classified pixels is resolved after the skin and non-skin class data are projected onto the LDA-calculated projection vector. Figure 41, part 2 demonstrates the within-class and between-class scatter of both skin and non-skin classes after projection onto the LDA-generated projection vector. After projection, 2-D data is transformed into 1-D. Figure 41, part 2 demonstrates the 1-D data with vertical axis showing samples (20 pixels for each class) and horizontal axis showing the projected gray level for each class. As the gray levels are plotted in the horizontal direction, skin and non-skin within-class and between-class scatter is measured in the horizontal direction. A close observation of Figure 41, part 2 demonstrates that the scatter within each class is almost equal. Moreover, the scatter between the two classes after LDA projection is increased compared to the scatter between the original classes. Further, we notice that after projection, each class's mean resides within its own class whereas the mean of the original skin cluster was within the non-skin cluster (Figure 41 part 1). This implies that projected classes onto the LDA projection vector have a noticeable separation.

Table 2: Between-class and within-class scatter values for skin and non-skin classes

Gray Levels	Orginal Classes	Classes Afer LDA Projection
Scatter Between (SB)	0.0049	0.065
Scatter Within (SW1)	0.19	0.14
Scatter Within (SW2)	0.14	0.15

Mathematical interpretation of scatter improvement is shown in Table 2. Table 2 demonstrates scatter values (between and within classes) for original classes and projected classes (classes after applying LDA). For this particular example, between-class scatter for the original classes is 0.0049, and is 0.065 after applying LDA, which is an improvement factor of 13. However, within-class scatter for both projected classes is adjusted close to the minimum standard deviation of the two classes before (minimum of {0.19, 0.14}) they were projected onto the LDA driven projection vector. This implies that both classes become confined clusters with a low standard deviation and therefore can help with segmenting skin pixels from non-skin pixels. Table 2 shows a within-class scatter of 0.19 and a non-skin original class scatter of 0.14. However, after projecting both classes onto the LDA projection vector, within-class scatter for skin and non-skin classes is adjusted to 0.14 and 0.15 respectively.

7 Conclusions

There have been many studies in the areas of skin detection in the visible spectrum, but we have extended our studies into the infrared band and found a viable end-to-end solution for skin detection. We have developed an understanding of skin texture characteristics and an image processing tool to detect human skin in single thermal band (MWIR) images.

As thermal sensors are capable of capturing data during both day and night conditions, our methodology now allows for human skin detection during these conditions. We have utilized PCA to understand skin features from Gray Level Co-occurrence Matrix (GLCM)-driven skin texture features. This process has helped with reducing the features' dimensionality, improving results, and increasing processing speed. The LMS, PCA, and LDA algorithms were employed to discriminate skin from non-skin pixels in thermal images.

We have successfully analyzed results from all three classification techniques and recognized the best classification algorithm. After conducting a deeper analysis on a data set employing the LDA algorithm, we have learned that projecting both classes onto the LDA-driven projection vector improves the between-classes scatter by factor of ~ 13 . By

comparison, after projecting the original classes onto the LDA-driven projection vector, the within-classes distance is reduced to about the minimum of original classes scatter (~0.14). Therefore, the mini-max property of the LDA algorithm helps with generating the best skin-only results compared to the LMS and PCA algorithm.

In general, this research provides a generalized approach for human skin detection in thermal images, providing a non-contact, remote, and passive method for human skin detection in day or night imagery for security and military applications.

7.1 Open Questions

We have learned that the algorithm declared skin and non-skin regions are sensitive to the GLCM kernel size which is used for generating the GLCM-driven features. A bigger window size falsely determines a given pixel's association based on the majority of the skin or non-skin pixels within that window. Determining an optimal window size to generate the best results (maximal skin-only and minimum non-skin regions) will be a valuable addition to this research.

At present, this algorithm is designed and tested with high-resolution MWIR images only. Future investigations of expanding this research for detecting facial skin in the bands of uncooled sensors (LWIR) will also add to the human skin detection community.

7.2 Productivity of the Research

Three papers were published and presented to the scientific community in conferences.

Two papers cleared for public release are listed below:

1. Kaur, B., Hutchinson, A.J., Nelson, J.K., A. O’Kane, B. "Human Facial Skin Detection in Thermal Imagery." Military Sensing Symposium (MSS),. 2012
2. Kaur, B., Hutchinson, A.J, Leonard, K.R, and Nelson, J.K. “Human Facial Skin Detection in Thermal Video to Effectively Measure Electrodermal Activity (EDA)”. SPIE DSS conference at Fl, Orlando Apr. 2011.

