AN IMAGE BASED SYSTEM TO OBJECTIVELY SCORE THE DEGREE OF REDNESS IN PSORIASIS LESIONS

By

WAN MAISARAH BINTI WAN MUDA

FINAL PROJECT REPORT

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfillment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

AN IMAGE BASED SYSTEM TO OBJECTIVELY SCORE THE DEGREE OF REDNESS IN PSORIASIS LESIONS

by

Wan Maisarah binti Wan Muda

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

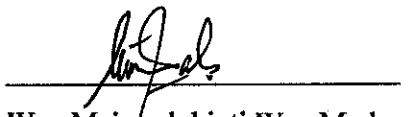
Ms. Zazilah May
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

June 2006

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.



Wan Maisarah binti Wan Muda

ABSTRACT

Nowadays, many skin diseases exist, ranging from harmless such as benign tumors to highly cancerous ones such as malignant melanoma. The visual resemblance of skin lesions requires experienced dermatologists for diagnosis and treatment of skin diseases. One of the most common types of the skin diseases is psoriasis which is chronic inflammatory skin condition, characterized by localized, widespread well-demarcated red plaques often topped by silvery scales. The basic characteristics of psoriasis lesions namely redness, thickness, and scaliness provide a mean of assessing the severity of psoriasis. Dermatologists are using Psoriasis Area and Severity Index (PASI) score, which takes into account signs such as redness, plaque thickness and scaling in order to assess psoriasis disease severity. The objective of this project is to generate the score of the redness and score of the area covered by psoriasis in order to build automated imaging system capable of classifying the severity of the disease. This system would assist dermatologists to give the suitable treatment to the different levels of psoriasis severity based on the PASI score. The psoriasis lesion images will be analyzed to classify the severity based on color, shape, size, and other features by using the Digital Image Processing Tools in MATLAB7 software. The entire information obtained through the computer vision and image processing as well as MATLAB7 software is applied towards the development of this project. The project will be implemented in two stages. The first stage (semester 1) involves literature review, research, data gathering, learning and training of the software or program and the second stage (semester 2) is analysis of core features, design, testing and analysis of results.

ACKNOWLEDGEMENTS

Firstly, grateful to Allah S.W.T. for the completion of this Final Year Project (FYP), An Image Based System to Objectively Score the Degree of Redness in Psoriasis Lesions, which focus on generating the score for redness severity and the score for area covered by the lesions in order to classify the severity of psoriasis lesions. Thankfully to the FYP Committee especially to Ms. Azrina Abdul Aziz and Miss Nasreen Badruddin for their continuous guidance by organizing the FYP Lecture Series.

The deepest gratitude to the project supervisor, Ms. Zazilah May for giving this opportunity to work under her guidance and support. Special thanks to the FYP technician, Miss Siti Hawa Mohd Tahir for her assistance in helping the author completed the tasks related and also for her supports and commitments.

The author would also like to thank the faculty for giving her the opportunity to participate in Engineering Design Exhibition 2006 (EDX 17), which give her a chance to improve presentation and interpersonal skills thus gain more experiences that might be useful for her future undertakings.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

a*	Chromaticity layer where color falls along the red-green axis
A	Area of the real body image
A_h	Area of involvement score on the head
A_l	Area of involvement score on the lower extremities
A_t	Area of involvement score on the trunk
A_u	Area of involvement score on the upper extremities
b*	Chromaticity layer where color falls along the blue-yellow axis
b	The range of intensity
B	Blue
Br	Brightness
BSA	Body Surface Area
BSAh	Body Surface Area on the head
BSAl	Body Surface Area on the lower extremities
BSAt	Body Surface Area on the trunk
BSAu	Body Surface Area on the upper extremities
CAD	Computer-Aided Diagnosis
CIE	Commission Internationale de L'Eclairage
cm	Centimeter
cm^2	Centimeter squared
EDX	Engineering Design Exhibition
e.g	For the sake of an example (Latin *exempli gratia*)
ELM	Epiluminescence Microscopy
FYP	Final Year Project
G	Green
GUI	Graphical User Interfase
GUIDE	Graphical User Interface development environment
I	Intensity
i.e.	That is (Latin *id est*)
kg	Kilogram
L*	Lightness
M	Total pixels of the lesion
m^2	Meter squared
N	Size of the lesion

NPF	National Psoriasis Foundation
p	Ratio for the size of the lesion over the total pixels of the lesion
PASI	Psoriasis Area and Severity Index
R	Red
$\mathbf{R_h}$	Redness score on the head
$\mathbf{R_l}$	Redness score on the lower extremities
$\mathbf{R_t}$	Redness score on the trunk
$\mathbf{R_u}$	Redness score on the upper extremities
S	Scale
$\mathbf{S_h}$	Scaliness score on the head
$\mathbf{S_l}$	Scaliness score on the lower extremities
$\mathbf{S_t}$	Scaliness score on the trunk
$\mathbf{S_u}$	Scaliness score on the upper extremities
T	Threshold value
TDS	Total Dermatoscopic Score
$\mathbf{T_h}$	Thickness score on the head
$\mathbf{T_l}$	Thickness score on the lower extremities
$\mathbf{T_t}$	Thickness score on the trunk
$\mathbf{T_u}$	Thickness score on the upper extremities
V	Redness value
Y	Luminance

CHAPTER 1
INTRODUCTION

1.1 Background of Study

Skin is the largest organ of our bodies that provides the outer covering and protects us from heat, light, injury, and infection. There are a number of conditions that can change the look and the condition of the skin. Some of the common skin diseases are Acne, Eczema, Psoriasis, Warts, Seborrhoeic Dermatitis, and Tinea [1]. Medical imaging has become a very important technology in medical diagnosis and treatment of skin diseases. Most medical imaging modalities generate digital images, which can be easily manipulated by computers.

Psoriasis is a type of skin disease which can be diagnosed using medical imaging. The basic characteristics of psoriasis lesions are redness, thickness, and scaliness, which provide the means of assessing the severity of psoriasis. The current gold standard for assessment of extensive psoriasis has been the Psoriasis Area and Severity Index (PASI). The PASI is a scoring procedure that is often used to evaluate psoriasis clinically and to measure outcomes in clinical trials. It scores the severity of lesions in terms of redness, thickness, and scaliness, and the score is weighted according to the area affected [12].

Digital Image Processing Tools in MATLAB software will be used to obtain the intended result. It will be used in generating an average score in order to classify the severity of the redness where the severity is rated for each index on a 0 – 4 scale. The Digital Image Processing also used in calibrating the affected area by the determination of the number of pixels in the digital image of the skin. This calibration is needed to generate a percentage of skin covered with psoriasis for each area.

1.2 Problem Statement

1.2.1 Problem Identification

Dermatologists have based the diagnosis of skin lesions on visual assessment of pathological skin and evaluation of macroscopic features. This process is highly dependent on the experience and visual acuity of the dermatologists. However, human vision lacks accuracy, reproducibility, and quantification in gathering information from an image. Besides, the number of dermatologists is not sufficient, especially in remote areas.

Therefore, it is important to equip clinics with computer-aided diagnosis (CAD) system to support early diagnosis of dangerous skin diseases. Recent advances in Color Image Processing and machine learning techniques have made it possible to build a content-based skin image retrieval system that is able to classify the types of skin lesions and to retrieve similar images of a patient's skin lesions from skin image database. Several groups have worked on a number of noninvasive methods to detect abnormal cells.

For psoriasis, there is no cure and the effective medicines exist only to control the symptoms. There is no blood test for psoriasis. Physicians usually diagnose it by examining the infected skin. Less often, a small piece of skin infected by the psoriasis is cut out and examined under a microscope. The psoriasis disease comes in many different forms and is categorized by doctors as mild, moderate, or severe, depending on the level of redness, thickness, scaliness, and percentage of body surface area involved. The severity of the disease is also commonly measured in clinical trials by using an objective scaling system called Psoriasis Area and Severity Index (PASI), which takes into account signs such as redness, plaque thickness and scaling.

PASI is used by dermatologists to assess psoriasis disease severity. Clinical trials of psoriasis present a greater challenge for measurement of psoriasis disease severity. Objective measures are needed that are reliable, valid, and consistent from investigator to investigator. Fortunately, psoriasis lesions are quite visible and therefore relatively easy to quantify; unfortunately, simple quantitation of the lesions

is not a complete assessment of severity, as the impact of the lesions is experienced differently by different patients.

For this Medical Imaging of Skin Diseases project, the aim is to build an automated imaging system capable of classifying the type of skin disease whether it is serious or benign disease. More specifically, this project aims in building a computer-aided diagnosis (CAD) system for a quick identification and diagnosis of psoriasis disease with the reliable, valid, and higher accuracy information in order to measure the overall psoriasis severity and coverage for the better treatments.

1.2.2 Significance of the Project

Most skin diseases, even as fatal as malignant melanoma such as psoriasis, can be cured based on physical features and color information if they are recognized early enough. With the current situation of skin diseases in Malaysia, an automated skin lesions classification and retrieval system based on 3-D images would be of great significance to the diagnosis process. This system would help dermatologists to provide quick identification and diagnosis of skin diseases.

The automated imaging system also can be applied for the Psoriasis Area and Severity Index (PASI) which is the most widely used tool to assess psoriasis disease severity in clinical trials, although it can be exceedingly cumbersome for use in daily clinical practice. Typically, the PASI would be calculated before, during, and after a treatment period in order to determine how well psoriasis responds to the treatment under test (a lower PASI means less psoriasis, generally). Because clinical trials rely on the PASI for inclusion criteria, having an automated system of PASI score on a clinic patient may be useful for determining the level of disease severity quickly and accurately for the better cure and treatments.

1.3 Objectives

1. To generate the score for redness level of psoriasis lesion and area covered by psoriasis lesion to be used in Psoriasis Area and Severity Index (PASI) score.
2. To build an automated imaging system capable of classifying the level of severity of skin diseases and retrieving the similar images stored in a skin image database.
3. To provide computer-aided diagnosis (CAD) system for a quick identification and treatment of skin diseases.

1.4 Scope of Study

The concepts and digital techniques for processing and analyzing medical images after they have been generated or digitized are organized into three sections that correspond to the fundamental classes of algorithms: enhancement, segmentation and quantification.

Enhancement algorithms are used to reduce image noise and increase the contrast of structures of interest. In many cases, enhancement improves the quality of the image and facilities diagnosis. Enhancement techniques are generally used to provide a clearer image for a human observer, but they can also form a preprocessing step for subsequent automated analysis.

Segmentation is the stage where a significant commitment is made during automated analysis by delineating structures of interest and discriminating them from background tissue. This separation, which is generally effortless and swift for the human visual system, can become a considerable challenge in algorithm development. In many cases the segmentation approach dictates the outcome of the entire analysis, since measurements and other processing steps are based on segmented regions.

Quantification algorithms are applied to segmented structures to extract the essential diagnostic information such as color value and the area of the lesions. Computerized analysis offers the exciting option of escaping from the anthropocentric description of images, and go beyond the limitations of the human visual and cognitive system.

Since this project is directly using Digital Image Processing, it incorporated the usage of computer vision and MATLAB programming language. The entire information obtained through the computer vision and image processing as well as MATLAB software is applied towards the development of this project.

CHAPTER 2
LITERATURE REVIEW

2.1 Common Components of Computer-Aided Diagnosis (CAD) Systems for Skin Diseases

2.1.1 Image Acquisition

To acquire the image of skin lesion:

- Epiluminescence microscopy (ELM) – allows light to penetrate deeper into the skin and make subsurface structures visible [5].
- Raman Spectroscopy – obtains Raman spectra by pointing a laser beam at a sample to excite molecules in the sample and a scattering effect is observed [4].
- Oblique-incidence diffuse reflectance spectroscopic imaging – obtains the information of the top layers of the skin tissues [6].

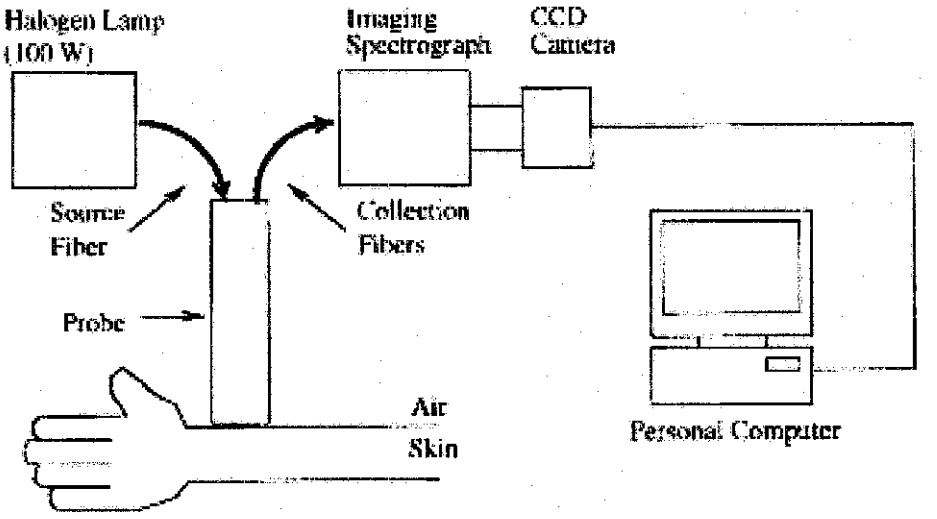


Figure 1 The Diffuse Reflectance Spectroscopic Imaging (DRSI) system

2.1.2 Image Segmentation

To identify the pigmented lesions from the image:

- The most often used technique is thresholding operation, which is based upon a simple concept. A parameter θ called the brightness threshold is chosen and applied to the image a[m,n] as follows [2]:

$$\begin{aligned} &\text{If } a[m,n] \geq \theta && a[m,n] = object = 1 \\ &\text{Else} && a[m,n] = background = 0 \end{aligned}$$

- This version of the algorithm assumes that we are interested in light objects on a dark background. For dark objects on a light background we would use [2]:

$$\begin{aligned} &\text{If } a[m,n] < \theta && a[m,n] = object = 1 \\ &\text{Else} && a[m,n] = background = 0 \end{aligned}$$

- In the skin lesion segmentation system, basically three different algorithms are used to segment a lesion, namely global thresholding, dynamic thresholding, and a 3-D color clustering concept. Best results were achieved with the following color channels; thresholding with the blue channel of the RGB color model and b channel of the CIE(L*a*b*) color model, 3-D color clustering with the X, Y, and Z channels of the CIE XYZ color model [5].

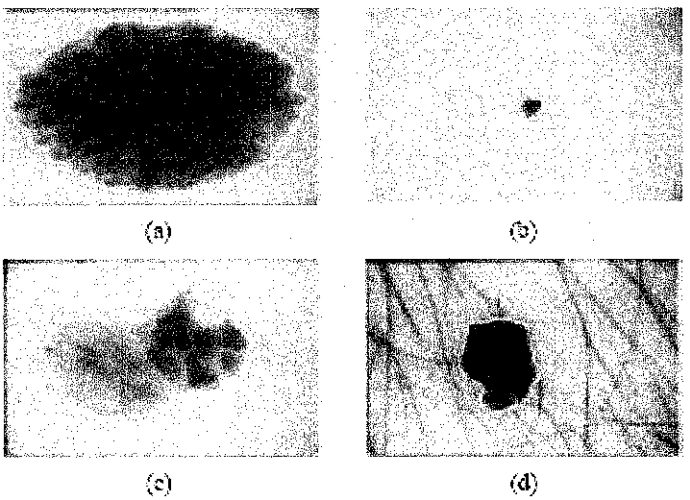


Figure 2 Segmentation results using thresholding operation

2.1.3 Features Extraction

To classify a skin lesion as malignant melanoma or benign tumor

- ABCD rule – refers to Asymmetry, Border structure, variegated Color and Diameter where a weighted combination of different scores provides total dermatoscopic score (TDS) to classify the lesions [5].
- ABDD rule – defines eight standard ELM criteria to use for diagnosis: pigment network, brown globules, black dots, radial streaming, psudopods, overall pigmentation and depigmentation [5].

2.1.4 Features Selection

To select features that have enough discriminative power to distinguish classes of skin lesions and are insensitive to irrelevant variability in the input features:

- Use Principal Component Analysis to reduce the dimensions of the inputs by discarding the input dimensions with low variance [4].
- Applied node pruning on neural network to select features and therefore the input units [4].
- Perform a feature selection by application of the sequential forward selection algorithm [5].

2.1.5 Image Classification and Retrieval

To classify skin lesions:

- B^+ and B-Tree – hierarchical structure with nodes containing pointers and keys and efficient for 1-D feature vectors [8].
- Clustering – similar items are grouped together into clusters [8].
- k-d tree – an extension of binary tree: k-dimensional vector is in place of single valued number, which allows for efficient searching in multidimensional feature space [8].
- Grid files – extension of fixed-grid structure in which n-dimensional space is divided into equal-size hypercubes, each containing feature vectors [8].

2.2 Psoriasis

2.2.1 Psoriasis

Psoriasis is an inflammatory skin disease in which skin cells replicate at an extremely rapid rate. New skin cells are produced about eight times faster than normal – over several days instead of a month – but the rate at which old cells slough off is unchanged. This causes cells to build up on the skin's surface, forming thick patches, or plaques, of red sores (lesions) covered with flaky, silvery-white dead skin cells (scales) [9].

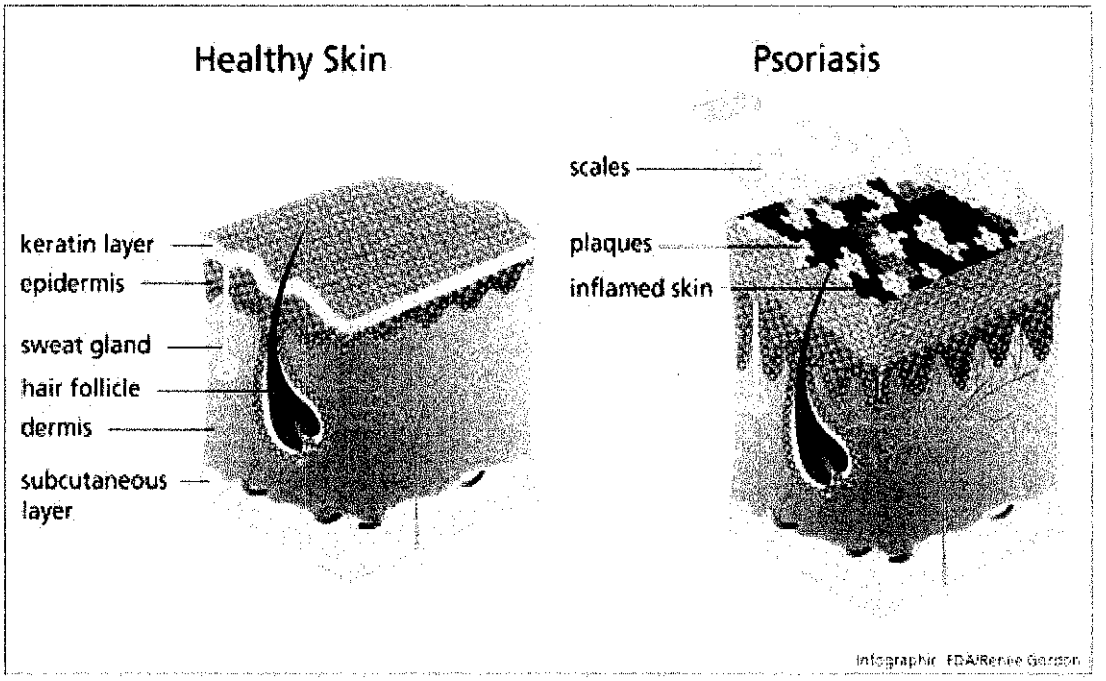


Figure 3 The difference between normal skin and skin covered by psoriasis

In psoriasis, an activated immune system triggers the skin to reproduce every three to four days, building up on the outer layers (epidermis and keratin). The epidermis thickens, blood flow increases and reddens the skin, and silver-gray scales cover it. Rarely life-threatening, at its mildest, psoriasis can be itchy and sore. At its worst, it's painful, disfiguring, and debilitating. The National Psoriasis Foundation (NPF) says that about two-thirds of the people with psoriasis have a mild form of the disease and one-third has moderate or severe psoriasis [9].

2.2.2 Type of Psoriasis

There are several distinct forms of psoriasis, which not only have different appearances, but also may require different types of treatments [10].

Plaque psoriasis (75% – 80% cases) – This is the most common form of psoriasis. It is characterized by raised, thickened patches of red skin covered with silvery-white scales. The most commonly affected areas include the knees, elbows, scalp, behind the ears, sacrum, navel, between the buttocks and genitalia.

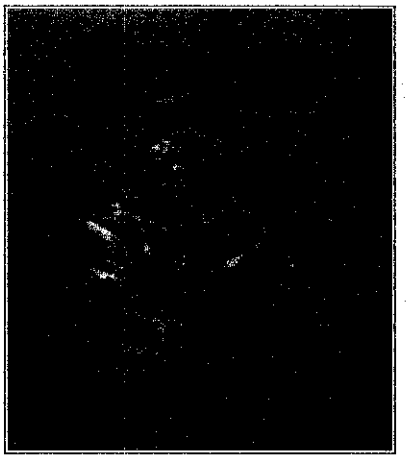


Figure 4 Plaque psoriasis on the elbow

Scalp psoriasis (50% cases) – This is plaque-type psoriasis. Due to the thickness of the psoriasis, it is very difficult to treat because topical formulations do not penetrate well.

Figure 5 Scalp psoriasis

Guttate psoriasis (15% – 20% cases) – This form of the disease most commonly occurs in children. It is characterized by droplet-shaped lesions that range in diameter and some very thin scales, which are found mainly on the trunk and may involve the face.

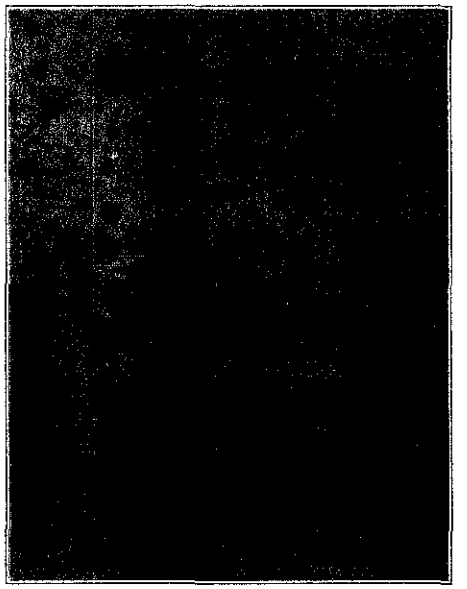


Figure 6 Guttate psoriasis on the back

Pustular psoriasis (2% cases) – This is a difficult-to-treat, less common form of psoriasis that generally occurs in older patients. It may present as new or as a flare up of plaque psoriasis and involves areas of reddened skin, particularly on the hands and soles of the feet.

Figure 7 Pustular psoriasis on sole of foot

Erythrodermic psoriasis (1% – 2% cases) – This is characterized by inflammatory lesions that may cause extreme reddening of all or most of the body. It generally occurs in people with chronic plaque psoriasis and has an average age of onset of 50 years.

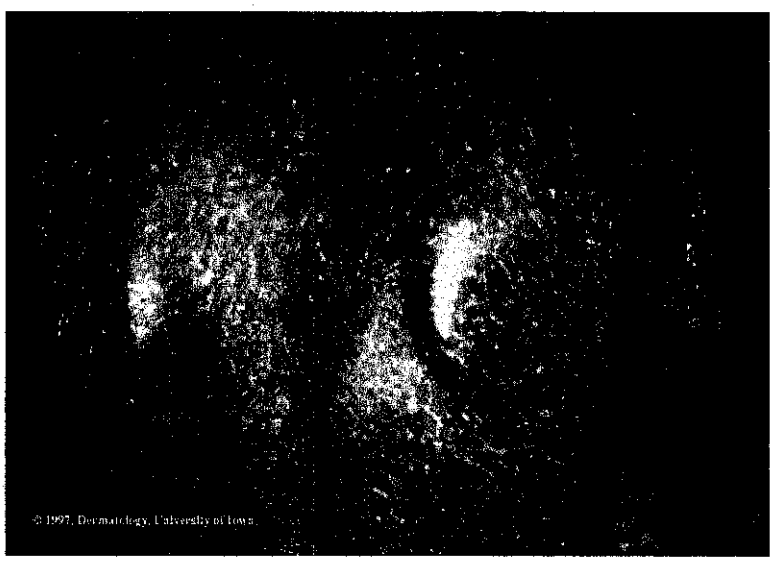


Figure 8 Erythrodermic psoriasis on the back

Inverse psoriasis – This form is typically found in folds or creases (i.e., armpits or groin, under pendulous breasts or in skin folds of obese patients). Its lesions are usually smooth and red, but do not have scaling.

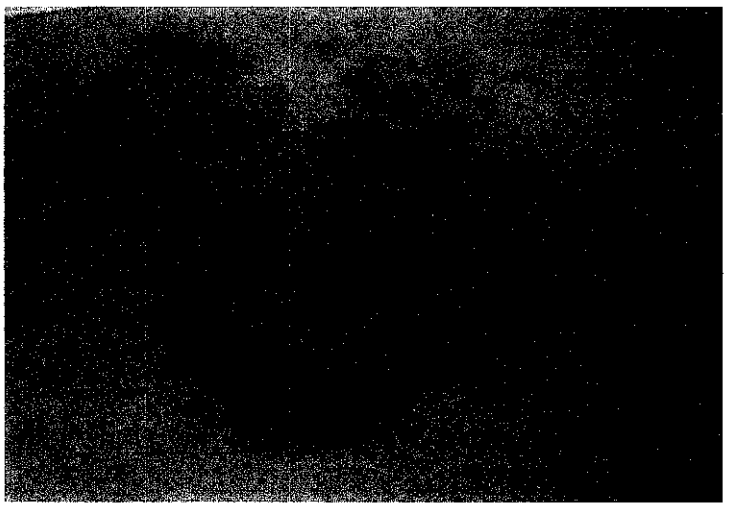


Figure 9 Inverse psoriasis

Psoriatic arthritis (23% cases) – This is a specific type of arthritis, which causes inflammation and swelling primarily in the hands, feet or in larger joints such as the knees, hips, elbows, and the spine. It may cause stiffness, pain, and joint damage. It is rare that a person can have psoriatic without having psoriasis.

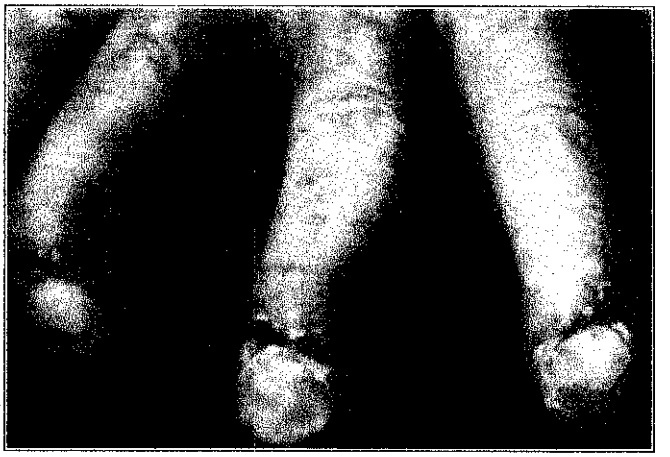


Figure 10 Psoriatic arthritis associated with skin and nail psoriasis

Nail psoriasis (50% cases) – This is characterized by large, deep, random pits of the nail plate. In one study, nail psoriasis was the first sign of disease in 4% of patients, but nail changes eventually occur in most patients.

Figure 11 Nail psoriasis

2.3 Psoriasis Area and Severity Index (PASI)

2.3.1 Psoriasis Area and Severity Index (PASI)

A patient's Psoriasis Area and Severity Index (PASI) is a measure of overall psoriasis severity and coverage. It is a commonly-used measure in clinical trials for psoriasis treatments. Typically, the PASI would be calculated before, during, and after a treatment period in order to determine how well psoriasis responds to the treatment under test (a lower PASI means less psoriasis, generally) [12].

The PASI measure the average redness, thickness, and scaliness of the lesions (each graded on a 0 – 4 scale), weighted by the area of involvement. The body is divided into four regions comprising the head, which covered 10% of a human skin surface, upper extremities (20%), trunk (30%), and lower extremities (40%). Each of these areas is scored by itself, and then the four scores are combined into the final PASI. In each of the regions, the fraction of total surface area affected is graded on a 0 – 6 scale (0 for no involvement; up to 6 for greater than 90% involvement). The various body regions are weighted to reflect their respective proportion of body surface area (BSA) [11].

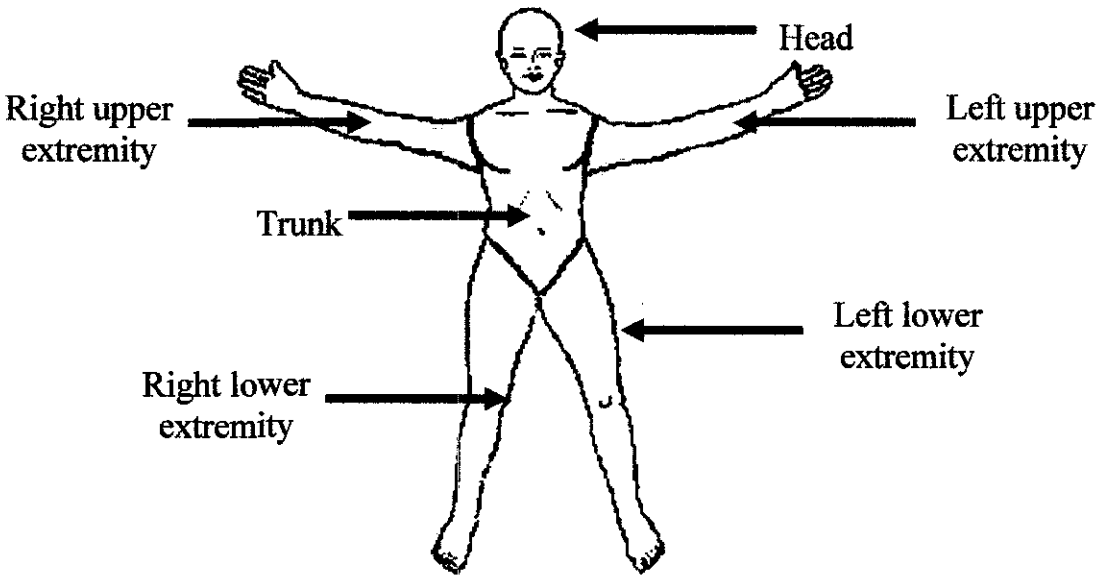


Figure 12 Four body regions with the difference percentage of human skin

This PASI will range from 0 (no psoriasis) to 72 (covered head-to-toe with complete redness, thickness and scaliness). PASI scores are nearly continuous, with 0.1 increments within these values. Since the initial use of the PASI, many have critiqued its utility. Although it has limitations, overall it has been a very practical measure of disease severity for clinical trials in patients with severe disease. In addition, many have attempted to revise the PASI and create new and improved methods of disease assessment [12].

2.3.2 The Steps in Generating PASI Score

The basic characteristics of psoriasis lesions are redness, thickness and scaliness which provide a means of assessing the severity of psoriasis. The steps in generating PASI score [11]:

a) Divide body into four areas

- Head (h) – consists 10% of a person's skin
- Arms or upper extremities (u) – consist 20% of a person's skin
- Trunk to groin (t) – consist 30% of a person's skin
- Legs to top of buttocks or lower extremities (l) – consist 40% of a person's skin

b) Generate an average score for the redness, thickness and scaliness

- 0 = no involvement
- 1 = slight and mild
- 2 = moderate
- 3 = severe
- 4 = most severe

Table 1 The description of severity in the difference score

Score	Redness	Thickness	Scaliness
0	No redness	No thickness	No scale
1	Slight pink	Feels firm	Slight scale
2	Pink	Raised	Scaly
3	Red	Thick	Flaky
4	Dark red	Very thick	Very flaky

c) Sum scores of redness, thickness and scaliness for each area

- Head = $R_h + T_h + S_h$
- Upper extremities = $R_u + T_u + S_u$
- Trunk = $R_t + T_t + S_t$
- Lower extremities = $R_l + T_l + S_l$

d) Generate a percentage for skin covered with psoriasis for each area (A_h, A_u, A_t, A_l) and convert that to a 0 – 6 scale

- 0 = 0% (clear)
- 1 = 0 – <10%
- 2 = 10 – <30%
- 3 = 30 – <50%
- 4 = 50 – <70%
- 5 = 70 – <90%
- 6 = 90 – 100%

e) Multiply the sum of the individual-severity scores for each region (part c) by the weighted area-of-involvement score for that respective region (part d) and multiply that by 0.1, 0.2, 0.3 and 0.4 for head, upper extremities, trunk and lower extremities, respectively

- Head = 0.1 ($R_h + T_h + S_h$) A_h
- Upper extremities = 0.2 ($R_u + T_u + S_u$) A_u
- Trunk = 0.3 ($R_t + T_t + S_t$) A_t
- Lower extremities = 0.4 ($R_l + T_l + S_l$) A_l

f) Add these scores to get the PASI score

PASI score =

$$0.1(R_h + T_h + S_h)A_h + 0.2(R_u + T_u + S_u)A_u + 0.3(R_t + T_t + S_t)A_t + 0.4(R_l + T_l + S_l)A_l$$

The highest potential PASI score is 72 and the lowest is 0:

Highest PASI score

= 0.1 (4 + 4 + 4) 6 + 0.2 (4 + 4 + 4) 6 + 0.3 (4 + 4 + 4) 6 + 0.4 (4 + 4 + 4) 6

= 0.1 (72) + 0.2 (72) + 0.3 (72) + 0.4 (72)

= 72

Lowest PASI score

= 0.1 (0) + 0.2 (0) + 0.3 (0) + 0.4 (0)

= 0

When conducting a clinical trial for treatment for psoriasis, a predetermined primary endpoint is required on which the efficacy of the drug will be assessed. In clinical trials, endpoints such as PASI 75 (a 75% reduction in disease activity) and PASI 50 (a 50% reduction in disease activity) are used. For patients with severe psoriasis, many clinicians consider at least a 75% improvement to be clinically meaningful in terms of treatment success.

2.4 The CIE L*a*b* Color Space

Color is a sensation created in response to excitation of our visual system by electromagnetic radiation known as light. A color can be specified by a tri-component vector. The set of all colors form a vector space called color space [13].

A few color terms:

- Intensity (I) – a measure, over some interval of the electromagnetic spectrum, of the flow of power that is radiated from, or incident on a surface.
- Brightness (Br) – the attribute of a visual sensation according to which an area appears to emit more or less light.
- Luminance (Y) – since brightness perception is very complex, the Commission Internationale de L'Eclairage (CIE) defined another quantity luminance (Y) which is radiant power weighted by a spectral sensitivity function that is characteristic of human vision.
- Lightness (L*) – human vision has a nonlinear perceptual response to luminance which is called lightness.

The CIE adopted standard color curves for a hypothetical standard observer. These color curves specify how a specific spectral power distribution (SPD) of an external stimulus (visible radiant light incident on the eye) can be transformed into a set of three numbers that specify the color. The CIE color specification system is based on the description of color as the luminance component Y and two additional components X and Z. The spectral weighting curves of X and Z have been standardized by the CIE based on statistics from experiments involving human observers. The CIE XYZ tristimulus values can be used to describe any color [13].

The xy diagram is very useful in presenting the relative positions of colors and can also be used to see what colors will result from mixing two others (for additive colors only, such as a computer screen, not for subtractive mixes like paint colors). However, this color space, as with XYZ, is not perceptually uniform, and a given

separation between two colors on the diagram corresponds to different perceived differences, depending on their relative positions [14].

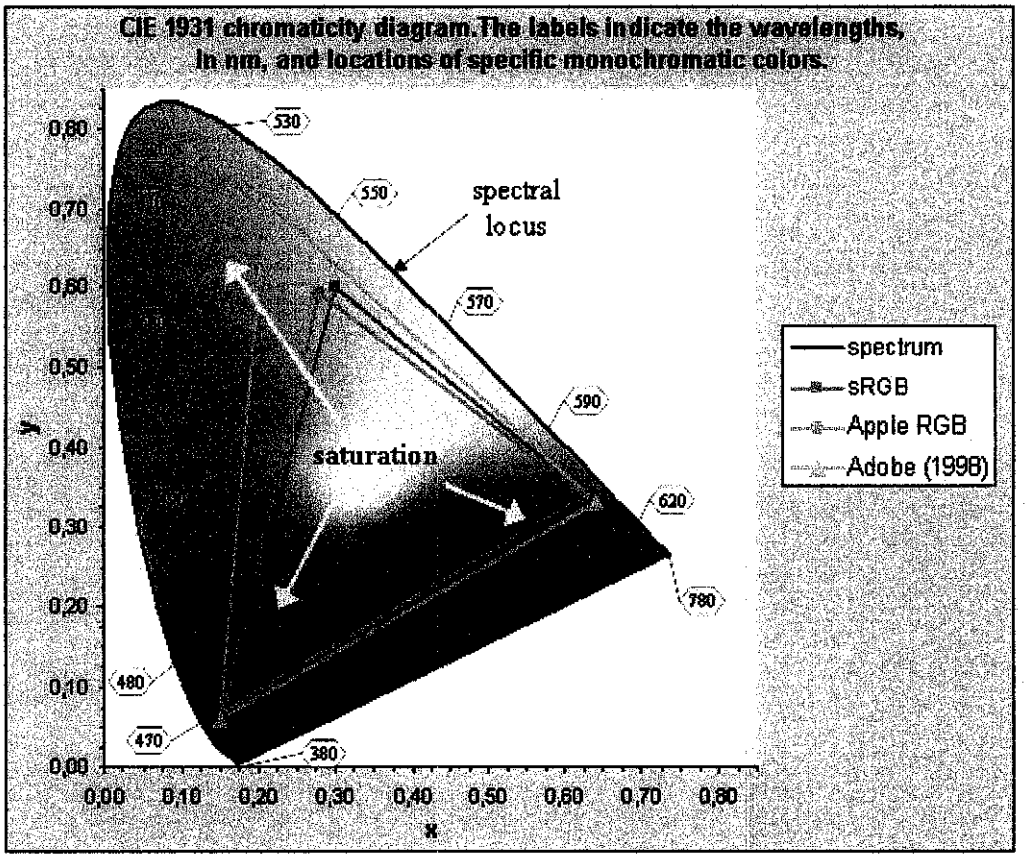


Figure 13 CIE 1931 chromaticity diagram

Traditionally color images have been specified by the non-linear red (R'), green (G') and blue (B') tristimulus values where color image storage, processing and analysis is done in this non-linear RGB (R'G'B') color space. The CIE have also derived and standardized two other color spaces, called L*u*v* and L*a*b*, from the CIE XYZ color space which are perceptually uniform. Starting with the XYZ data, there is an assortment of mathematical transformations that are available to obtain various alternate color notations: xyY, L*a*b*, L*u*v*, L*C*h*, and even RGB [14].

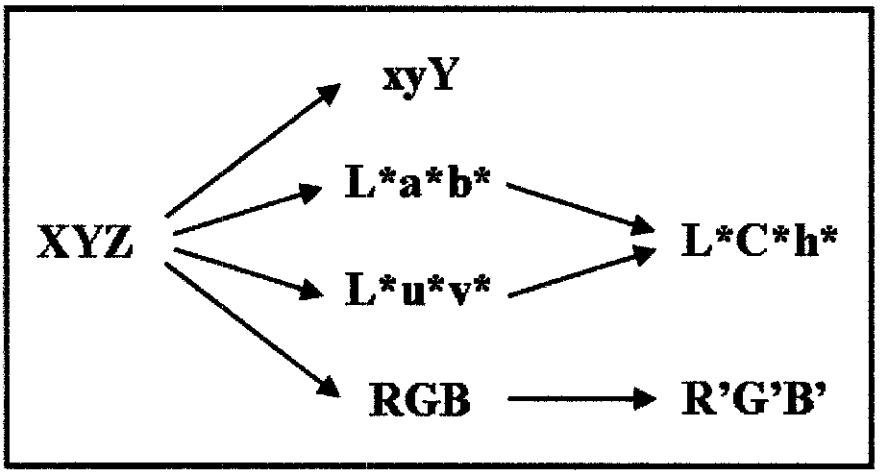


Figure 14 Color space transformations

The L*a*b* color space is the second uniform color space standardized by CIE. It is derived based on the CIE XYZ space and white reference point. The white reference point [Xn, Yn, Zn] is the linear RGB = [1, 1, 1] values converted to the XYZ values using the following transformation [13]:

$$\begin{pmatrix} Xn \\ Yn \\ Zn \end{pmatrix} = \begin{pmatrix} 0.4125 & 0.3576 & 0.1804 \\ 0.2127 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9502 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

In this space, the L* value is a measure of the lightness, while a* and b* define together the hue and saturation of the color. Specifically, the a* axis runs from red to green, and the b* axis from yellow to blue. The lightness L* component and the a* and b* components which are representative of chrominance are given by

$$L^* = 116\,(Y/Yn)^{1/3} - 16$$

$$a^* = 500\,[(X/Xn)^{1/3} - (Y/Yn)^{1/3}]$$

$$b^* = 200\,[(Y/Yn)^{1/3} - (Z/Zn)^{1/3}]$$

with the constant that X/Xn, Y/Yn, Z/Zn > 0.01. This constraint will be satisfied for most practical purposes. Hence, the modified formulae described in for cases that do not satisfy this constraint can be ignored in practice [13].

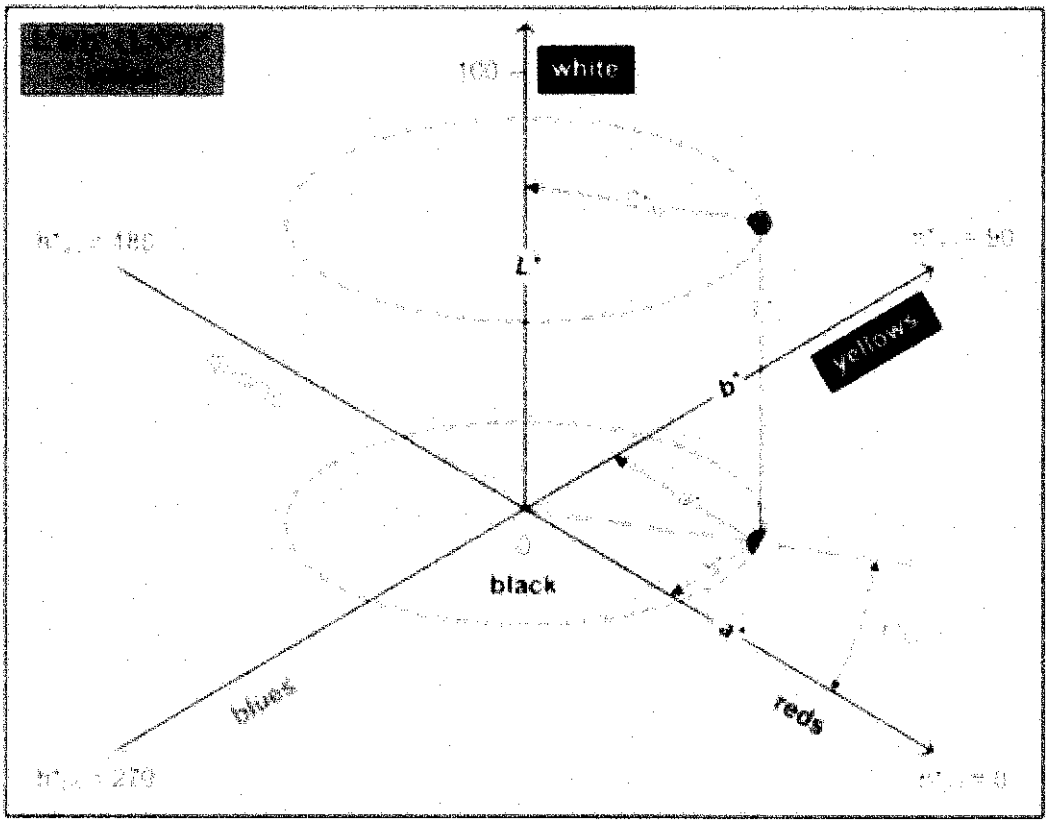


Figure 15 The L*a*b* color space coordinate

The L*a*b* spaces are very useful in applications where precise quantification of perceptual distance between two colors is necessary.

2.5 K-Means Clustering Using L*a*b* Color Space

Color images are usually represented and handled in RGB coordinates. In this format, an image F of size M x N may be represented as F = R(F);G(F);B(F), i.e., as a set of three MxN matrices respectively containing the red, green, and blue components of F. The RGB color space allows for a very straightforward representation of colors but, unfortunately, it has a Riemannian nature; this means that it is not a uniform space and perceived differences among colors can be assessed only by means of complicated metrics.

Therefore, in order to have at disposal a simple measure for evaluating perceptive distances, the uniform L*a*b* color space has been chosen, which is an orthogonal Cartesian coordinate system endowed with the simple Euclidean metric. It was verified that the CIE(L*a*b*) space is perceptually uniform and gives better results for segmentation of color images [15].

Clustering is a way to separate groups of objects. K-means clustering treats each object as having a location in space. It finds partitions such that objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. K-means clustering requires a specified number of clusters to be partitioned and a distance metric to quantify how close two objects are to each other. K-means divide a collection of objects into K groups. The algorithm iterates over two steps:

- Compute the mean of each cluster.
- Compute the distance of each point from each cluster by computing its distance from the corresponding cluster mean. Assign each point to the cluster it is nearest to.

Iterate over the above two steps till the sum of squared within group errors cannot be lowered any more. The initial assignment of points to clusters can be done randomly. In the course of the iterations, the algorithm tries to minimize the sum, over all groups, of the squared within group errors, which are the distances of the points to the respective group means. Convergence is reached when the objective function (i.e., the

residual sum-of-squares) cannot be lowered any more. The groups obtained are such that they are geometrically as compact as possible around their respective means [16].

The k-means algorithm for partitioning (or clustering) N data points into K disjoint subsets S_j containing N_j data points so as to minimize the sum-of-squares criterion

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} |x_n - \mu_j|^2,$$

where x_n is a vector representing the nth data point and μ_j is the geometric centroid of the data points in S_j. In general, the algorithm does not achieve a global minimum of J over the assignments. In fact, since the algorithm uses discrete assignment rather than a set of continuous parameters, the minimum it reaches cannot even be properly called a local minimum. Despite these limitations, the algorithm is used fairly frequently as a result of its ease of implementation.

The algorithm consists of a simple re-estimation procedure as follows. Initially, the data points are assigned at random to the K sets. For step 1, the centroid is computed for each set. In step 2, every point is assigned to the cluster whose centroid is closest to that point. These two steps are alternated until a stopping criterion is met, i.e., when there is no further change in the assignment of the data points [17].

2.6 Body Surface Area

In physiology and medicine, the Body Surface Area (BSA) is the measured or calculated surface of a human body. For many clinical purposes BSA is a better indicator of metabolic mass than body weight because it is less affected by abnormal adipose mass. Estimation of BSA is simpler than many measures of volume [18].

BSA measurement used in many medical tasks. Various BSA formulas have been developed over the years, originally by Dr.s Du Bois & Du Bois, followed by Gehan and George, Haycock, Boyd and Mosteller. These formulas all give slightly different results. It is probably not worth the trouble to debate about which formula may or may not be slightly better. A bigger issue is lack of standardization. The Mosteller formula is gaining support as a common standard because it is much simpler and can be memorized and easily calculated with a hand-held calculator [19].

$$\text{Mosteller BSA } (m^2) = \left[\frac{\text{Height (cm)} \times \text{Weight (kg)}}{3600} \right]^{0.5}$$

2.7 Graphical User Interface (GUI)

2.7.1 GUIDE

GUIDE, the MATLAB Graphical User Interface development environment, provides a set of tools for creating graphical user interfaces (GUIs). These tools greatly simplify the process of designing and building GUIs. The GUIDE tools can be used to [20]:

- Lay out the GUI – the GUI can be laid out easily by using the GUIDE Layout Editor. GUIDE stores the GUI layout in a FIG-file.
- Program the GUI – GUIDE automatically generates an M-file that controls how the GUI operates.

2.7.2 Laying Out a GUI

Start GUIDE by typing guide at the MATLAB command prompt. This displays the GUIDE Quick Start dialog. From the Quick Start dialog, a new GUI can be created from one of the GUIDE templates.

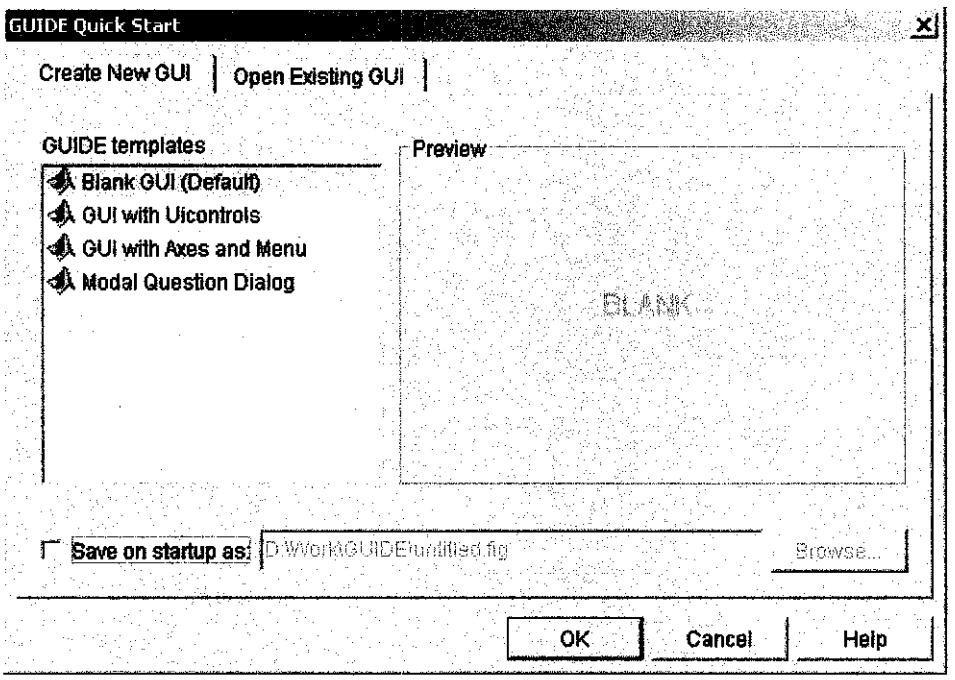


Figure 16 The GUIDE Quick Start dialog

The GUI will be displayed in the Layout editor, which is the control panel for all of the GUIDE tools. The GUI can be laid out by dragging components, such as push buttons, pop-up menus, or axes, from the component palette, at the left side of the Layout Editor, into the layout area [20].