References

References

- [1] Ahmad, M. S., Shahid M.N., Nisa, M. "Application of texture analysis in the assessment of chest radiographs." *International Journal of Video & Image Processing and Network Security IJVIPNS* Vol: 9 No: 9
- [2] Albrechtsen, Fritz. "Statistical Texture Measures Computed from Gray Level Concurrence Matrices." *Image Processing Laboratory, Department of Informatics, University of Oslo*. Nov. 2008.
- [3] Anderson, R. and Parrish, J. *The Optics of Human Skin*, vol. 77, pp. 13–19. 1981
- [4] Attas, M, et al., "Visualization of cutaneous hemoglobin oxygenation and skin hydration using near-infrared spectroscopic imaging", *Skin Research and Technology*, vol. 7, pp. 238-245. 2001,
- [5] Baraldi, A. and Parmiggiani, F., "An investigation of the Textural Characteristics Associated with Gray Level Co-occurrence Matrix Statistical Parameters." *IEEE Trans. On Geoscience and Remote Sensing*, vol. 33, no. 2, pp. 293-304, 1995.
- [6] Brand, J. and Mason, J. "A comparative assessment of three approaches to pixel-level human skin-detection," *Proceedings of 15th International Conference on Pattern Recognition*, vol. 1, pp. 1056–1059, Sep. 2000.
- [7] Brooks, A.L. "Improved Multispectral Skin Detection and its Application to Search Space Reduction for Dismount Detection Based on Histograms of Oriented Gradients." *Wright-Patterson Air Force Base, Ohio*. Thesis, 2010
- [8] Cronin, J.G. and Wright J. Rapid assessment and initial patient treatment team – a way forward for emergency care. *Accident and Emergency Nursing*. pp 87-92.2005;
- [9] Dinesh, S., "Application of Opening by Reconstruction to Characterize the Size Distribution of Catchments Extracted from Digital Elevation Models." *Applied Mathematical Sciences*. vol. 1, no. 13, pp. 615-628, 2007.

- [10] Duda, R.O., Hart, P.E., and Stork, D.G. Pattern Classification. USA & Canada: John Wiley & Sons, Inc. pp 117-134., 2001
- [11] Emergency Nurse Association. Making the Right Decision: A Triage Curriculum 2nd Edition, Emergency Nurses Assn. Des Plains, IL. 2001
- [12] Fact sheet: Civil air patrol archer system, <http://atg.cap.gov>, August 2005.
- [13] Freedman, L. W., Scerbi, A. S., Dawson, M. E., Raine, A., McClure, W. O., & Venables, P. H. "The relationship of sweat gland count to electrodermal activity." *Psychophysiology*, 31, pp 196-200, 1994.
- [14] Friston, K.J., Ashburner, J.T., Kiebel, S.J., Nichols, T.E., and Penny, W.D. "Statistical Parametric Mapping: The Analysis of Fundamental Brain Images." Elsevier, United Kingdom. pp 16,117. 2010.
- [15] Gonzalez, R. C. and Woods, R. E., "Digital Image Processing." Upper Saddle River, N.J.: Prentice-Hall, 2002.
- [16] Haralick, M., Shanmugam, K., Dinstein, I., "Textural Features for Image Classification." *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-3, no.6, pp.610-621, 1973.
- [17] Irvine, J. M., Wiederhold, B. K., Gavshon, L. W. et al., "Heart rate variability: a new biometric for human identification," in *Proceedings of the International Conference on Artificial Intelligence (IC-AI '01)*, Las Vegas, USA, pp. 1106-1111, Jun. 2001.
- [18] Jiang, Z., Wu, Z., Yao, M. "Skin Detection on Images with Color Deviation" *Broadband, Wireless Computing, Communication and Applications (BWCCA)*. pp. 171 – 174, 2010
- [19] Jolliffe I.T. *Principal Component Analysis*, Series: Springer Series in Statistics, 2nd ed., Springer, NY, XXIX, 487 p, 2002
- [20] Kaur, B., Hutchinson, A.J, Leonard, K.R, and Nelson, J.K. "Human Facial Skin Detection in Thermal Video to Effectively Measure Electrodermal Activity (EDA)". *SPIE DSS conference at Fl, Orlando 2011*.
- [21] Kaur, B., Hutchinson, A.J., Nelson, J.K., A. OKane, B. "Human Facial Skin Detection in Thermal Imagery." *Military Sensing Symposium (MSS)*, 2012
- [22] Kerekes, J.P. and Baum, J.E., "Hyperspectral Imaging System Modeling," *Lincoln Laboratory Journal*, 14, 1, pp. 117-130, 2003