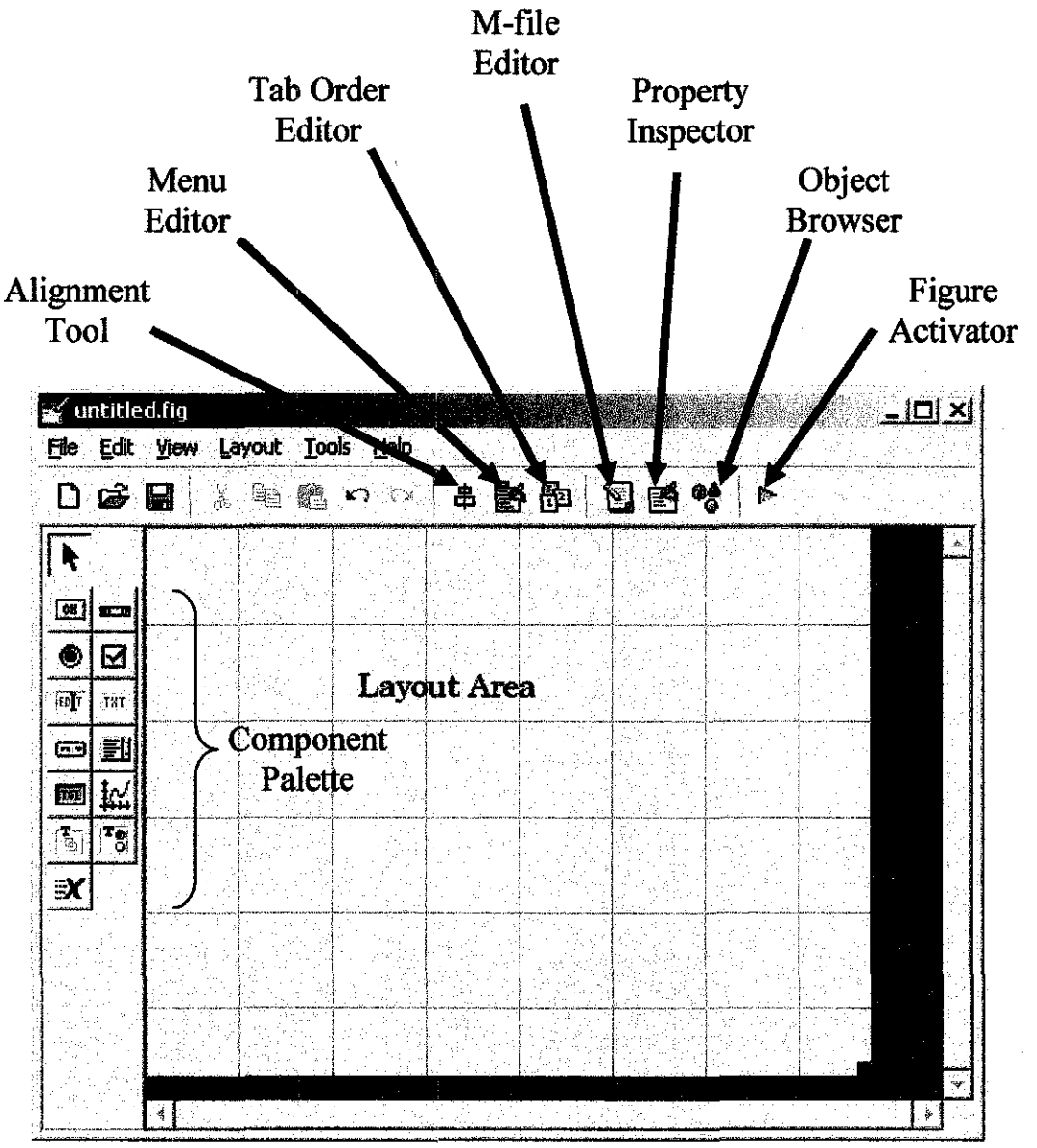


Figure 17 The Layout Editor with a blank GUI template

2.7.3 Programming a GUI

After laying out the GUI and setting component properties, the next step is to program the GUI using the M-file Editor. The GUI can be programmed by coding one or more callbacks for each of its components. Callbacks are functions that execute in response to some action by the user. A GUI's callbacks are found in the M-file that GUIDE generates automatically. GUIDE adds templates for the most commonly used callbacks to this M-file [20].

The GUI M-file:

- Initializes the GUI
- Contains code to perform tasks before the GUI appears on the screen, such as creating data or graphics
- Contains the callback functions that are executed each time a user clicks a GUI component

Initially, each callback contains just a function definition line. Using the M-file Editor, codes can be added to the callbacks to perform the related functions. The M-file can be opened by clicking the M-file Editor icon on the Layout Editor Toolbar [20].

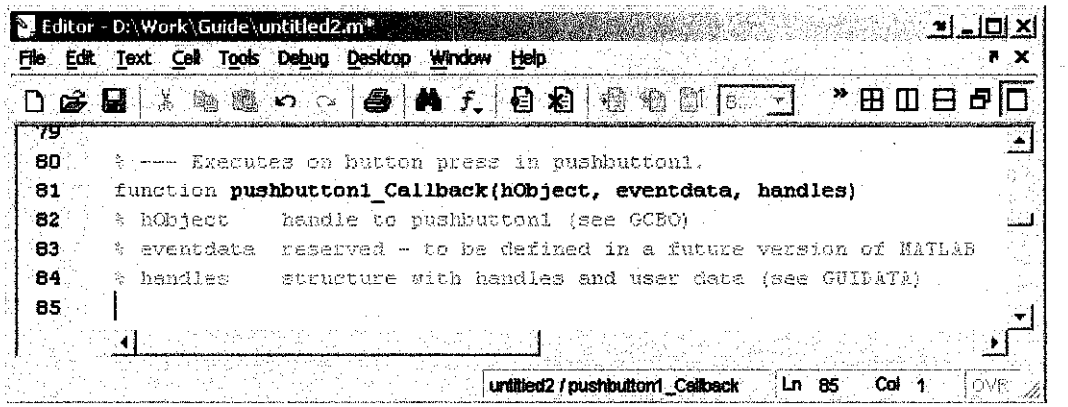


Figure 18 The M-file shows the Callback template for a push button

CHAPTER 3
METHODOLOGY

There are two main components in this section:

i. Project's Methodology
ii. Imaging Tasks

3.1 Project's Methodology

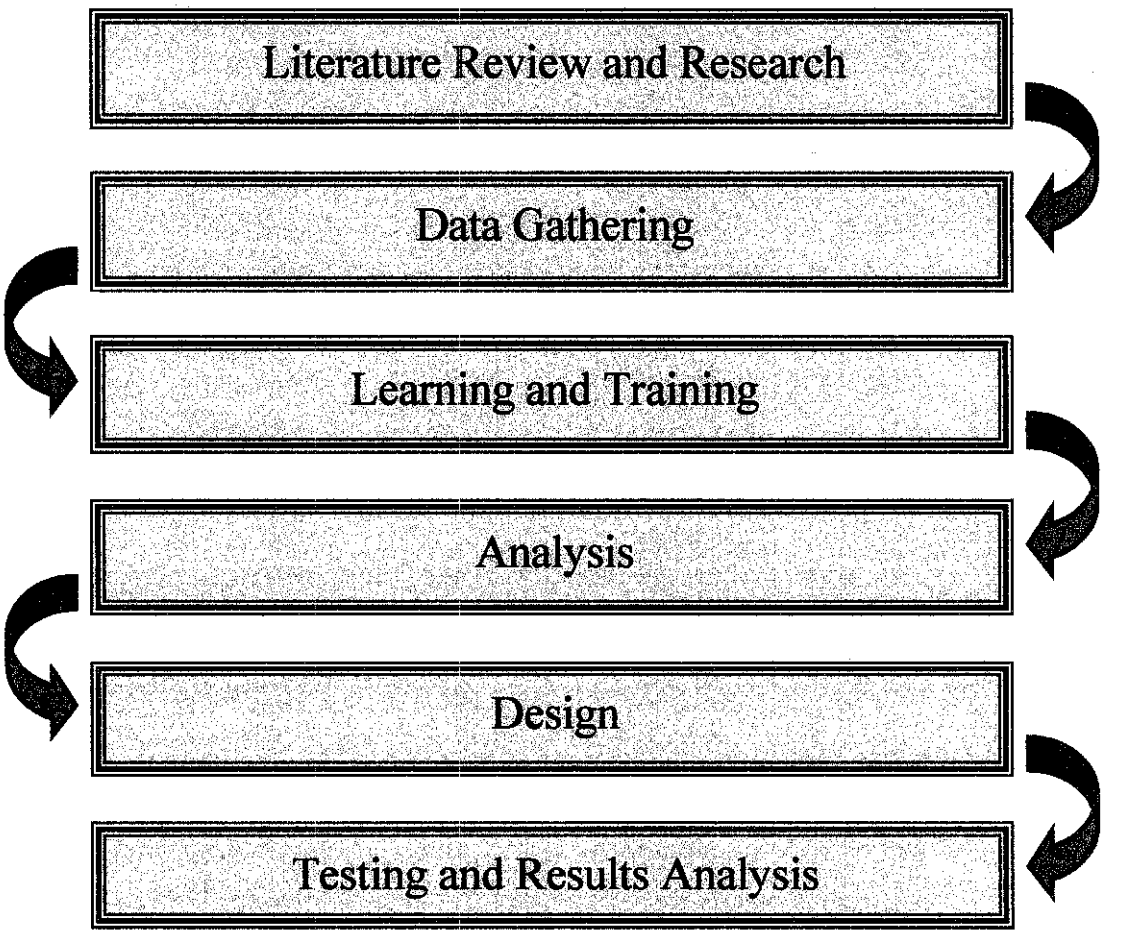


Figure 19 Flow chart of project methodology

This project which is implemented in two semesters needs several methods to be accomplished. Therefore, it is developed by stages as listed below:

Semester 1: Literature Review / Research / Data Gathering / Learning and Training

Semester 2: Analysis / Design / Testing / Analysis of Results

3.1.1 Literature Review and Research

Literature review and research will be done by seeking information trough books, internet and journals especially about the psoriasis topics and the current techniques which is related to skin diseases. This stage provides important and useful knowledge for design stage.

3.1.2 Data Gathering

All the relevant data information obtained from research for this project are gathered and revised. This stage involved a detailed study of the current system, leading to specifications of a new system.

3.1.3 Learning and Training

It is important to learn and train how to use the related software or program especially the Digital Image Processing Tools in MATLAB7 software. The average score in classifying the redness severity is determined by using the Color Image Processing from the Digital Image Processing Tools. Besides, this software is used in order to calculate the affected area to get the percentage of area covered by psoriasis.

3.1.4 Analysis

The analysis of core features is needed after the development of image database. System analysis includes sub-dividing of complex process involving the entire system, identification of data store and manual processes. The main point in this systems analysis stage is a specification of what the new system is to accomplish based on the user requirements.

3.1.5 Design

The new system is designed based on the user requirements and the detailed analysis of a new system. It is a most crucial phase in the development of a system. The detail process involved in the design stage will be discussed in the Imaging Tasks.

3.1.6 Testing and Results Analysis

Before actually implementing the new system into operations, a test run of the system is done to ensure the device function well. It is an important stage of a successful system. After codifying the whole programs of the system, a test plan should be developed and run on a set of test data. The output of the test run should match the expected results. For this project, a set of 200 psoriasis images have been used to test this image based system. The result shows that the accuracy of this system in classifying the redness severity level of psoriasis lesions is about 90%.

3.2 Imaging Tasks

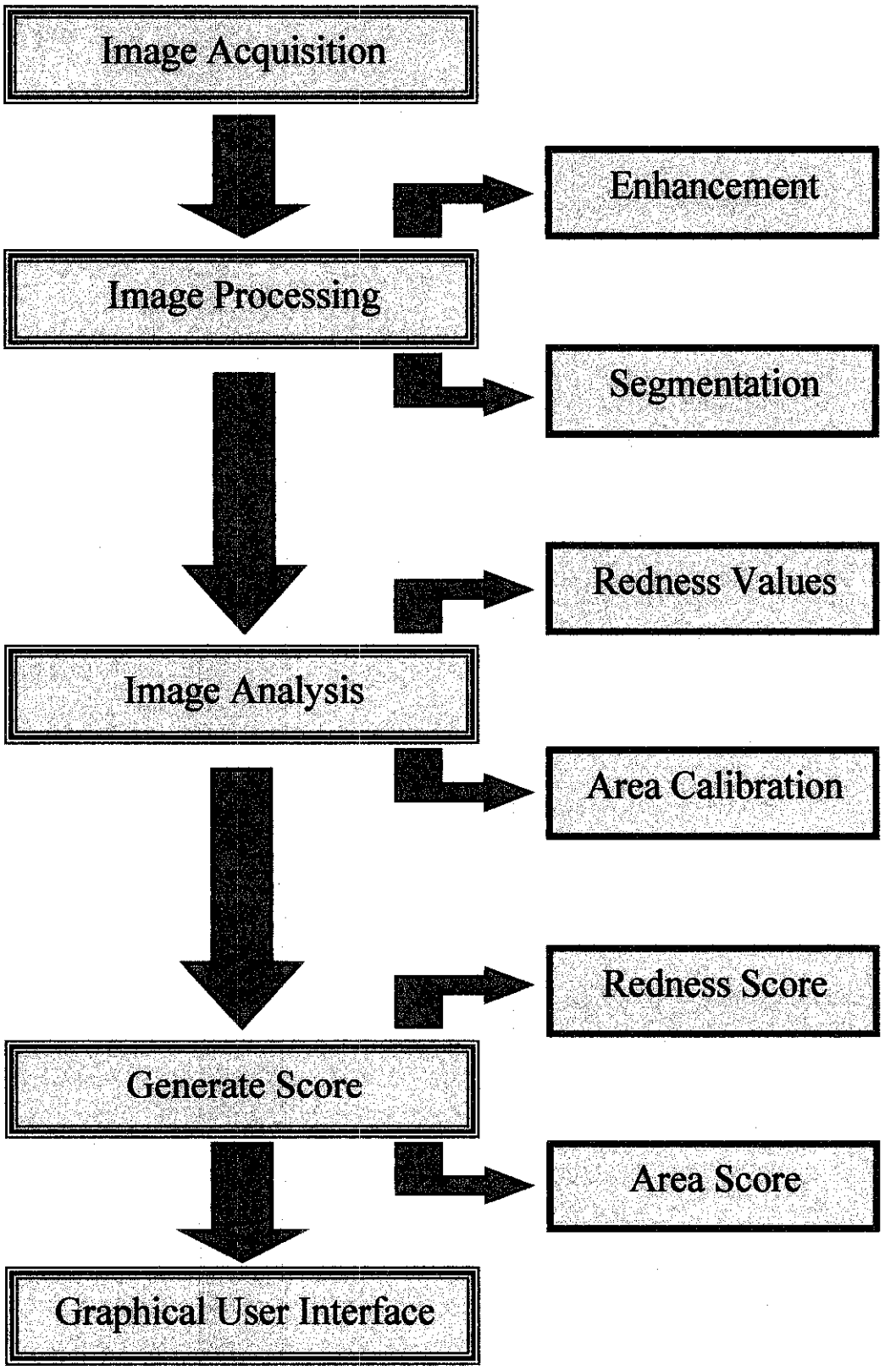


Figure 20 Flow chart of imaging tasks

3.2.1 Image Acquisition

The images of skin lesion are acquired using the specific methods and equipments as stated in the literature review. For this project, most of the images are obtained from psoriasis and dermatologist websites.

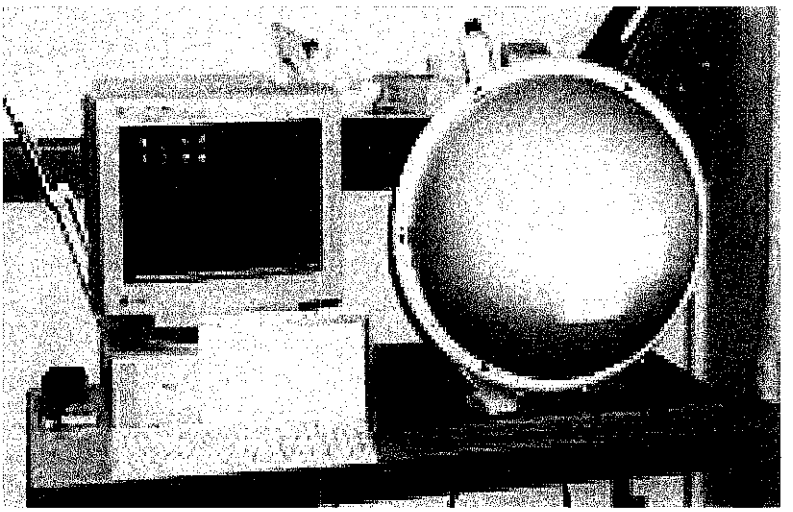


Figure 21 The equipments for image acquisition

The medical images need to be stored in the graphics file of the MATLAB software (work folder) first before it can be load and read.

Syntax: *uigetfile*

uigetfile is a function of a standard open file dialog box to load an image from the graphics file. It displays a dialog box for the user to fill in, and returns the filename (e.g. psoriasis) and path strings (e.g. jpg) and the index of the selected filter. A successful return occurs only if the file exists. If the user selects a file that does not exist, an error message is displayed, and control returns to the dialog box. The user may then enter another filename, or press the 'cancel' button.

Syntax: *imread*

imread is used to read a grayscale or color image which has been loaded from the graphics file.

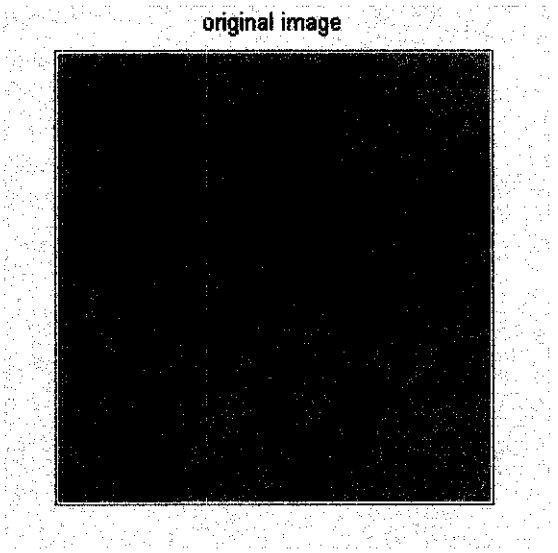


Figure 22 The original image of psoriasis lesion

3.2.2 Image Processing

3.2.2.1 Image Enhancement

Syntax: *imadjust*

imadjust adjust the image intensity values or colormap to new values such that 1% of data is saturated at low and high intensities of the original image. This increases the contrast of the output image. This enhancement is important for the better segmentation and also to obtain the greater values of the redness.

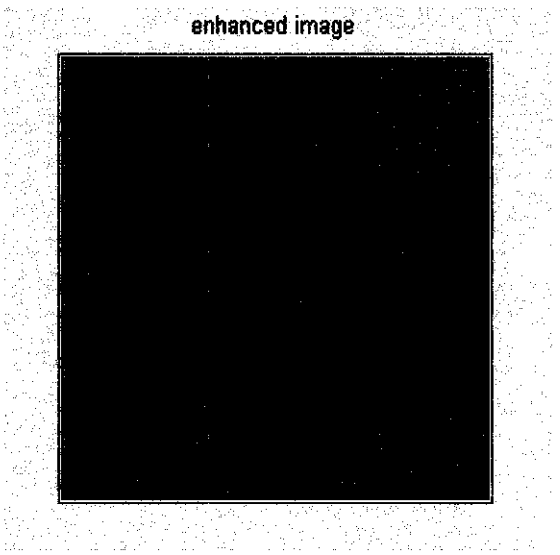


Figure 23 The enhanced image of psoriasis lesion

3.2.2.2 Image Segmentation

The enhanced image will be segmented into three clusters using the k-means clustering method. This number of cluster is enough to segment the lesion and the skin. The lesion can be obtained from one of the clusters after the segmentation. From the following figure, the lesions image is in the cluster 3 whereas the skin obtained in the cluster 2.

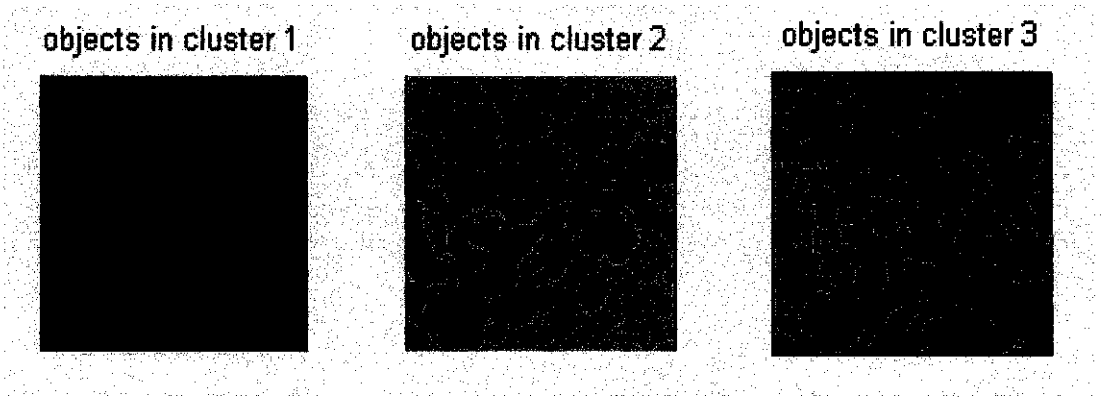


Figure 24 The different segmented images in the different clusters

3.2.3 Image Analysis

3.2.3.1 Redness Values

For image analysis, the segmented image or lesion image will be converted into grayscale image in order to obtain the data from the image.

Syntax: *rgb2gray*

rgb2gray converts the color image to the grayscale intensity image by eliminating the hue and saturation information while retaining the luminance.

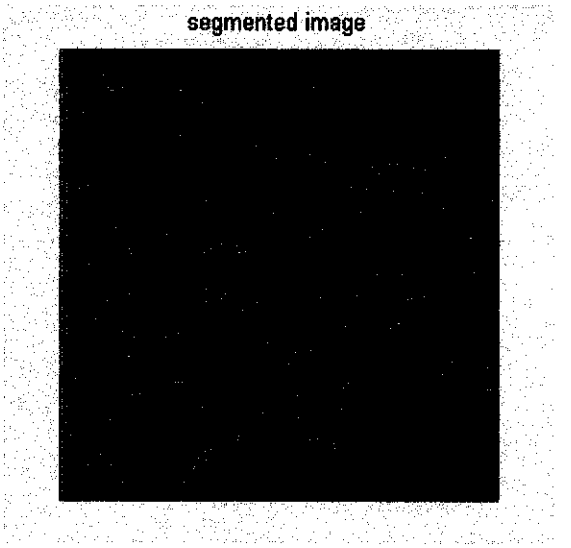


Figure 25 The lesion image obtained from segmentation

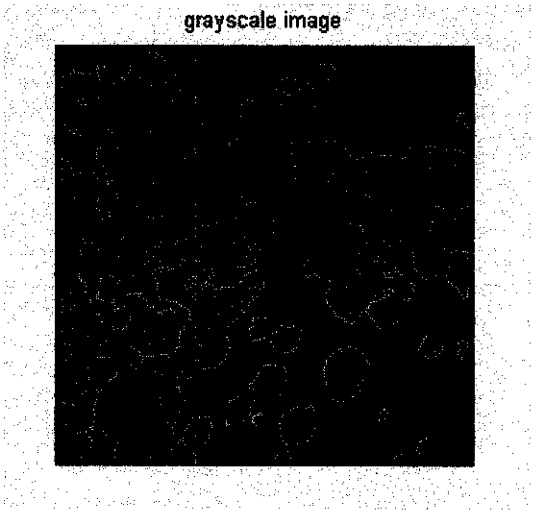


Figure 26 The grayscale image

The image data from the grayscale image can be displayed by a histogram.

Syntax: *imhist*

imhist displays a histogram for the intensity image whose number of bins is specified by the image type. The default value of bins for grayscale image is 256.

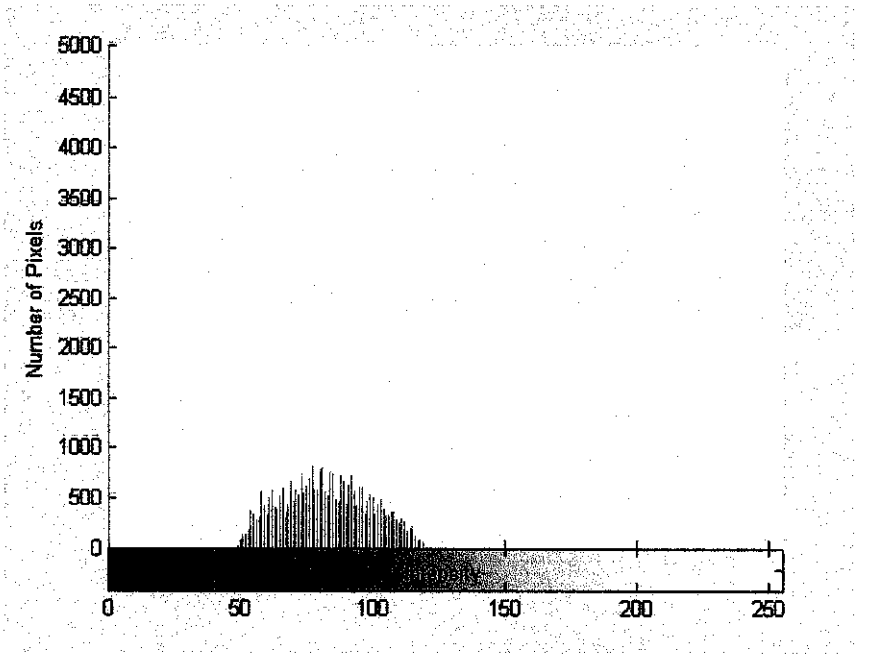


Figure 27 The histogram of the image data

From the histogram, the plot at x = 0 shows the background of the image while the other plots or bins represent the image lesions. To get the size of the lesion, N, the background will be excluded by obtaining only the values in the range of (x > 0).

[y x] = imhist(g);

N = y(x>0);

The next step is to convert the grayscale image into binary image in order to get the total pixels of the lesion.

Syntax: *bw = roicolor(g,low,high)*

This syntax returns a binary image, bw, with 0's outside the region of interest and 1's inside. The pixels of the selected region which is the lesion region lie within the range *[low,high]*.

The thresholding can be summarizes as below:

bw = (g >= low) & (g <= high) = 1

In this project, the selected region must be higher than 0, which is the intensity value for the background and lower or equals to 255. So, the selected value for high = 255 which is the highest intensity value whereas the value for low = 10 which is lower than the possible intensity value for lesion and near to the 0 (background value).

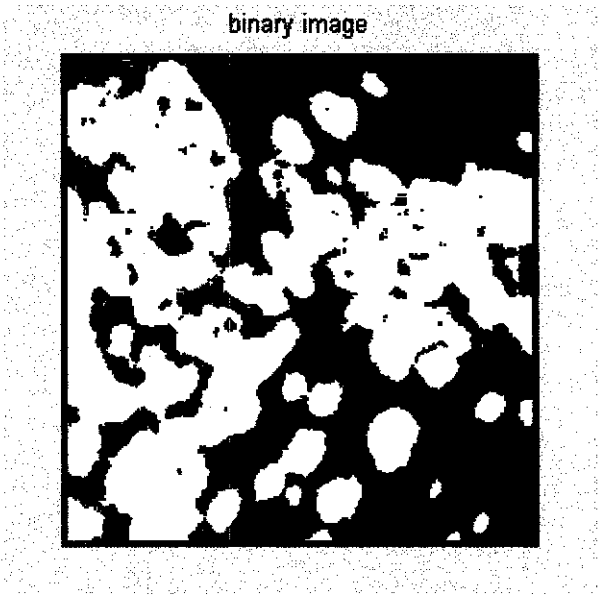


Figure 28 The binary image with 0's outside the region and 1's inside

The total pixels of the lesion can be determined from the binary image.

Syntax: *total = bwarea(bw)*

bwarea compute the area of objects in binary image. *total* is a scalar whose value corresponds roughly to the total number of 'on' pixels in the image, but may not be exactly the same because different patterns of pixels are weighted differently [20]. From this syntax, the total pixels of the lesion, M, can be obtained.

The mean intensity value of the lesion can be calculated using the obtained values; the size of the lesion, N, and the total pixels of the lesion, M.

Numerator = Size of the lesion (N)

Denominator = Total pixels of the lesion (M)

Set the range of the intensity:

b = (1:255)

p is corresponding to the image data:

$$\mathbf{p = [N / M]}$$

The mean intensity of the lesion:

$$\mathbf{Mean = b*p}$$

The mean intensity values of the different lesions will be used as the redness values in order to generate score for the severity level of psoriasis lesions.

3.2.3.2 Area Calibration

For the area calibration, the total pixels of the lesion are obtained from the binary image using the *bwarea* syntax.

Syntax: *total = bwarea(bw)*

The reference value or the calibrating factor is estimated by obtaining the relationship between a 1 cm line and the number of pixels from the binary image.

Syntax: *pixval on;*

This syntax turns on interactive display of information about image pixels in the current figure. It install a black bar at the bottom of the figure which displays the (x,y) coordinates for whatever pixel the cursor is currently over. If user clicks on the image and hold down the mouse button while moving the cursor, *pixval* also displays the Euclidean distance between the previous point and the current cursor location. It draws a line between these points to indicate the distance being measured.

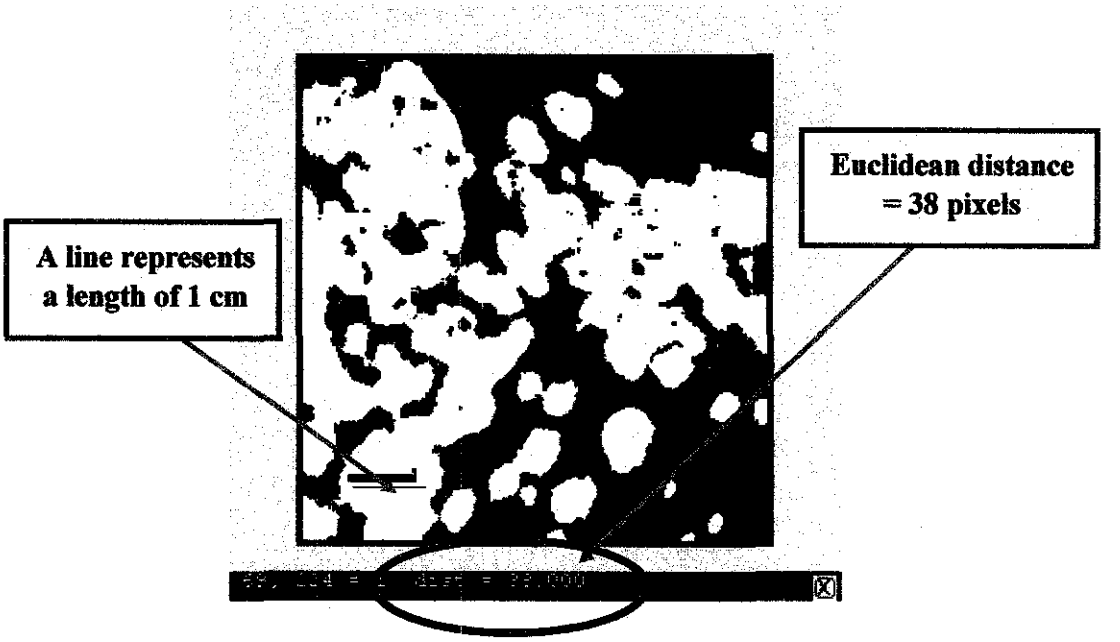


Figure 29 The Euclidean distance obtained from a binary image

From here, it is obtained that 1 cm = 38 pixels.

1 cm = 38 pixels

1 cm^2 = 1444 pixels

The total pixels for 1 cm^2 will be used as reference or calibrating factor to get the area of the lesion in cm^2. The area can be calculated by dividing the total pixels of the lesion image by the total pixels for 1 cm^2 image.

area (cm^2) = total / 1444

3.2.4 Generate Score

3.2.4.1 Generate Score for Redness

Five images from the different scores from 0 to 4 are used as the reference images to get the different values of redness. The redness values of the lesion for each image are obtained in the image analysis stage:

- V0 = redness value for lesion image with score 0
- V1 = redness value for lesion image with score 1
- V2 = redness value for lesion image with score 2
- V3 = redness value for lesion image with score 3
- V4 = redness value for lesion image with score 4

The threshold values are calculated by determining the average values between the higher and the lower score.

$$T1 = (V0 + V1) / 2$$

$$T2 = (V1 + V2) / 2$$

$$T3 = (V2 + V3) / 2$$

$$T4 = (V3 + V4) / 2$$

The redness score is generated by using the threshold values as the range values for each score. The higher the redness values, the lower the score for psoriasis lesion.

Table 2 The generated score from the threshold values

Score	Range
0	redness >= T1
1	T1 > redness >= T2
2	T2 > redness >= T3
3	T3 > redness >= T4
4	redness < T4

3.2.4.2 Generate Score for Area Covered by Psoriasis

The area of the lesion image can be determined from the image analysis stage, where the total pixels of the lesion image, *total*, divide by the total pixels for 1 cm^2 image.

$$1\ cm = 38\ pixels$$
$$1cm^2 = 1444\ pixels$$

$$area\ (cm^2) = \frac{total}{1444}$$

The next step is to calculate the area of the lesion on the real body, which can be obtained by multiplying the area of the lesion image with the image scale, S.

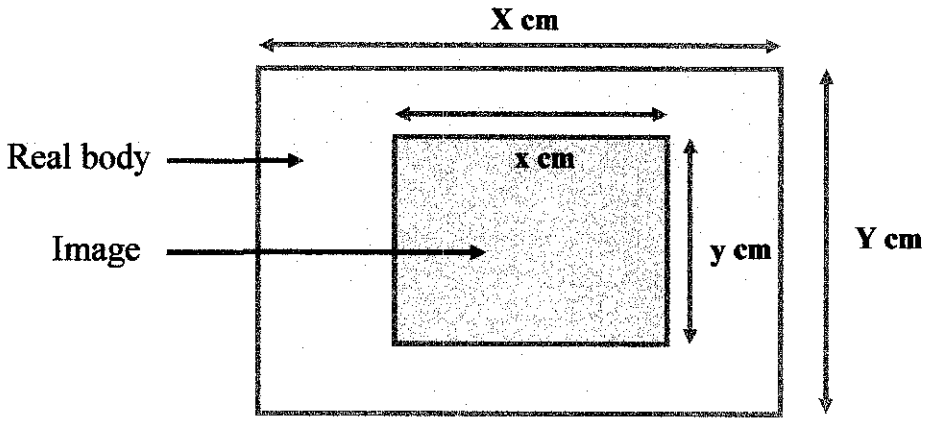


Figure 30 The scale of the image from the real body

The image scale can be calculated by dividing the area of the real body image by the area of the image.

$$\text{Scale, } S = \frac{XY}{xy}$$

$$A\ (cm^2) = S \times \text{area} = S \times \frac{\text{total}}{1444}$$

The Body Surface Area (BSA) is measured using the Mosteller BSA equation where the user needs to enter the height in cm and the weight in kg to get the BSA in m^2.

$$\text{Mosteller BSA } (m^2) = \left(\frac{\text{Height (cm)} \times \text{Weight (kg)}}{3600}\right)^{0.5}$$

The BSA will be converted into cm^2 by multiplying with 10000:

$$\textbf{BSA } (cm^2) = \textbf{BSA } (m^2) \times \textbf{10000}$$

The BSA for each part of the body (head, upper extremities, trunk, and lower extremities) is determined first before calculating the percentage area of the lesion.

Head (BSAh)	= 0.1 x BSA (cm^2)
Upper extremities (BSAu)	= 0.2 x BSA (cm^2)
Trunk (BSAt)	= 0.3 x BSA (cm^2)
Lower extremities (BSAl)	= 0.4 x BSA (cm^2)

For the lesion on the head	= (A / BSAh) x 100%
For the lesion on the upper extremities	= (A / BSAu) x 100%
For the lesion on the trunk	= (A / BSAt) x 100%
For the lesion on the lower extremities	= (A / BSAl) x 100%

This percentage will be used to get the score for the area covered by psoriasis lesion.

Table 3 The range of percentage area for each score

Score	Range
0	**0% (clear)**
1	**<10%**
2	**10 – <30%**
3	**30 – <50%**
4	**50 – <70%**
5	**70 – <90%**
6	**90 – 100%**

3.2.5 Graphical User Interface (GUI)

A Graphical User Interface (GUI) named score is developed in this project to facilitate the scoring for redness and area covered by psoriasis.

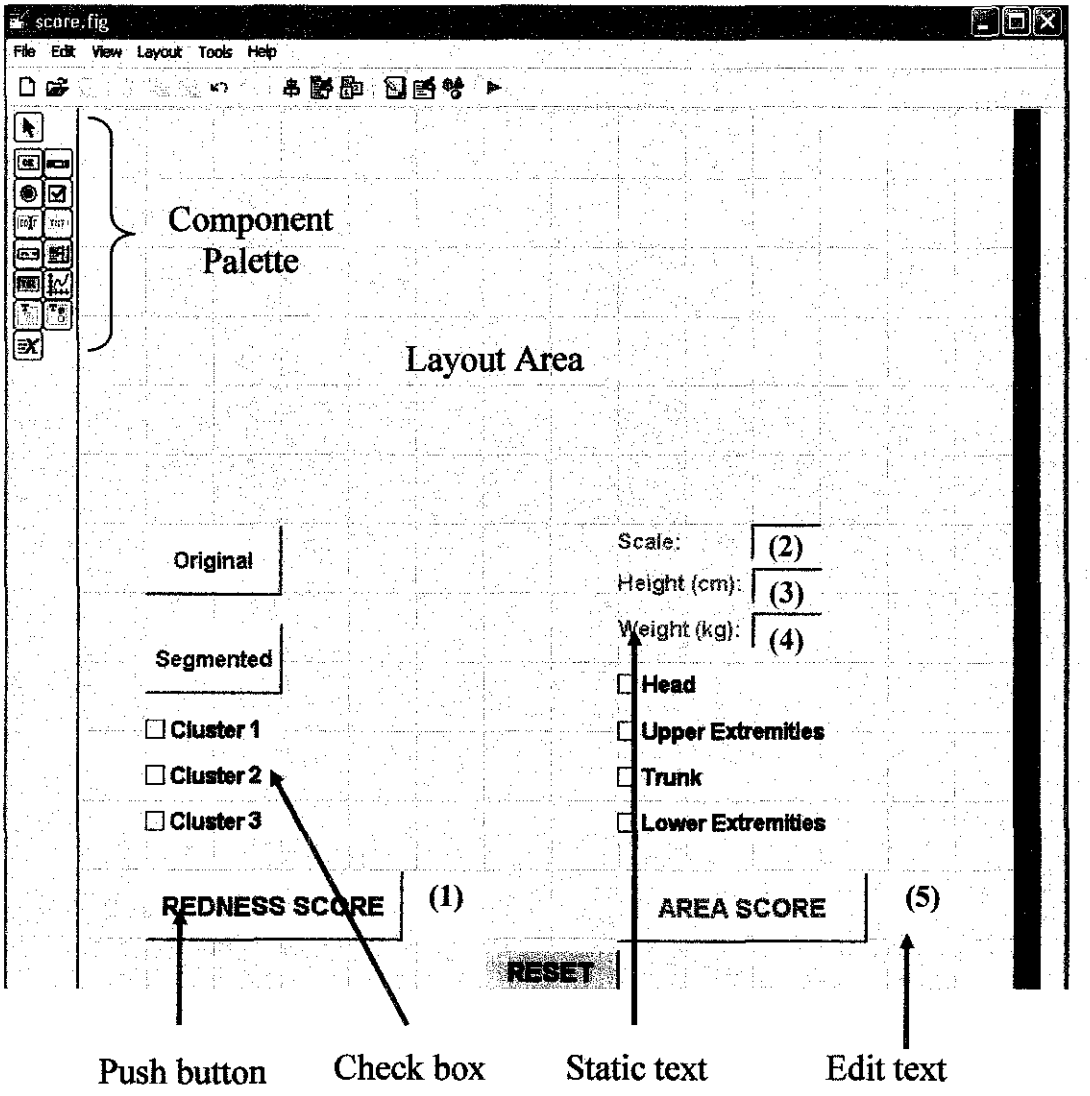


Figure 31 The Layout Editor of the designed GUI

The components used in this GUI namely push button, check box, static text, and edit text, are dragged from the component palette at the left side of the Layout Editor.

Table 4 The functions of the components used in the GUI

Components	Strings	Functions
Push button	Original	Load psoriasis image from graphics file, read the loaded image and display the original image
Push button	Segmented	Segment the image and display the segmented images in the different clusters
Check box	Cluster 1	Display and save the image if the lesion image is obtained in cluster 1
Check box	Cluster 2	Display and save the image if the lesion image is obtained in cluster 2
Check box	Cluster 3	Display and save the image if the lesion image is obtained in cluster 3
Push button	Redness score	Calculate and score the redness
Edit text	Label (1)	Display the redness score
Static text	Scale	Label the scale input
Edit text	Label (2)	Require user to enter the value of scale
Static text	Height (cm)	Label the height input
Edit text	Label (3)	Require user to enter the value of height
Static text	Weight (kg)	Label the weight input
Edit text	Label (4)	Require user to enter the value of weight
Check box	Head	Calculate and save the percent area if the lesion occurred on the head
Check box	Upper extremities	Calculate and save the percent area if the lesion occurred on the upper extremities
Check box	Trunk	Calculate and save the percent area if the lesion occurred on the trunk
Check box	Lower extremities	Calculate and save the percent area if the lesion occurred on the lower extremities
Push button	Area score	Score the percentage area covered by psoriasis
Edit text	Label (5)	Display the area score
Push button	Reset	Reset the redness score and area score

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Redness Score

The redness values of the lesions for five reference images which have been scored from 0 – 4 are obtained in the analysis stage.