- [23] Khandait, S.P.; Thool, R.C. "Hybrid Skin Detection Algorithm for Face localization in facial Expression Recognition". Advance Computing Conference. IACC. IEEE International. pp. 398 – 401, 2009
- [24] Krzywicki, A. T., He, G. G., Burns, R. P., Martin, D. J., Berntson, G. G., and O’Kane, B. L.. "A non-contact technique for measuring eccrine sweat gland activity." In preparation. 2011
- [25] Martinez, A. M. and Kak A. C. "PCA versus LDA". IEEE Transaction on Pattern Analysis and Machine Intelligence. Vol. 23, No. 2, pp. 228-233. 2001.
- [26] Miranda, A. A., Le Borgne, Y. A. and Bontempi, G. "New routes from minimal approximation error to principal components," Volume 27, Number 3. Neural Processing Letters, Springer, Jun. 2008
- [27] Nallaperumal, K., Subban, R., Babu, C. N. K., Selvakumar, R.K., Fred, L.A., Seldev, C., and Vinsley, S.S. "Skin Detection using Color Pixel Classification with Application to Face Detection: A Comparative Study." International Conference on Computational Intelligence and Multimedia Applications. 2007
- [28] NASA Office of Mission to Planet Earth, "A Guide to Teaching Earth System Science". Mar. 1994
- [29] Nunez, A.S. "A Physical Model of Human Skin and Its Application for Search and Rescue." Air Force Institute of Technology. Wright-Patterson Air Force Base, Ohio. Dissertation. 2009.
- [30] O’Kane, B. "Dynamics of human thermal signatures". Proceedings of the Inframation Conference. Las Vegas, Oct. 2004.
- [31] Partio, M, Cramariuc, B., Gabbouj, M., Visa, A., "Rock texture retrieval using gray level co-occurrence matrix." In 5th Nordic Signal Processing Symposium, 2002.
- [32] Polonskiy, Leonid. Method and apparatus for standoff detection of liveness, patent US7417727, issued Aug 26, 2008.
- [33] Pollina, D., Dollins, A., Senter, S., Brown, T., Pavlidis, I., Levine, J., Ryan, A. "Facial Skin Surface Temperature Changes During a ‘Concealed Information’ Test." Annals of Biomedical Engineering. vol. 34, no. 7, pp. 1182-1189, 2006.
- [34] Peskosky, K.P. "Design of a Monocular Multi-Spectral Skin Detection, Melanin Estimation, and False-Alarm Suppression System." Wright-Patterson Air Force Base, Ohio. Thesis. 2010.

- [35] Sharma, A., Horaud, R., Knossow, D., Lavante, E., “Mesh Segmentation Using Laplacian Eigenvectors and Gaussian Mixtures”, AAAI Fall Symposium on Manifold Learning and its Applications. pp 50-56, 2009
- [36] Shlens, J. “A Tutorial on Principal Component Analysis.” Systems Neurobiology Laboratory, Salk Institute for Biological Studies and Institute for Nonlinear Science, University of California. Dec. 2005.
- [37] Stevenson, B, et. All. “The civil air patrol archer hyperspectral sensor system,” S. H. Wyatt, Ed., vol. 5787, no. 1. SPIE, pp. 17–28, 2005
- [38] Vezhnevets, V., Sazonov, V., and Andreeva, A. “A survey on pixel-based skin color detection techniques,” 2003.
- [39] Wendelken, S. M., McGrath, S.P., and Blike, G.T. “A medical assessment algorithm for automated remote triage.” In 25th Annual Engineering in Medicine and Biology Conference, IEEE. pp. 123–135. 2003
- [40] Widrow, B., Stearns, S.D. Adaptive Signal Processing. Upper Saddle River, N.J: Prentice-Hall, 1985.
- [41] Yazdi, M. and Gheysari, K. “A New Approach for the Fingerprint Classification Based on Gray-Level Co-Occurrence Matrix.” World Academy of Science, Engineering and Technology 47, 2008

Curriculum Vitae

Balvinder Kaur was born in New Delhi, India, and immigrated to the USA with her parents and siblings as a teenager. Although her immigration to another country was challenging, a stable home environment encouraged her curiosity and further education. She was strong-willed to be the first female engineer in her family and earned her Bachelor's degree in Electrical Engineering in 1999 from GMU. She worked as a professional electrical engineer in the US defense industry and then as a government employee for the US Army Night Vision Electronics Sensors Directorate. After a decade of completion of her Bachelor's degree, she returned to school to learn more in the areas of signal/image processing for static and dynamic target detection, recognition, classification, and tracking and to earn her Master's degree in Electrical Engineering. She wishes to continue working as a research scientist for the US Army and contribute in saving lives.