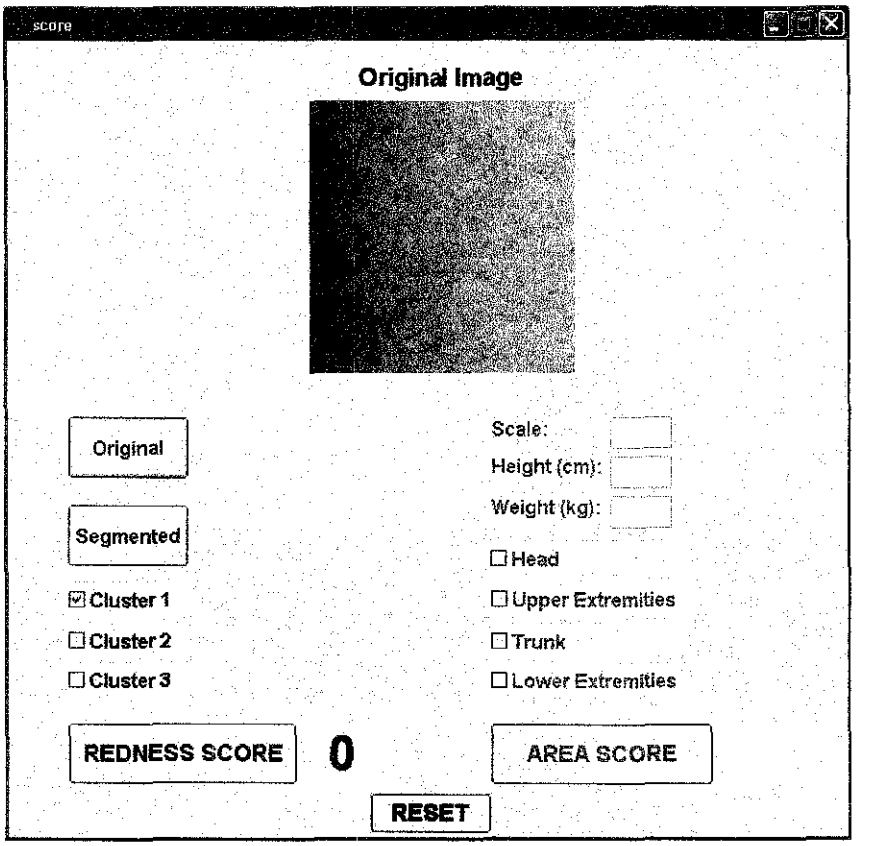


Figure 32 The reference image for redness score = 0

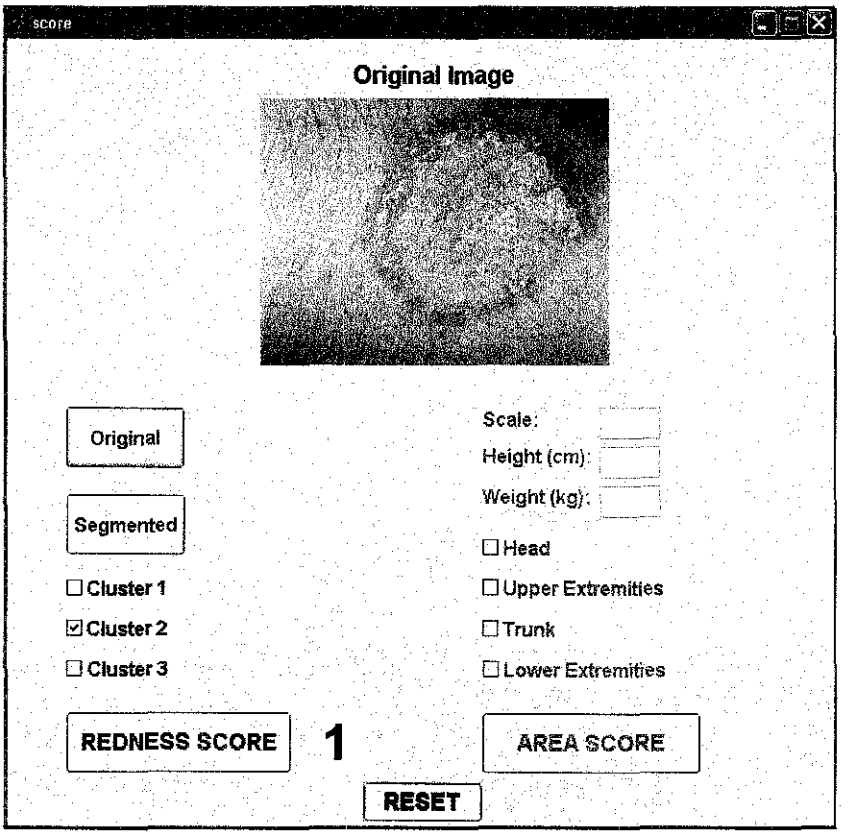


Figure 33 The reference image for redness score = 1

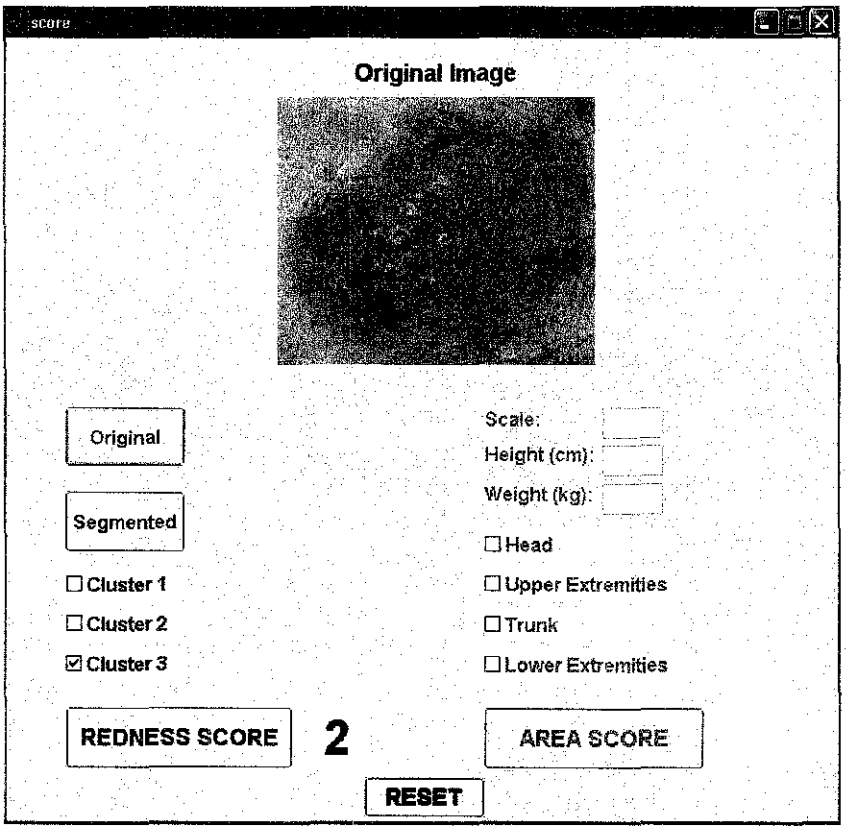


Figure 34 The reference image for redness score = 2

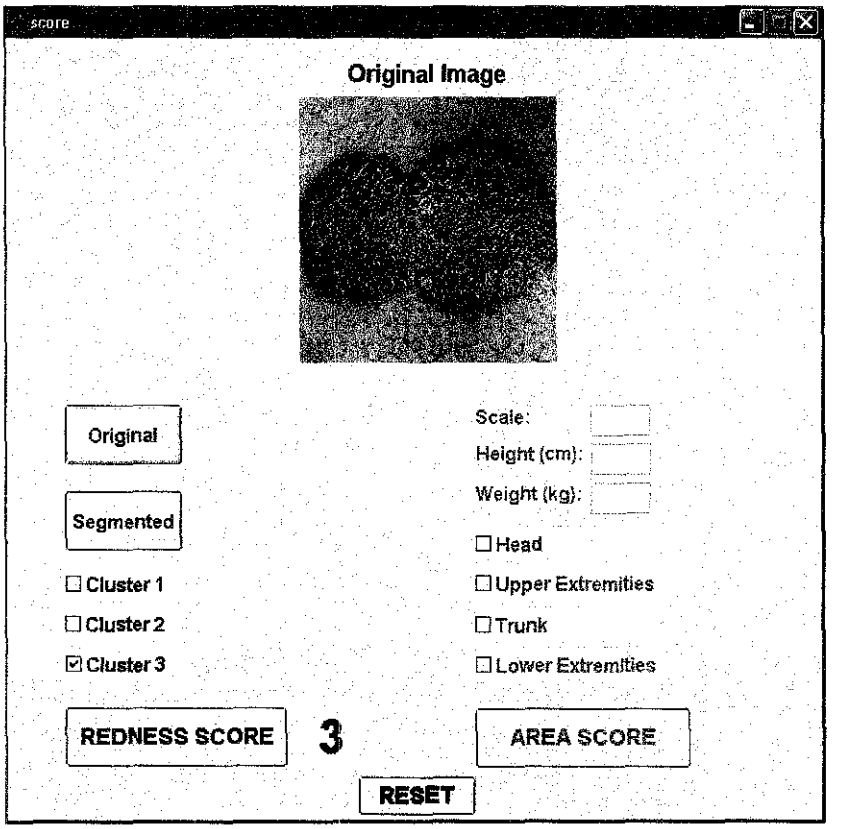


Figure 35 The reference image for redness score = 3

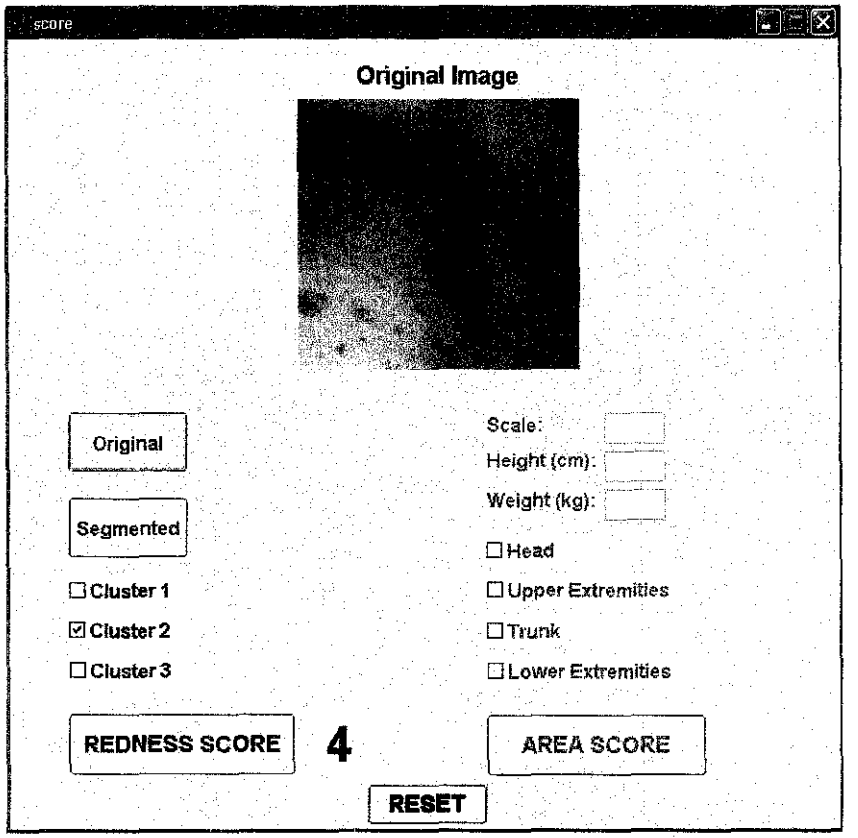


Figure 36 The reference image for redness score = 4

Table 5 The redness values obtained from the different score of lesions

Image	Redness
Psoriasis 0	**191.9326**
Psoriasis 1	**152.2451**
Psoriasis 2	**130.3933**
Psoriasis 3	**103.9350**
Psoriasis 4	**87.1560**

Table 6 The range of the redness values for the different scores

Score	Range
0	**172.0889 – 255.0000**
1	**141.3192 – 172.0888**
2	**117.1642 – 141.3191**
3	**95.5455 – 117.1641**
4	**1.0000 – 95.5454**

Sets of five images from each score (0 – 4) are used as the reference images in order to generate the redness score. The system will be tested after generating the score using a set of images. The accuracy of the system is determined from the results of the test. The set of images will be replaced with the new set of images if a low accuracy is obtained from the test. This procedure is repeated until the highest accuracy is achieved. The redness values of the final set of images are used to generate the redness score. So, the redness severity of the lesion images will be scored based on the range of redness values in Table 6.

The developed GUI facilitates the scoring procedure by a few steps taken as follows:

a) Click the original push button

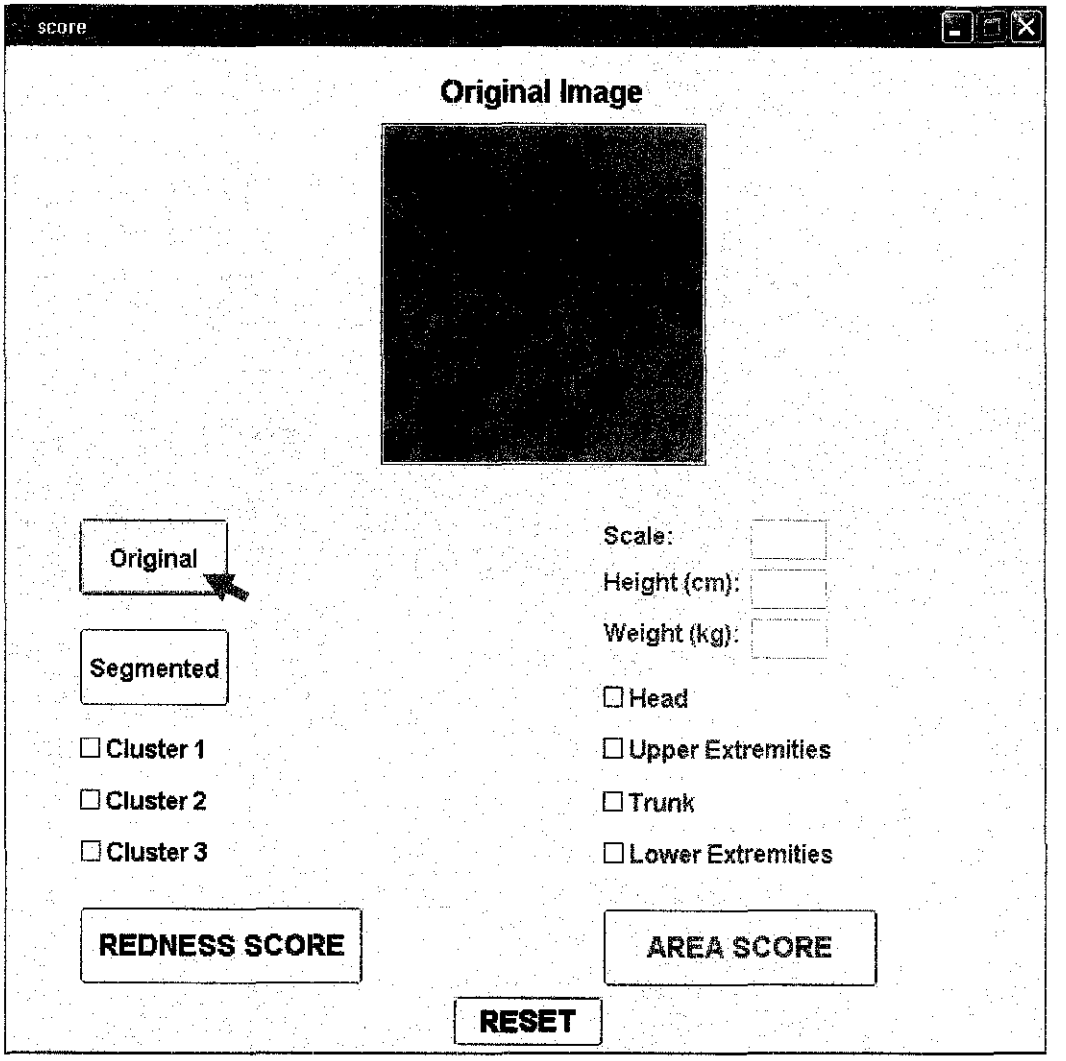


Figure 37 GUI result for the original push button

The original push button represents the original psoriasis image. By clicking the push button, the system displays a standard dialog box for a user to select and load an original psoriasis image before reading and displaying the image.

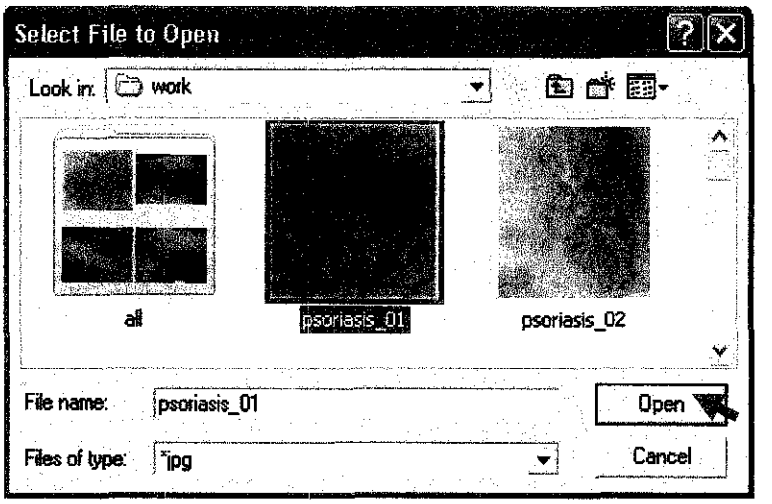


Figure 38 A standard open file dialog box to load a psoriasis image

b) Click the segmented push button

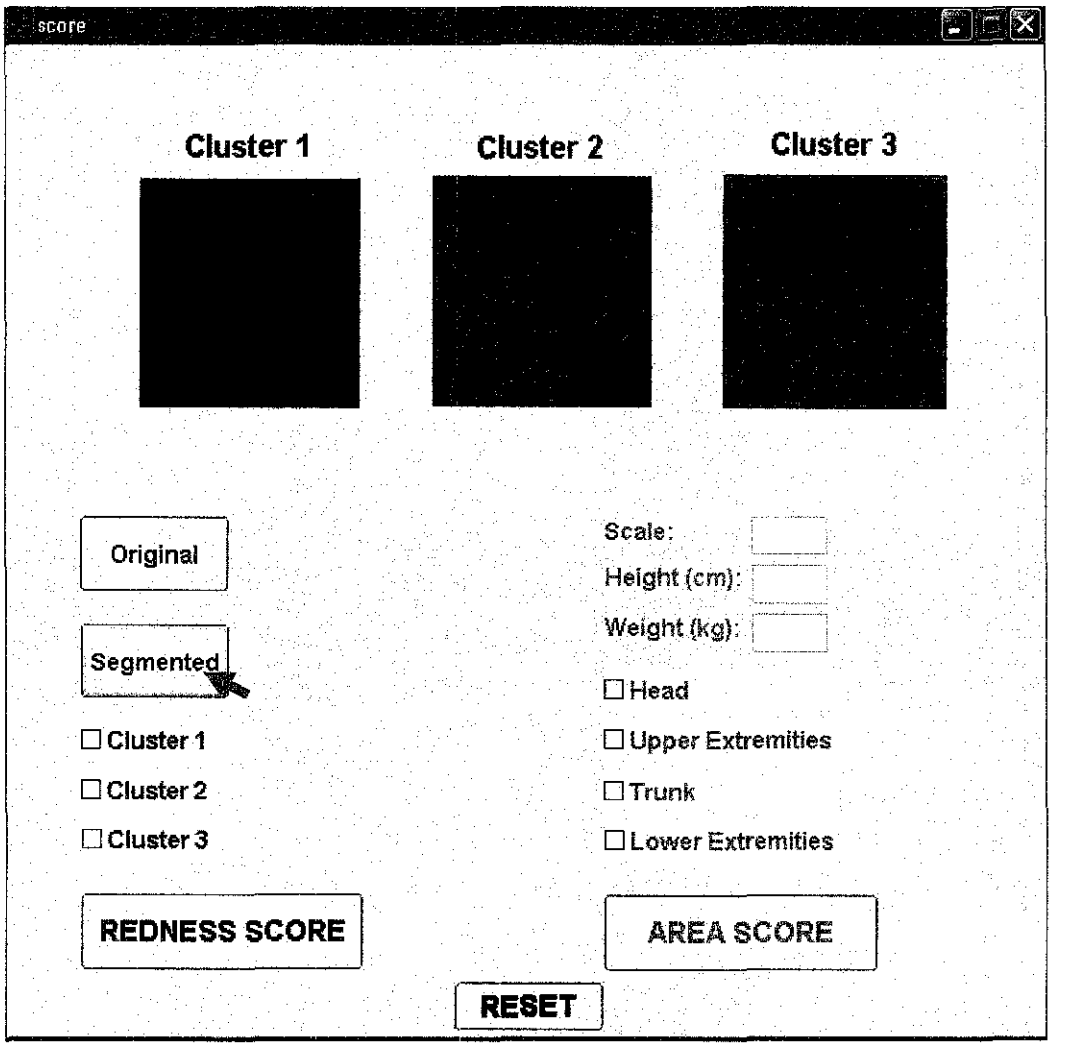


Figure 39 GUI result for the segmented push button

The segmented push button represents the segmented images of the psoriasis image. The original image will be enhanced and segmented by clicking this push button.

The image enhancement gives the better results in segmentation. Besides, the greater redness values can be obtained from the enhanced images which contribute to the wider range of redness for each score. It is important to have a wider range of values in each score in order to avoid misclassifying the severity of lesion images.

The thresholding technique and concept which is interested in light objects on a dark background is used for the segmentation process. The k-means clustering algorithm is applied in order to segment the lesion image. The image is converted to the L*a*b* color space where the color information is in the 'a*' (color falls along red – green axis) layer and 'b*' (color falls along blue – yellow axis) layer. Since the image is in a uniform color space, the difference between the color information can be measured using a simple Euclidean distance metric.

Three clusters are specified in this system, which is quite enough to extract the lesion image from the skin or background. Using this clustering algorithm, it obtains the mean of each cluster and computes the distance of each point from the corresponding cluster mean to assign each point to the cluster it is nearest to. Based on Figure 39, Cluster 1 has no image because there is no point nearest to the mean of this cluster. The lesion are obtained in the Cluster 2 whereas the skin in Cluster 3.

c) Tick the related check box

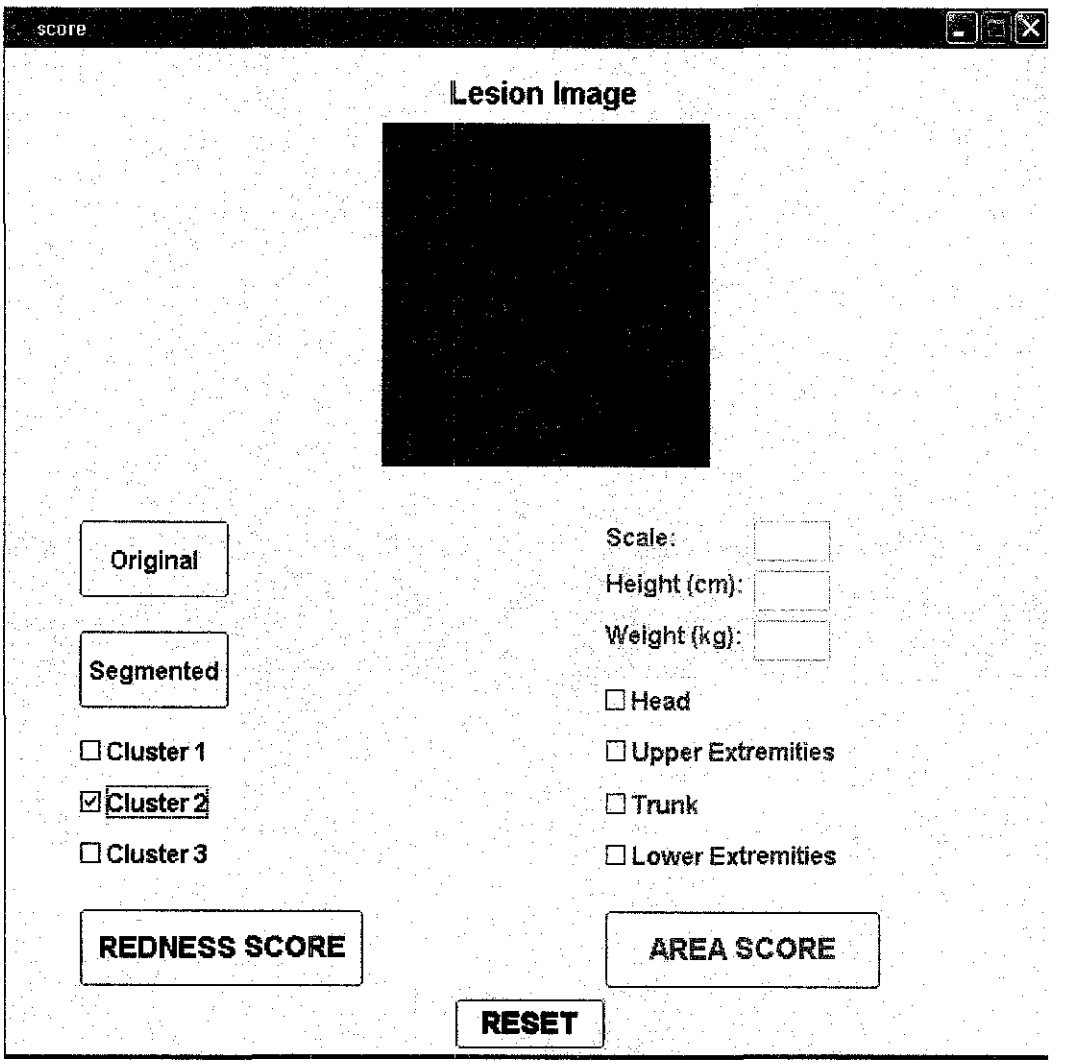


Figure 40 GUI result for the clusters check box

In this step, the image lesion is chosen based on the segmentation results. Each check box represents the image in each cluster. The lesion image can be chosen by ticking the related check box where the chosen image will be saved for the image analysis.

d) Click the redness score push button

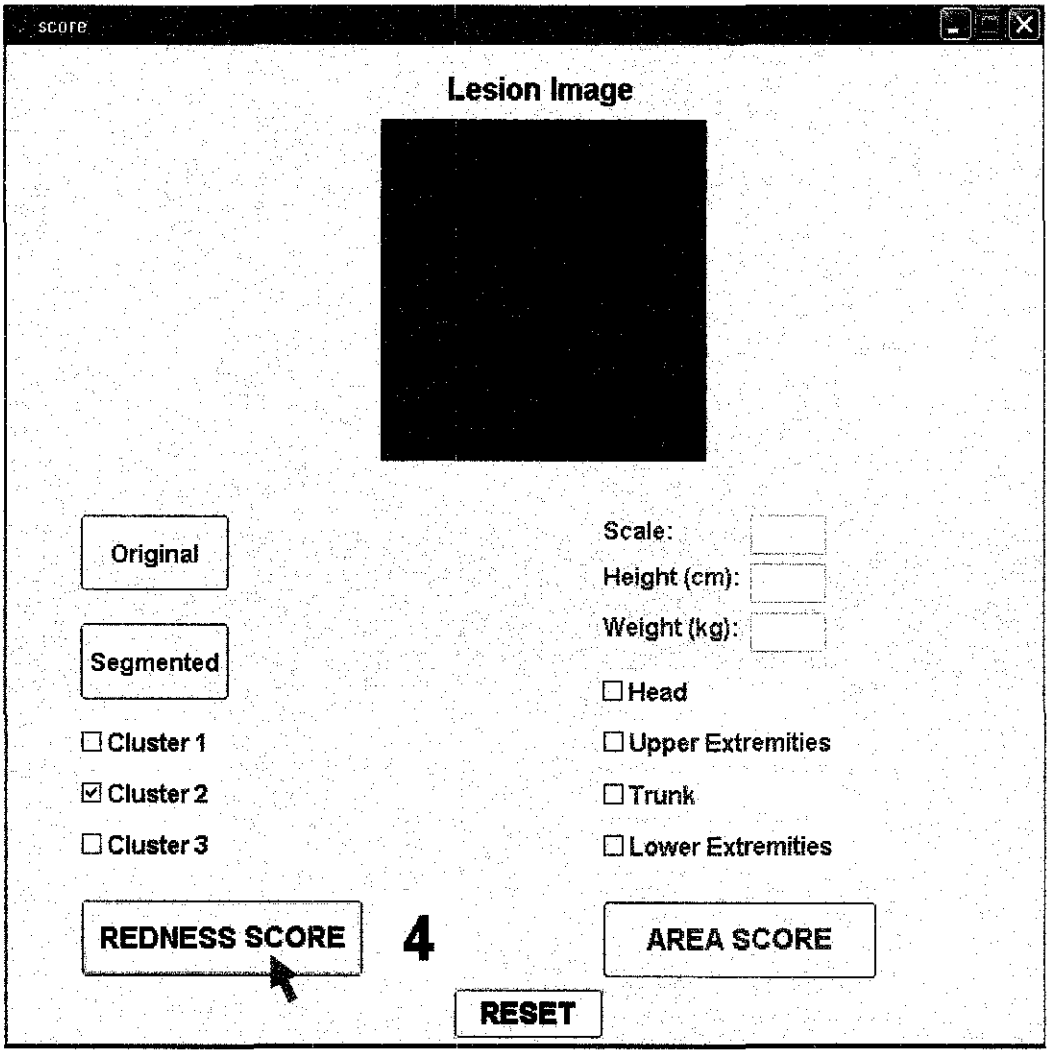


Figure 41 GUI result for the redness score push button

The final step in redness scoring procedure is clicking the redness score push button which represents the redness score from 0 – 4. By clicking this main push button, the system analyzes the saved lesion image to get the redness value of the lesion. The lesion will be scored based on the range of redness values in Table 6.

This redness score will be used in PASI scoring procedure to obtain the severity level of the lesion. The suitable treatments will be given to the patients with psoriasis skin disease based on the PASI score.

4.2 Area Score

The development of GUI gives the easier way to score the percentage of area covered by psoriasis lesions.

a) Enter the required information

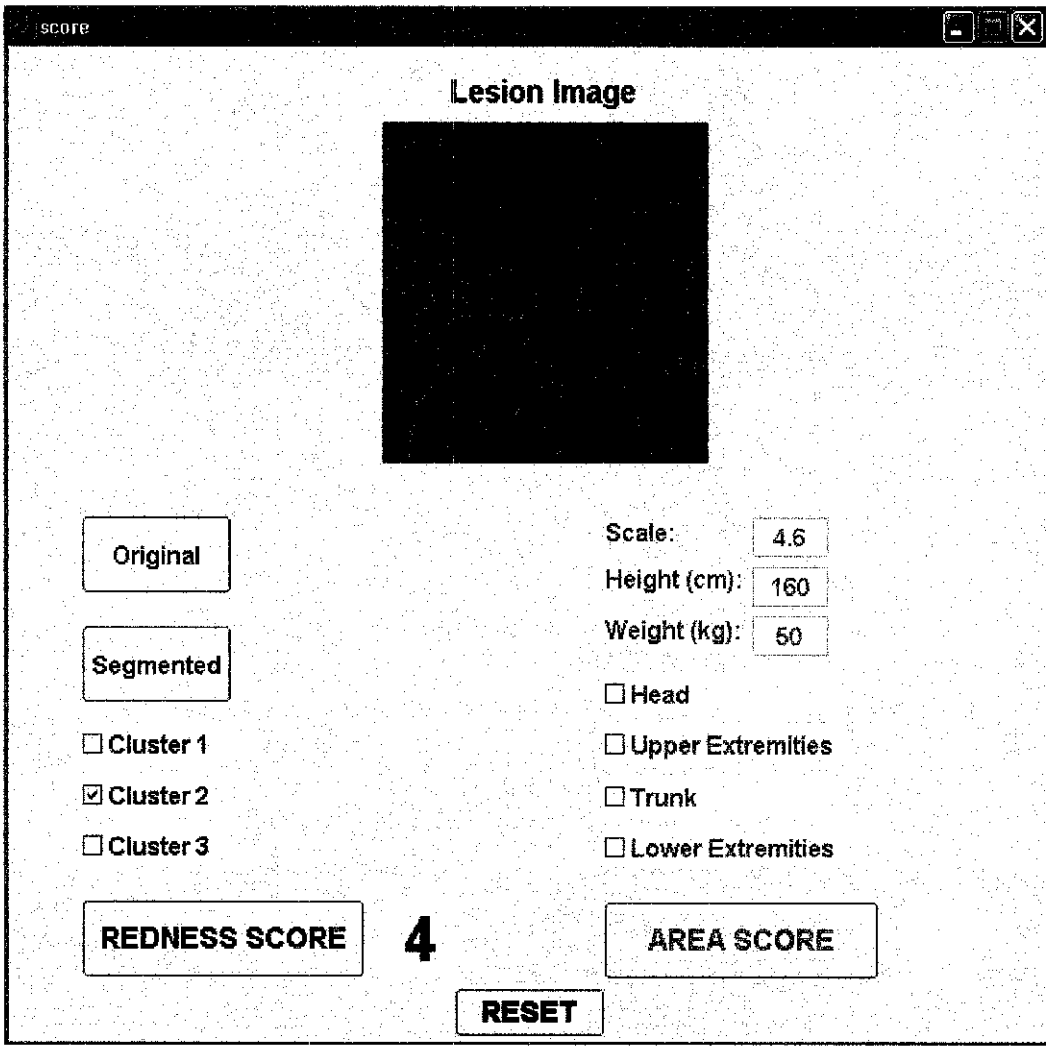


Figure 42 GUI result for entering some required information

Some information is required to get the score for the percentage of area covered by psoriasis. The scale is needed to calculate the actual value of the lesions area whereas the height and weight is used to calculate the Body Surface Area (BSA). The system saves all the entered values, which will be used in order to get the area score. The total pixels of the lesions are already obtained while scoring the redness severity.

Calculation example:

The image scale	= 4.6
Height	= 160 cm
Weight	= 50 kg
The total pixels of the lesion, total	**= 32448 pixels**
The area of the image lesion, area	= total / 1444
	= 32448 / 1444
	= 22.4709 cm^2
The area of the image on the real body, A	= area x S
	= 22.4709 x 4.6
	= 103.3661 cm^2
Body Surface Area, BSA	$= [(h \times w) / 3600]^{0.5}$
	$= [(160 \times 50) / 3600]^{0.5}$
	= 1.4907 m^2
	= 14907 cm^2

b) Tick the related check box

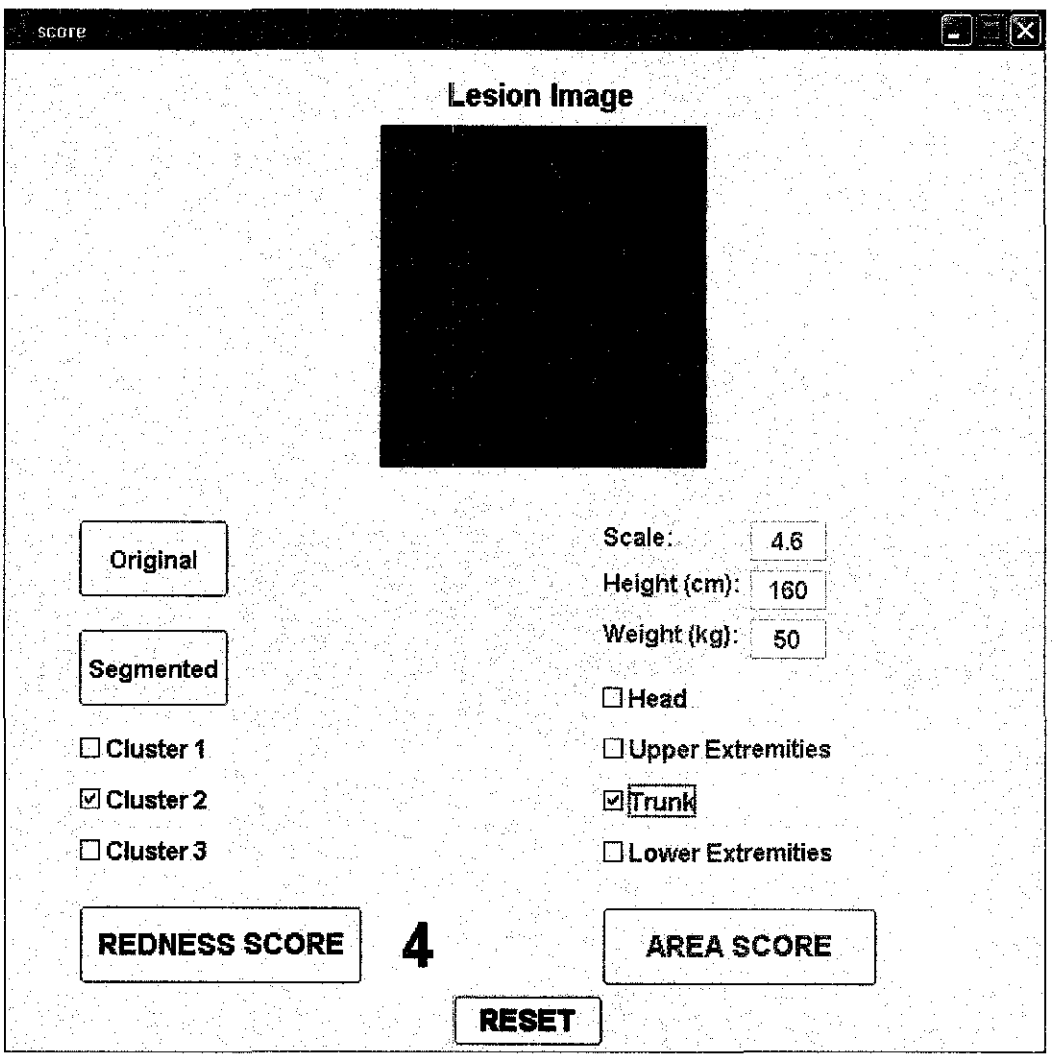


Figure 43 GUI result for the body parts check box

The next step is choosing any one part of the body where the psoriasis lesions appear. Each check box represents the calculation of BSA for each part of the body. For this example, the lesion is appearing on the trunk area. So the body surface area for trunk, BSAt, is calculated and saved in order to get the area score.

Body Surface Area for Trunk, BSAt = 0.3 x BSA

= 0.3 x 14907

= **4472.1 cm^2**

c) Click the area score push button

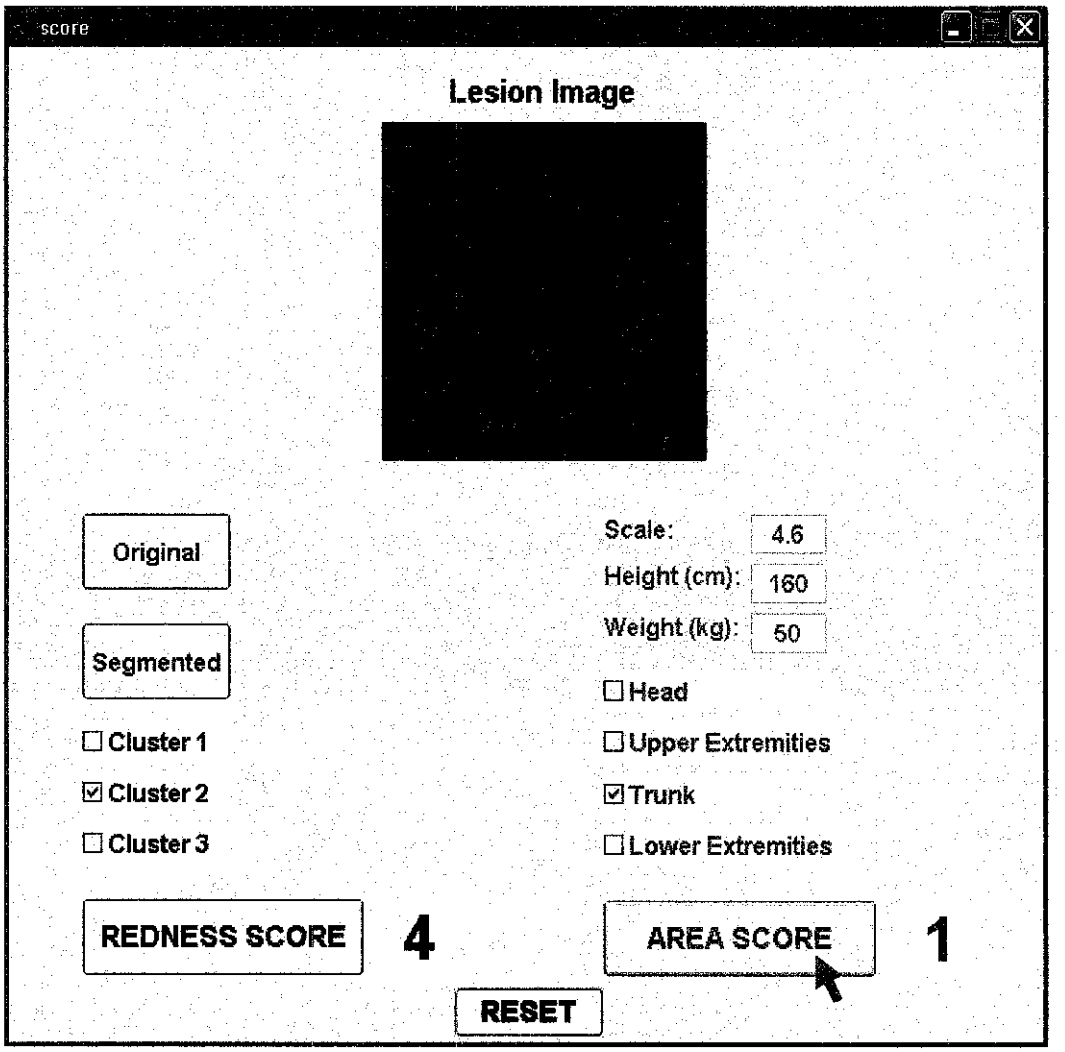


Figure 44 GUI result for the area score push button

The area score push button represents the score of the percentage area covered by psoriasis lesions. By clicking this push button, the lesion area is divided by the trunk surface area, BSAt, to compute the percentage of lesions area and displays the corresponding score. Based on Table 3, the percentage is in the range of (<10%) gives the score for area covered by psoriasis on the trunk, $A_t = 1$. This score will be used to calculate the PASI score which is a standard assessment of extensive psoriasis.

The percentage for area covered by psoriasis = (A / BSAt) x 100%

= (103.3661 / 4472.1) x 100%

= 2.31%

CHAPTER 5
CONCLUSION AND RECOMMENDATION

5.1 Conclusion

Because of the rapid technical advances in medical imaging technology and the introduction of new clinical applications, medical image analysis has become a highly active research field. Improvements in image quality, changing clinical requirements, advances in computer hardware, and algorithmic progress in medical image processing all have a direct impact on the state of the art in medical image analysis. This project could be a great significance to the diagnosis and treatment process in the current situation of skin diseases in Malaysia. The results are encouraging for the development of a noninvasive skin lesion classification system that dermatologists could use as an electronic second opinion in clinical settings.

From this image based system project, the level of redness severity for psoriasis lesion and the area covered by psoriasis can be scored by using the Image Processing Toolbox in MATLAB software. The capability of this system in classifying the redness severity is quite high with the percentage of accuracy up to 90%. But, the improvements and modifications are still needed to increase the system performance.

Having a quantitative measure of disease severity has facilitated clinical trials of new agents for psoriasis. PASI has led to new treatments that have advanced the dermatologists' ability to care for patients with severe disease. The development of these agents, some of which set new standards of high cost of dermatologic treatment, has led to new challenges as well.

5.2 Recommendation

Although this system have achieved a high accuracy in classifying the level of redness severity for psoriasis skin disease, some modifications and improvements is still needed in the development of this image based system project. This system can be improved by using the other image processing technique which gives the better results especially in the segmentation process.

Research on the various techniques which are related to skin diseases can be conducted in order to develop a flexible, faster and more user friendly system in the future. For now, the system is only able to analyze the lesion after selecting the lesion image from the segmentation. It is possible to have a system which can straight away scoring the severity level of the lesion after the image acquirement.

REFERENCES

[1] Robin Marks, Anne Plunkett, Kate Merlin, Nicole Jenner, *Atlas of Common Skin Diseases in Australia*, Department of Dermatology St. Vincent's Hospital, Melbourne, 1999

[2] Nixon, Aguado, *Feature Extraction & Image Processing*, Great Britain, Newnes, 2001

[3] A. Fadzil, Farah A Nordin, *Telehealthcare-Monitoring of Skin Diseases*, 3rd APT Telemedicine Workshop, Kuala Lumpur, 2005

[4] S. Sigurdsson, PA Philipsen, LK Hansen, J. Larsen, M. Gniadecka, HC Wulf, *Detection of Skin Cancer by Classification of Raman Spectra*, IEEE Transactions on Biomedical Engineering, 2004

[5] H. Ganster, A. Pinz, R. Rohrer et al., *Automated Melanoma Recognition*, IEEE Transactions on Medical Imaging 20(3), pp. 233 – 239, March 2001

[6] M. Mehrubeoglu, N. Kehtarnavaz, G. Marquez, M. Duvic, LHV Wang, *Skin Lesion Classification Using Oblique-Incidence Diffuse Reflectance Spectroscopic Imaging*, Applied Optics, 2002; 41(1): 182 – 192

[7] GA Hance, SE Umbaugh, RH Moss, WV Stoecker, *Unsupervised Color Image Segmentation*, IEEE Engineering in Med. and Bio. Mag., vol. 15, 1996

[8] A. Fadzil, Thai Vinh Tuan, *Content-Based Skin Image Retrieval*, Universiti Teknologi PETRONAS

[9] http://www.fda.gov/fdac/features/2004/504_psoriasis.html *Psoriasis: More Than Cosmetic*

[10] http://worldpsoriasisday.com/ *World Psoriasis Day Media Materials*

[11] S R Feldman, G G Krueger, *Psoriasis Assessment Tools in Clinical Trials*, ARD Online, 2004

[12] http://www.psorsite.com/docs/pasi.html *Dave's Psoriasis Info – The Psoriasis Area and Severity Index*

[13] Konstantinos N. Plataniotis, Anastasios N. Venetsanopoulos, *Color Image Processing and Applications*, Springer, 2000, pp. 1 – 4, 32 – 38

[14] http://www.graphicsnews.com, *The RGB Code: The Mysteries of Color Revealed*

[15] L. Luccheseyz, S. K. Mitray, *Unsupervised Low-Frequency Driven Segmentation of Color Images*, Dept. of Electrical and Computer Eng., University of California, Santa Barbara, 1999

[16] http://students.iiit.net/~arul/report/node18.html *Segmentation Using K-Means Algorithm*

[17] http://mathworld.wolfram.com/K-MeansClusteringAlgorithm.html *K-Means Clustering Algorithm*

[18] http://en.wikipedia.org/wiki/Body_surface_area *Body Surface Area*

[19] http://www.halls.md/body-surface-area/bsa.htm *Body Surface Area Calculator for Medication Dose*

[20] MATLAB7 Documentations

APPENDICES

APPENDIX A

STATISTICS ON SKIN DISEASES

COMMON SKIN DISEASES

Table 1 — Prevalence of self reported skin diseases in adults

	Prevalence % (95% confidence interval)			
	Past 2 weeks		Past 6 months (excluding last 2 weeks)	
Acne/pimples	16.2	(9.6-22.7)	9.4	(5.4-12.5)
Cold sores	15.1	(7.3-18.9)	30.3	(23.9-56.7)
Dermatitis/eczema	25.5	(18.1-32.8)	12.6	(7.9-17.3)
Psoriasis	4.5	(1.0-7.9)	3.5	(0.9-5.1)
Skin cancer	0.5	(0.0-0.9)	5.0	(2.3-7.7)
Thrush	2.5	(0.2-4.8)	5.7	(2.7-8.6)
Tinea	11.2	(5.9-16.5)	19.4	(15.8-24.9)
Urticaria/hives	1.1	(0.0-2.7)	0.9	(0.0-2.1)
Warts	16.1	(9.8-22-4)	8.6	(4.9-12.4)

DERMATOLOGY ADMISSIONS TO A BRITISH GENERAL HOSPITAL DURING WORLD WAR I

	1915		1916						
	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Total
Impetigo	122	172	151	161	220	170	147	116	1,259
Scabies	95	770	170	8	25	5	9	23	1,105
Boils	24	59	50	51	42	36	65	48	375
Pediculosis	–	17	62	69	36	3	5	6	198
Psoriasis	11	24	17	17	29	21	36	29	184
Eczema	7	22	18	11	37	22	33	31	181
Seborrhoea	8	12	13	11	40	18	18	11	131
Acne	6	22	19	12	8	6	10	7	90
Dermatitis	7	10	10	11	11	8	8	7	72
Syphilis	8	13	4	3	6	4	17	16	71
Folliculitis	1	11	13	9	8	2	5	1	50
Urticaria	2	4	3	3	2	11	8	1	34
Sycosis	4	2	4	1	3	2	5	7	28
Pityriasis rosea	3	2	2	5	3	2	2	3	22
Erythema	3	7	5	–	4	1	–	1	21
Carbuncle	2	4	3	1	4	2	1	2	19
Herpes zoster	2	2	2	–	3	3	1	–	13
Lupus	–	–	2	–	–	3	2	2	9
Erysipelas	–	–	–	–	1	–	1	2	4
Ecthyma	–	–	1	2	–	–	–	–	3
Lichen planus	–	–	1	1	–	–	1	–	3
Erythema nodosum	–	–	–	–	–	–	–	2	2
Ichthyosis	–	–	1	–	–	–	–	1	2
Sudamina	–	–	–	–	–	–	–	2	2

Adapted from Macpherson WG, Herringham WP, Elliott TR, Balfour A, eds. Medical services. Diseases of the war. In: *History of the Great War*. Vol 2. London, England: His Majesty's Stationery Office; 1923: 68.

MOST COMMON DIAGNOSES IN NEW PATIENTS SEEN AT DERMATOLOGY CLINIC, 17TH FIELD HOSPITAL, SAIGON, VIETNAM, JULY 1967 (% of total shown)

Diagnosis	No. of Cases	(%)
Pyoderma	47	(10.0)
Miliaria	43	(9.2)
Tinea	43	(9.2)
Verrucae	37	(7.9)
Eczematous dermatitis	26	(5.6)
Candidiasis	22	(4.7)
Infected eczematous dermatitis	20	(4.3)
Acne	18	(3.8)
Tinea versicolor	15	(3.2)
Urticaria	13	(2.8)
Contact dermatitis	11	(2.3)
Plantar warts	11	(2.3)
Alopecia areata	10	(2.1)
Pseudofolliculitis barbae	9	(1.9)
Psoriasis	7	(1.5)
Others	137	(29.2)
Total	469	(100.0)

Adapted from Allen AM. *Skin Diseases in Vietnam, 1969–72.* In: Ognibene AJ, ed. *Internal Medicine in Vietnam.* Vol 1. Washington, DC: Medical Department, US Army, Office of The Surgeon General and Center of Military History; 1977: 38.

PROPORTIONATE DISTRIBUTION OF SKIN DISEASES SEEN IN U.S. DERMATOLOGY CLINIC, 95TH EVACUATION HOSPITAL, DA NANG, VIETNAM, 15 MAY 1970 TO 31 JULY 1971 (14.5 MONTHS) (% of total shown)

Disease	No. of Cases	(%)
Verrucae, all types	729	(15.83)
Acne, all types	466	(10.12)
Dermatophytosis, all types	371	(8.06)
Pseudofolliculitis barbae	289	(6.28)
Penile ulcer [? chancroid]	221	(4.80)
Miliaria	199	(4.32)
Pyoderma, all types	178	(3.87)
Contact dermatitis	167	(3.63)
Urticaria	126	(2.74)
Tinea versicolor	123	(2.67)
Psoriasis	106	(2.30)
Atopic dermatitis	95	(2.06)
Dyshidrosis	95	(2.06)
Alopecia areata	82	(1.78)
Monilia	71	(1.54)
Lichen planus	70	(1.52)
Herpes progenitalis	68	(1.48)
Seborrheic dermatitis	56	(1.22)
Miscellaneous dermatoses and dermatitides	51	(1.11)
Insect bites	48	(1.04)
Molluscum contagiosum	41	(0.89)
Sebaceous cyst	40	(0.87)
Pityriasis rosea	39	(0.85)
Hand and foot eczema	37	(0.80)
Lichen simplex chronicus	35	(0.76)
Syphilis infection, late and early	33	(0.72)
Erythema multiforme	32	(0.69)
Nevi	32	(0.69)
Balanitis	31	(0.67)
Basal cell epithelioma	25	(0.54)
Keloids	24	(0.52)
Corns and calluses	24	(0.52)
Drug eruptions	21	(0.45)
Vitiligo	20	(0.43)
Photoallergy	15	(0.33)
Nummular eczema	14	(0.30)
Pruritus	14	(0.30)
No diagnosis	56	(1.22)
Others	461	(10.02)
Total	4,605	(100.00)

Adapted from Allen AM. *Skin Diseases in Vietnam, 1965–72.* In: Ognibene AJ, ed. *Internal Medicine in Vietnam.* Vol 1. Washington, DC: Medical Department, US Army, Office of The Surgeon General and Center of Military History; 1977: 39.

PSORIASIS

ARTHRITIS IN PSORIASIS

Type of Arthritis	Characteristics
Asymmetric involvement of a few joints of the fingers	Affects 70% of patients with psoriatic arthritis; "sausage digit"
Symmetric polyarthritis	Affects 15% of patients with psoriatic arthritis; resembles rheumatoid arthritis (seronegative)
Distal interphalangeal joint involvement	Affects 5% of patients with psoriatic arthritis; "classic" psoriatic arthritis
Arthritis mutilans	Affects 5% of patients with psoriatic arthritis; deforming arthritis with bony destruction, telescoping of digits, and ankylosis
Ankylosing spondylitis	Affects 5% of patients with psoriatic arthritis; may also have peripheral joints involved

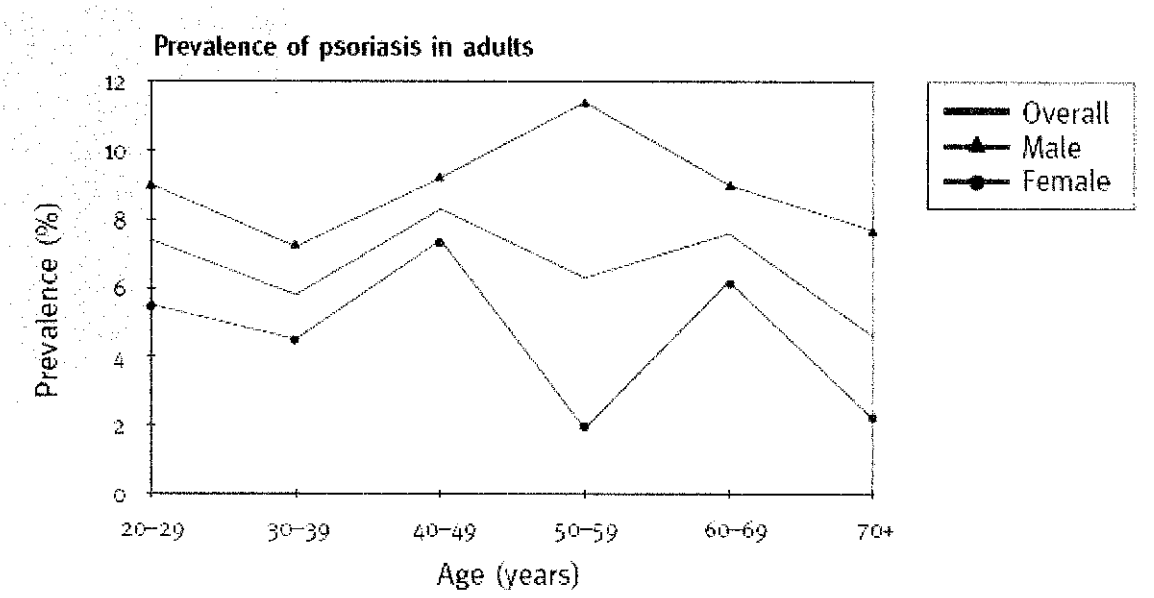
Data source: Moll JMH, Wright V. Familial occurrence of psoriatic arthritis. *Ann Rheum Dis.* 1973;22:181.

Severity of psoriasis in adults

		No. with psoriasis	Mild %	Moderate %	Severe %
	Overall	99	81.1	16.1	2.8
	Male	65	77.6	18.1	4.3
	Female	34	87.5	12.5	0.0
Age (years)	20-29	12	75.1	16.6	8.3
	30-39	12	66.6	33.4	0.0
	40-49	21	94.2	0.0	5.8
	50-59	19	77.3	22.7	0.0
	60-69	21	80.8	19.2	0.0
	70+	14	86.5	13.5	0.0

Prevalence of psoriasis in adults

		No. examined	Prevalence %	(95% CI)
	Overall	1,457	6.6	(5.4-7.9)
	Male	670	8.9	(6.8-11.0)
	Female	787	4.5	(3.2-6.3)
Age (years)	20-29	156	7.4	(3.9-10.9)
	30-39	211	5.8	(2.9-8.7)
	40-49	272	8.3	(5.0-11.6)
	50-59	267	6.3	(3.2-9.5)
	60-69	268	7.6	(4.1-11.2)
	70+	283	4.6	(2.7-7.8)



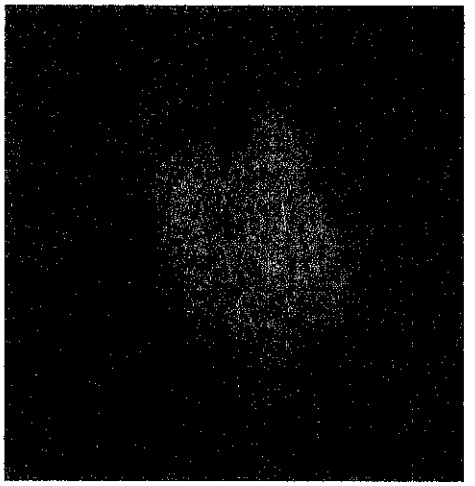
The Dermatology Life Quality Index (DLQI) by adults with psoriasis

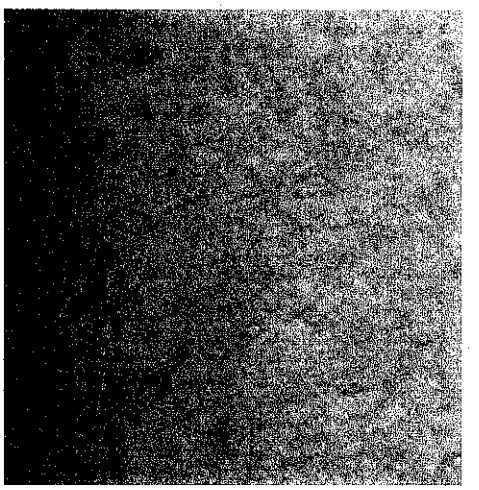
	DLQI question *Over the last week...*	ANSWERS % (n=90) very much	a lot	a little	not at all / not relevant
1	how itchy, sore, painful or stinging has your skin been?	2.2	6.7	46.7	44.4
2	how embarrassed or self-conscious have you been because of your skin?	1.1	5.6	20.0	73.3
3	how much has your skin interfered with you going shopping or looking after your home or garden?	0.0	3.3	5.6	91.1
4	how much has your skin influenced the clothes you wear?	0.0	0.0	13.3	86.7
5	how much has your skin affected any social or leisure activities?	0.0	2.2	3.3	94.4
6	how much has your skin made it difficult for you to do any sport?	0.0	1.1	2.2	96.7
7	how much has your skin been a problem at work or studying?	0.0	2.2	6.7	91.1
8	how much has your skin created problems with your partner or any of your close friends or relatives?	0.0	0.0	6.7	93.3
9	how much has your skin caused any sexual difficulties?	1.1	1.1	3.3	94.4
10	how much of a problem has the treatment for your skin been, for example, by making your home messy, or by taking up time?	1.1	3.3	17.8	77.8

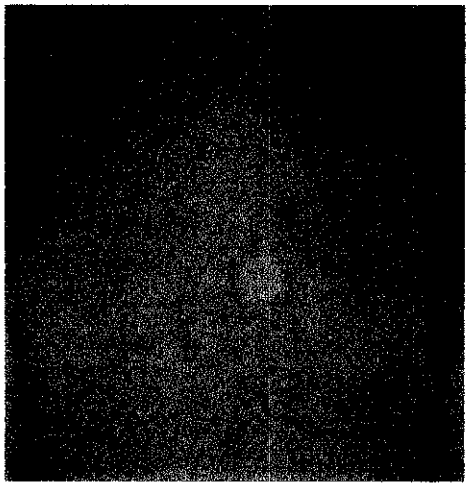
APPENDIX B

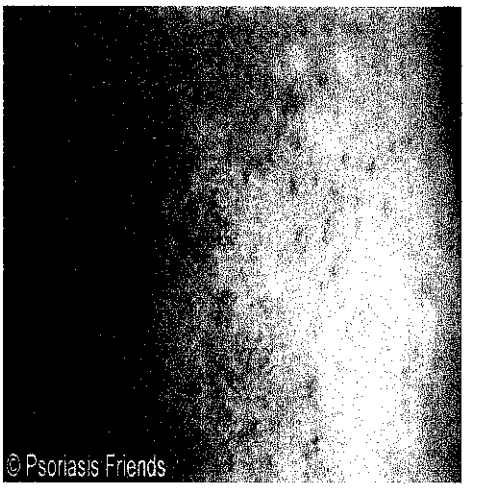
SCORING RESULTS

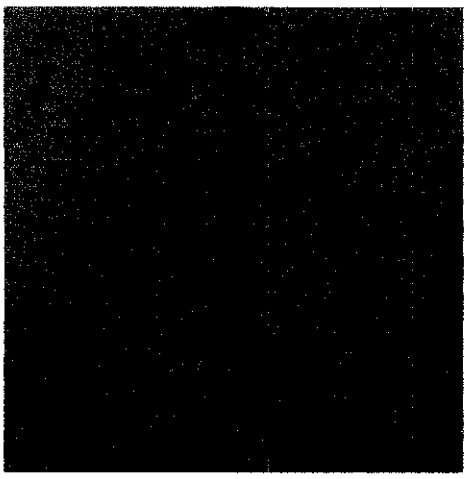
REDNESS SCORE = 0

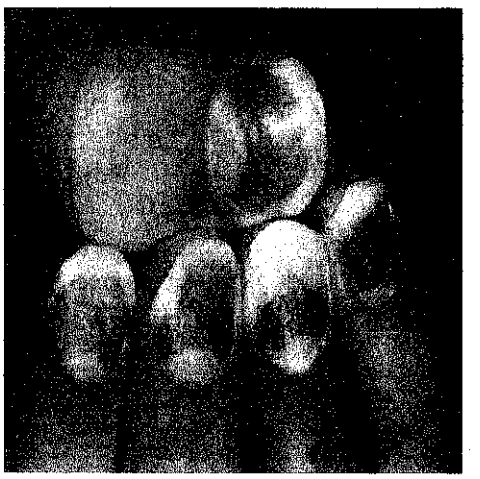




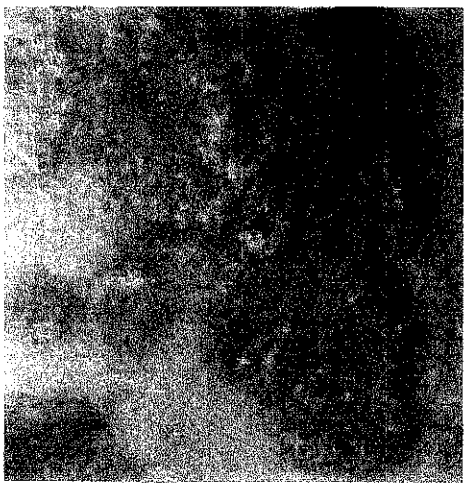
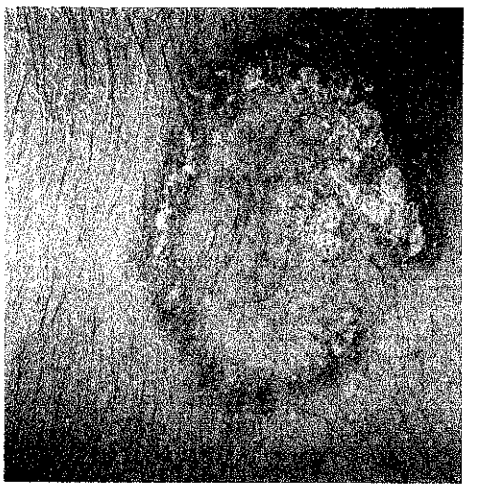


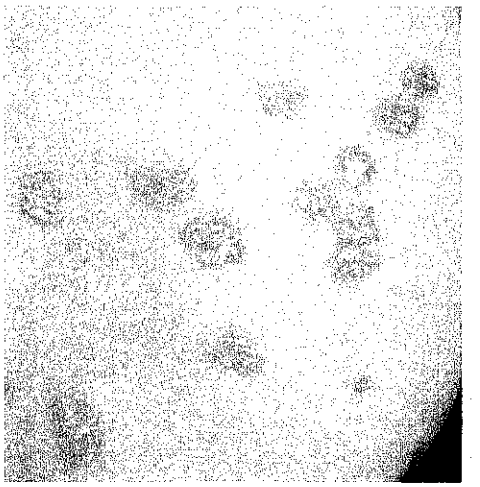


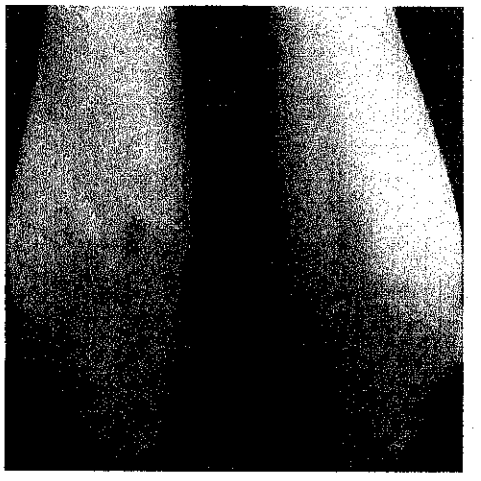


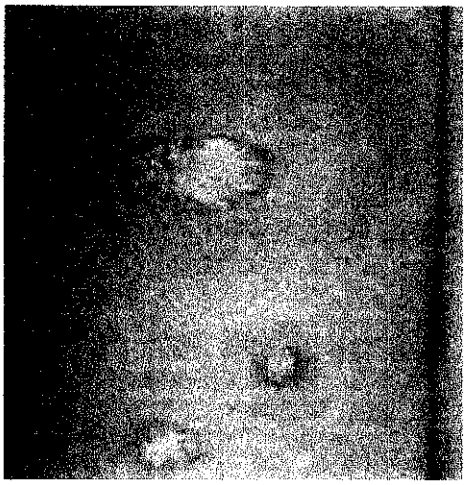


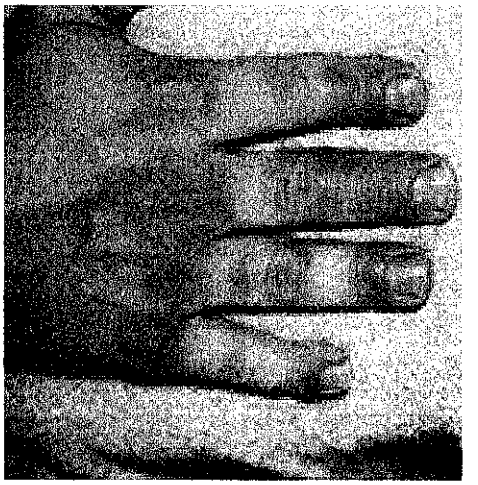
REDNESS SCORE = 1

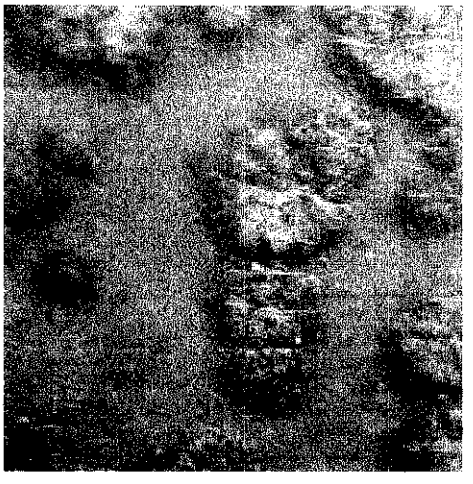



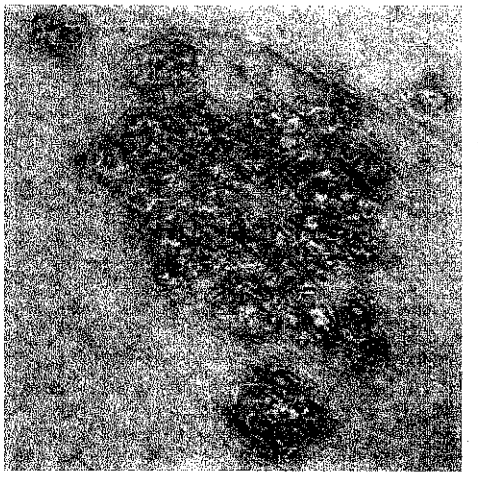


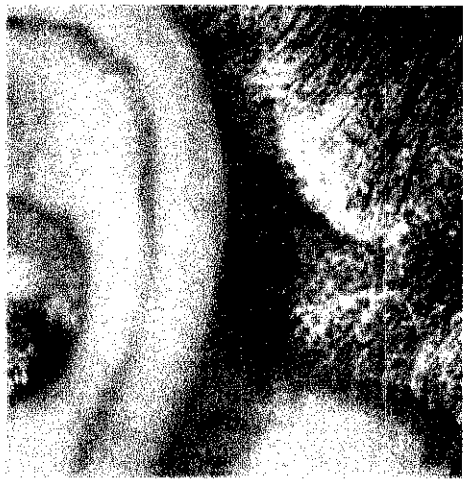


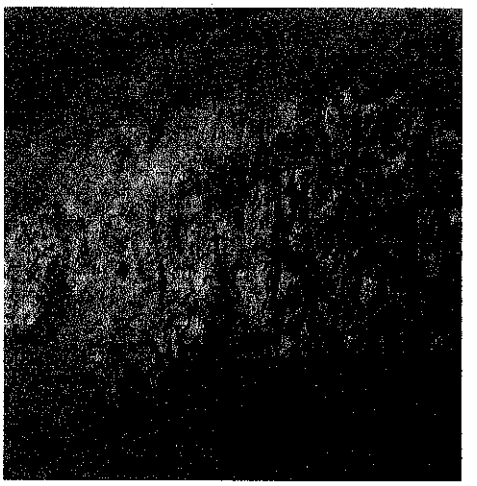


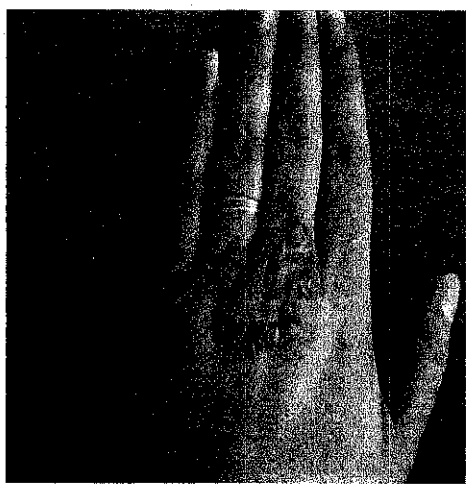


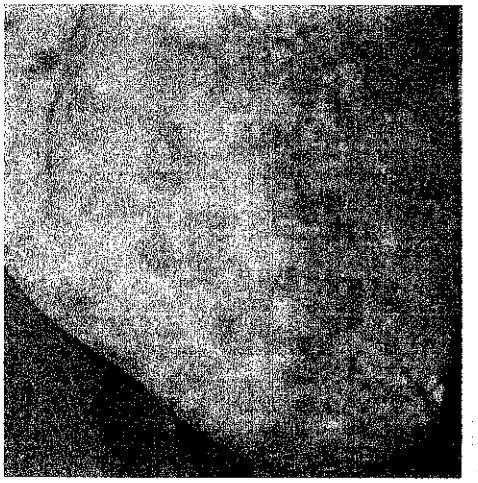


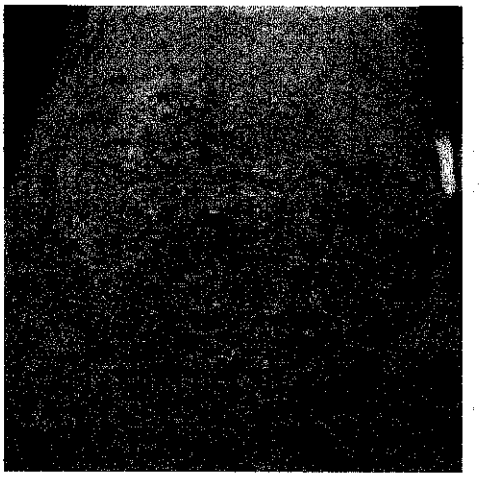


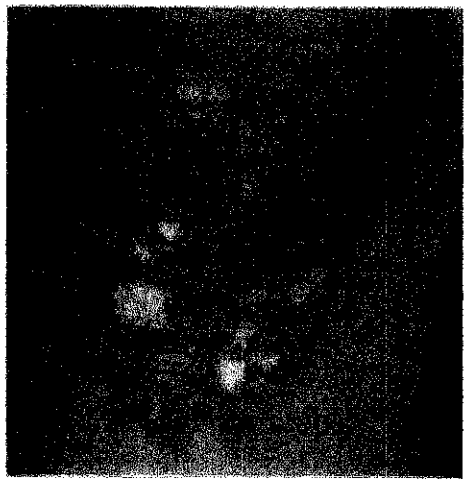


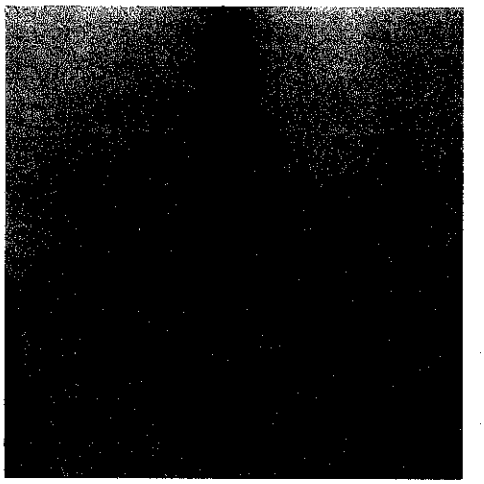


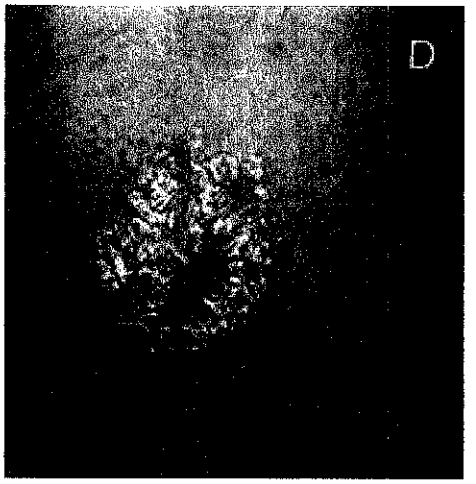


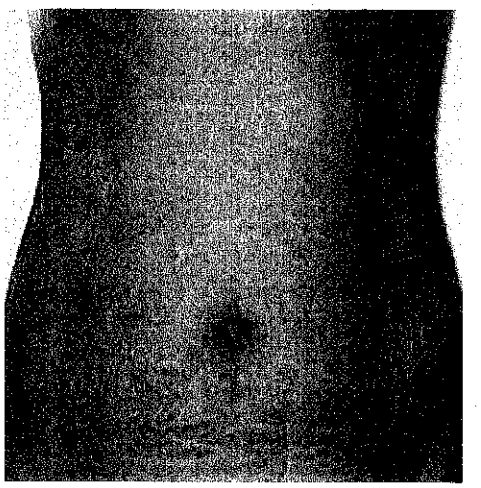


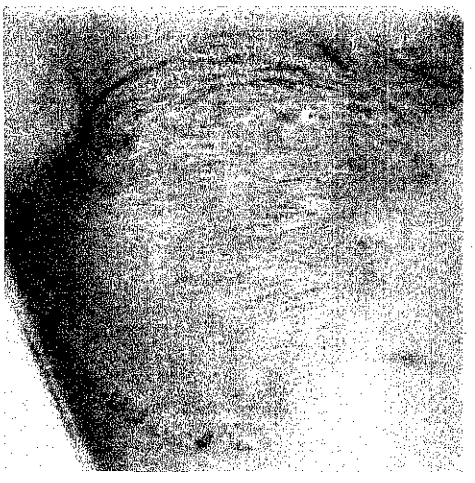


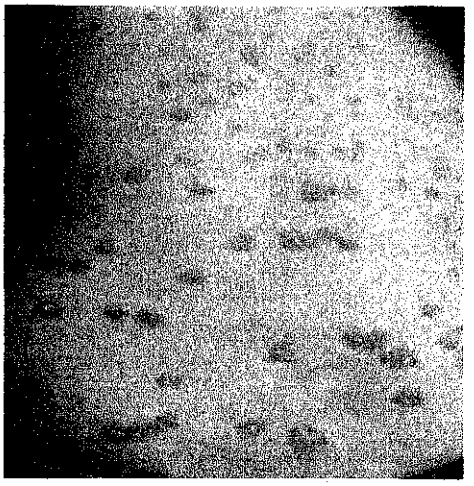


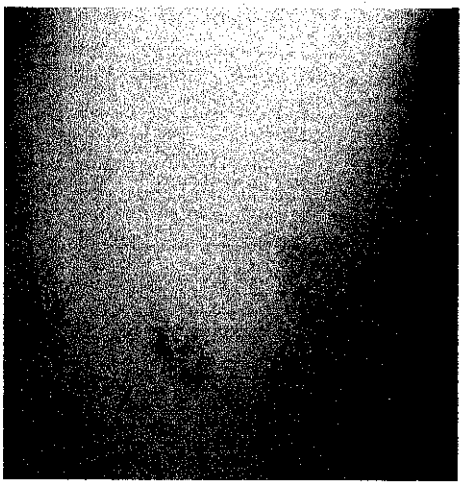


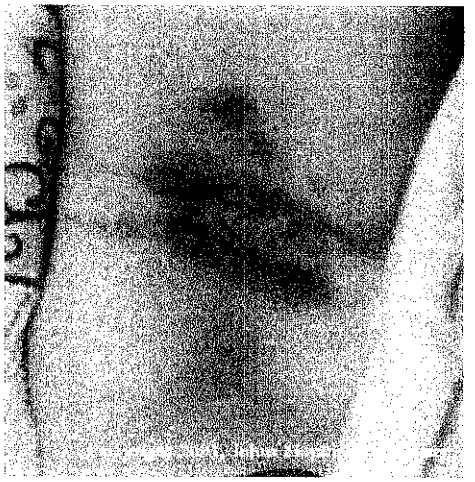

D

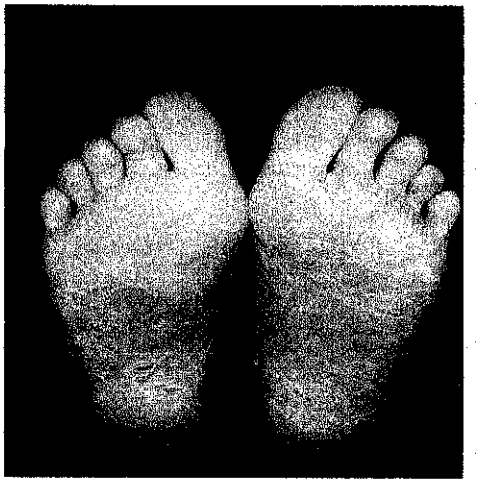


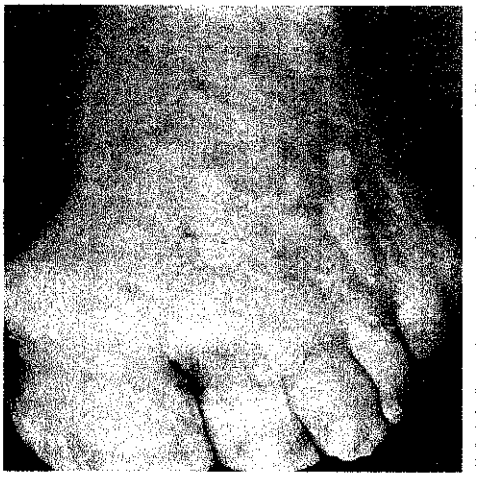


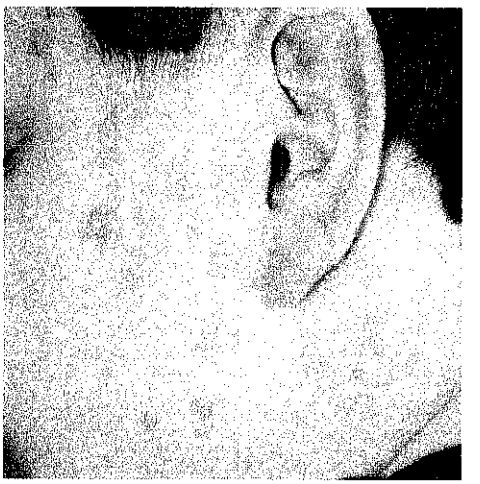


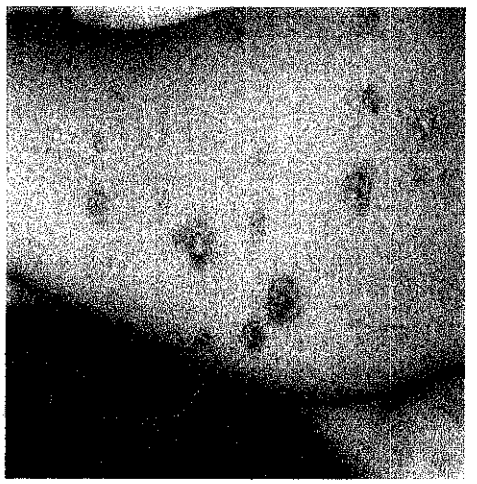


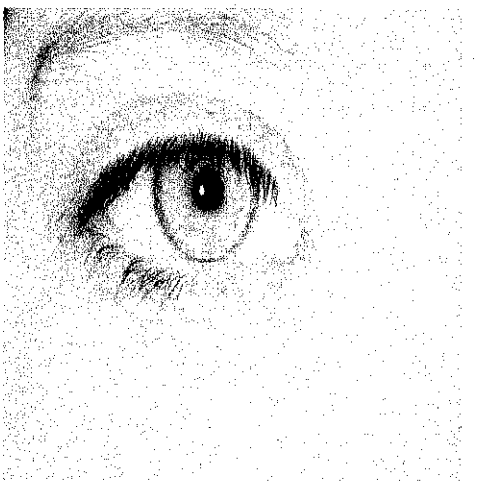


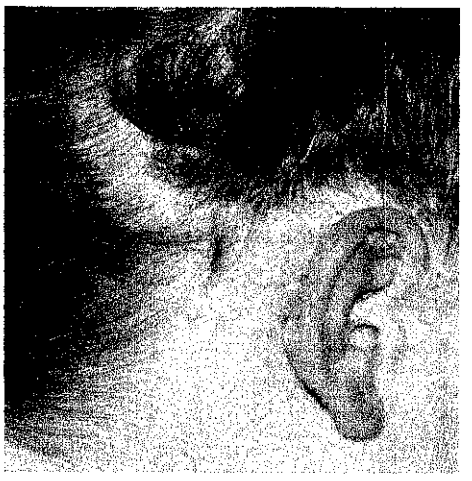


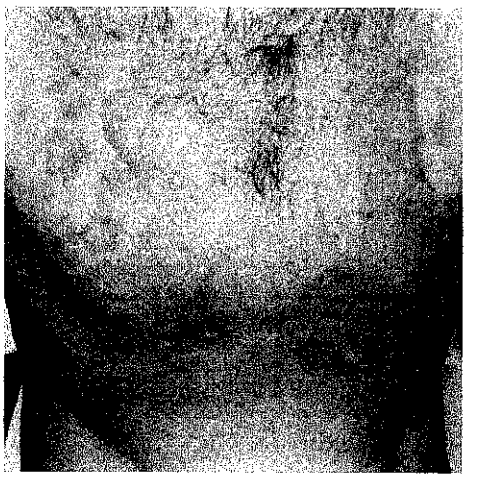


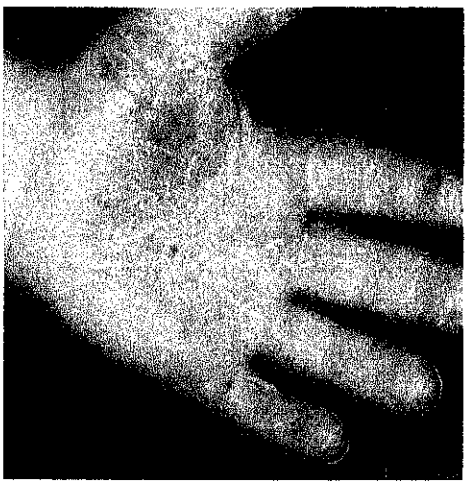


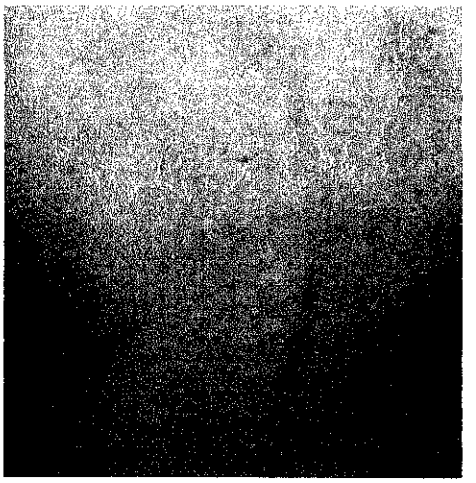


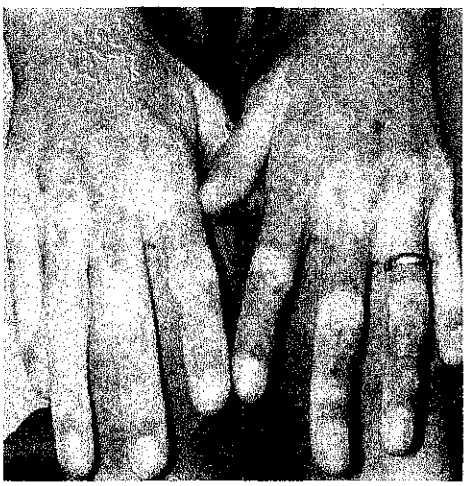


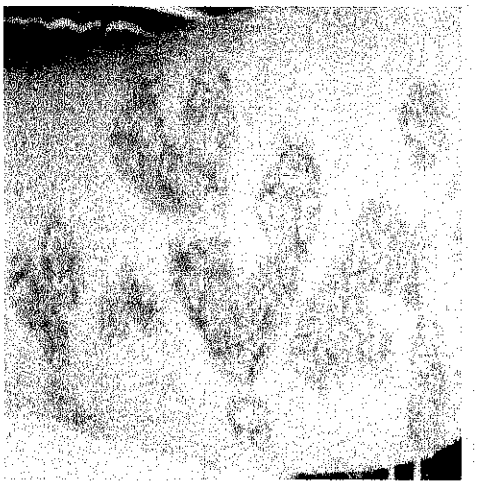


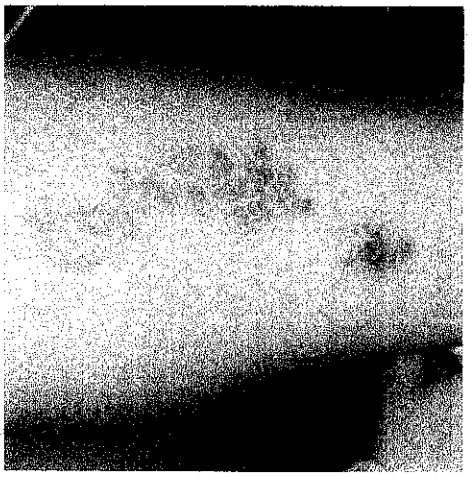


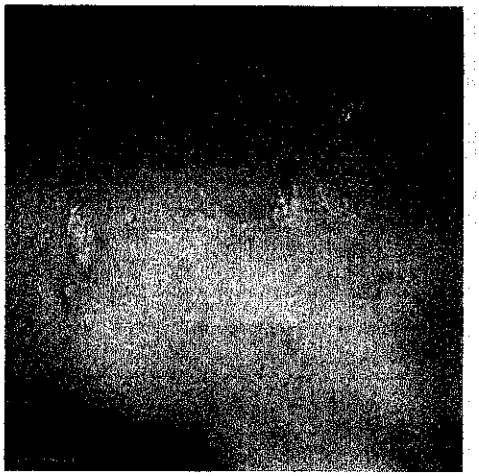




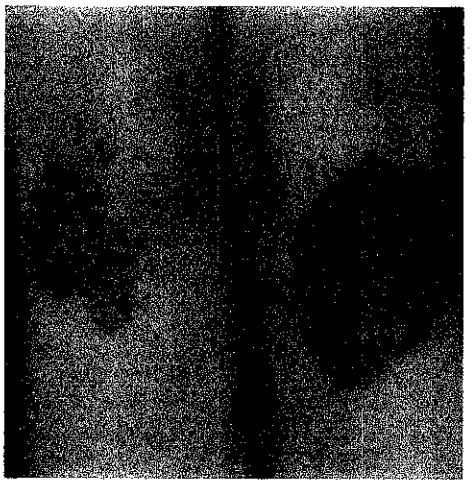


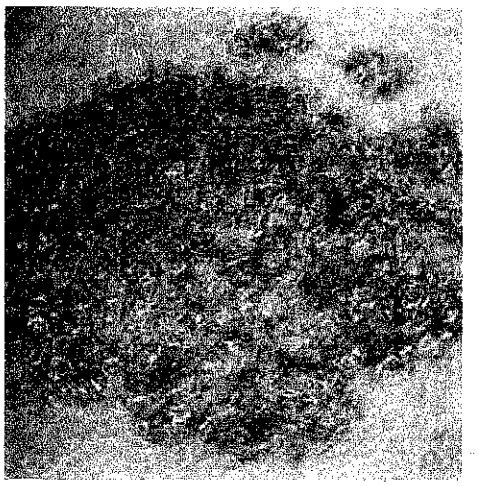


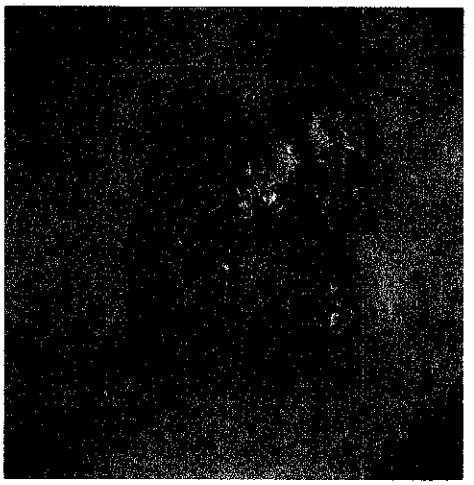


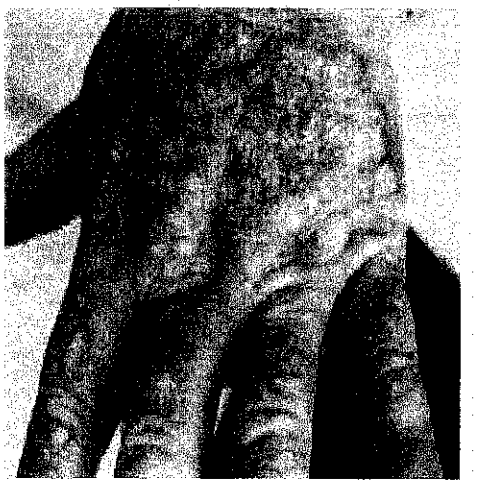


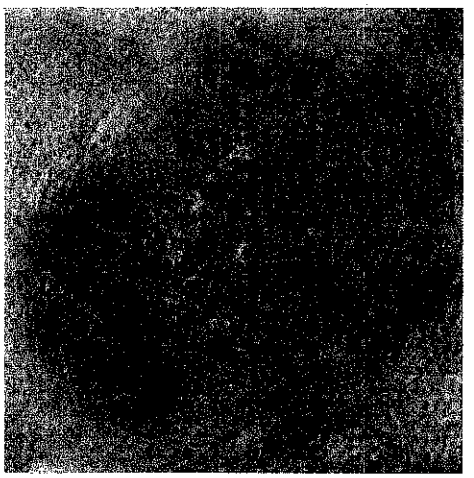
REDNESS SCORE = 2

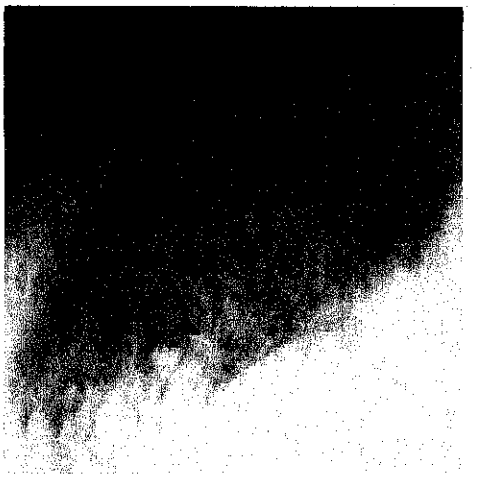


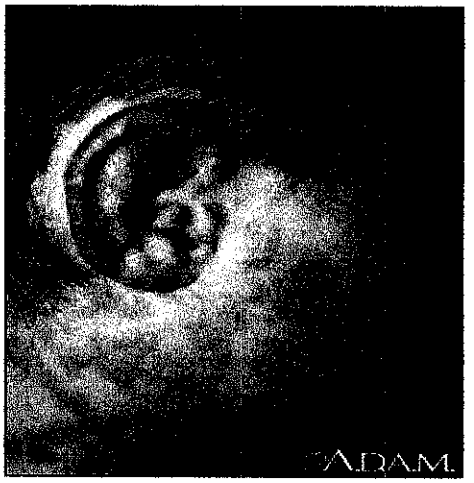


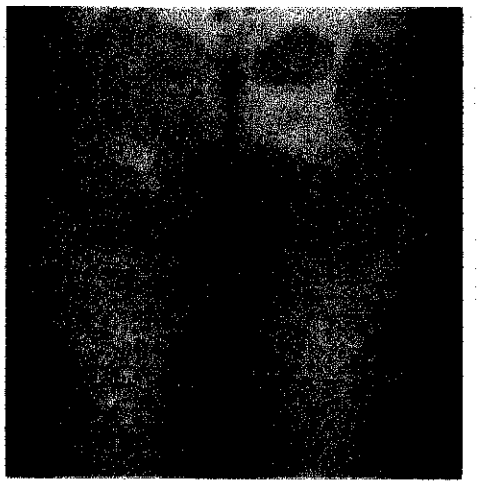


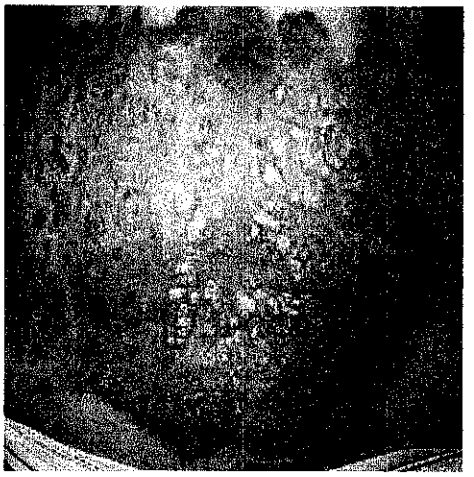


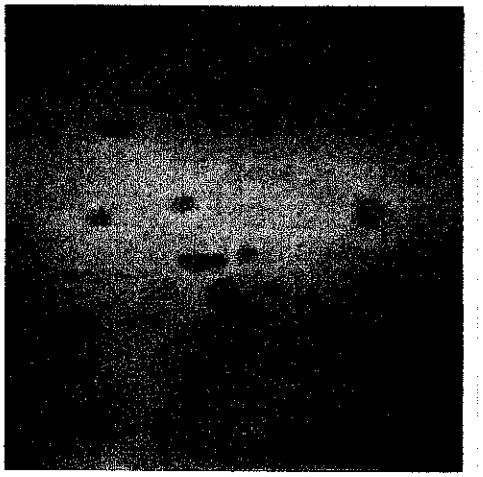


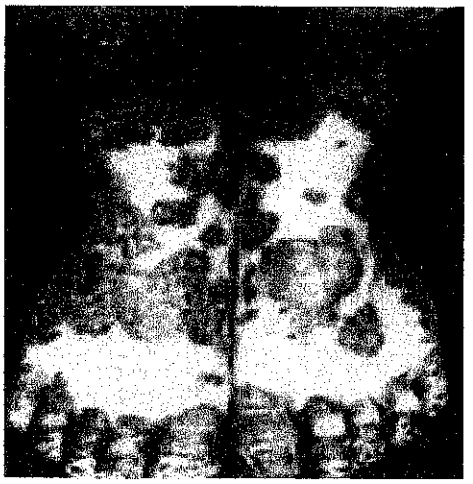


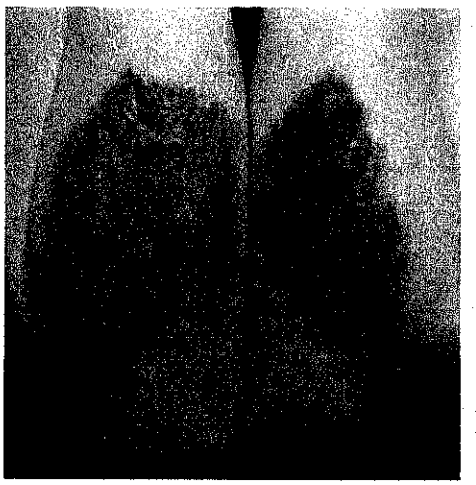

A.D.A.M.

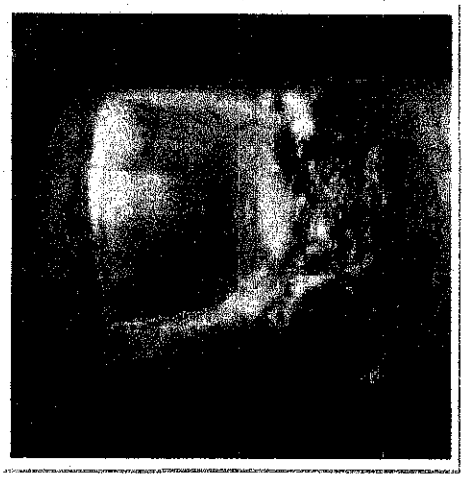


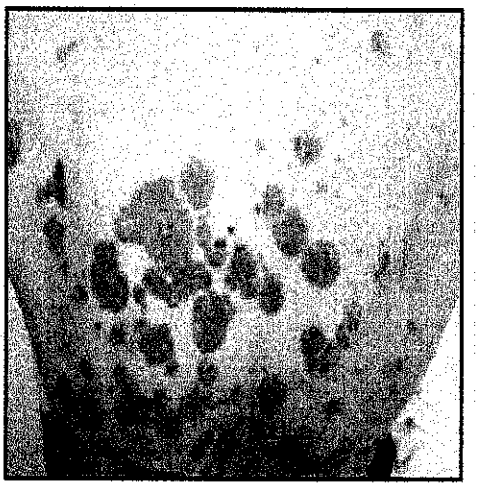


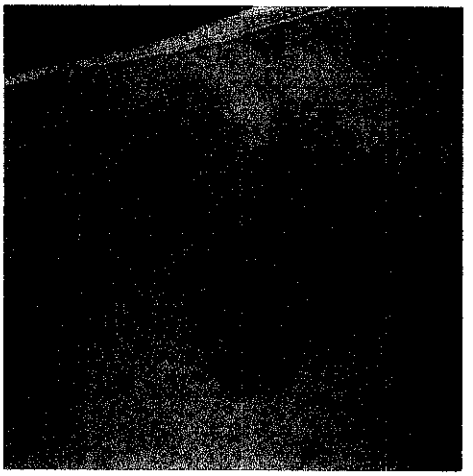


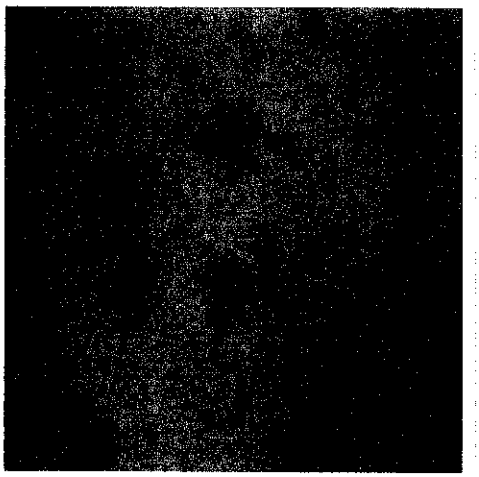


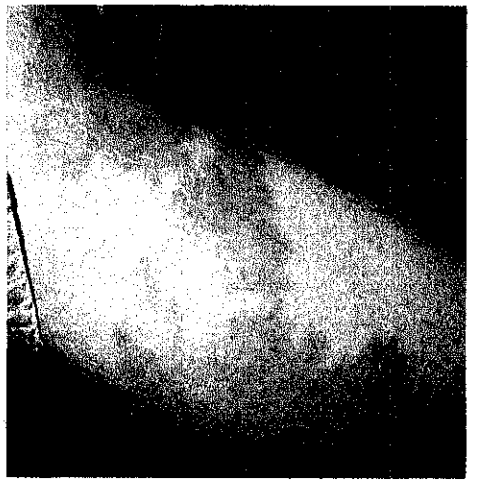


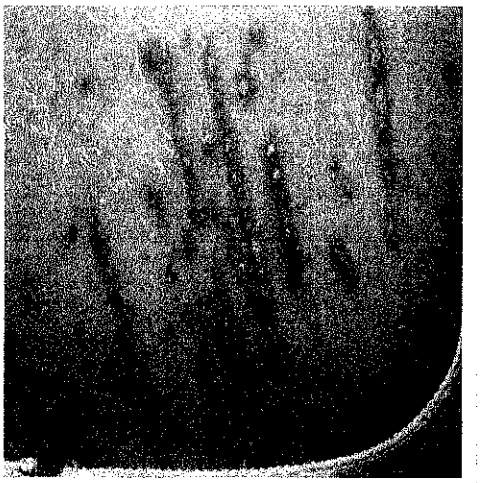


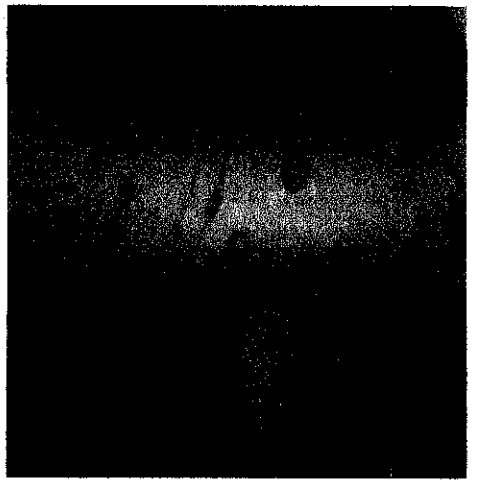


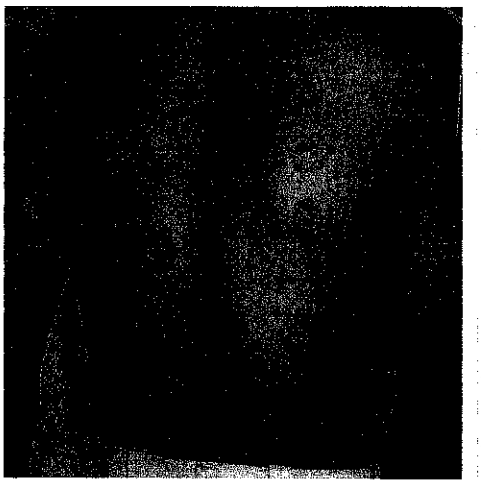


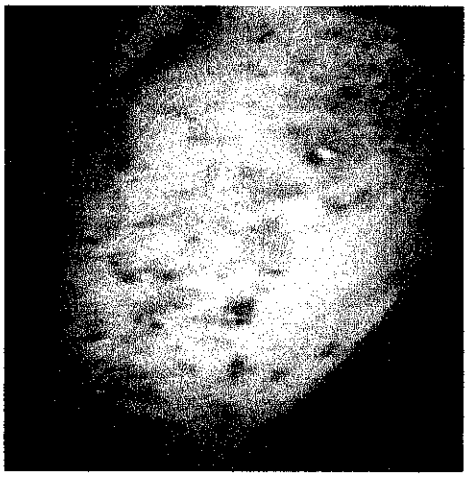


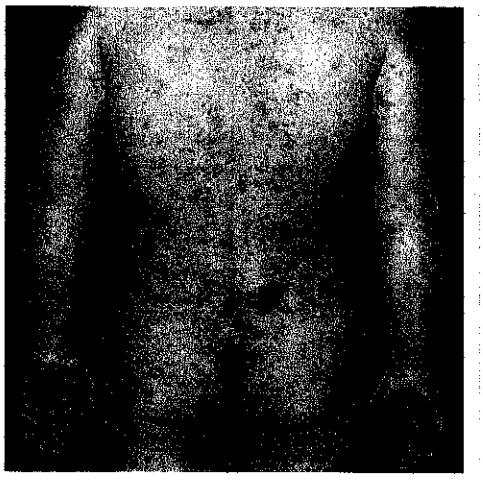


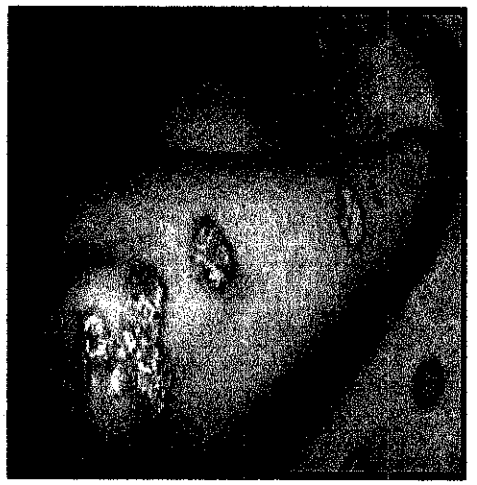


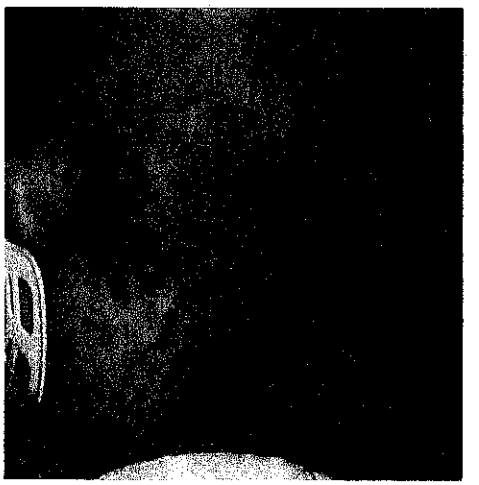


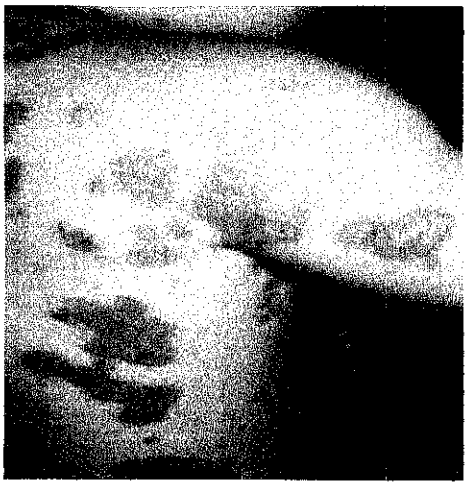


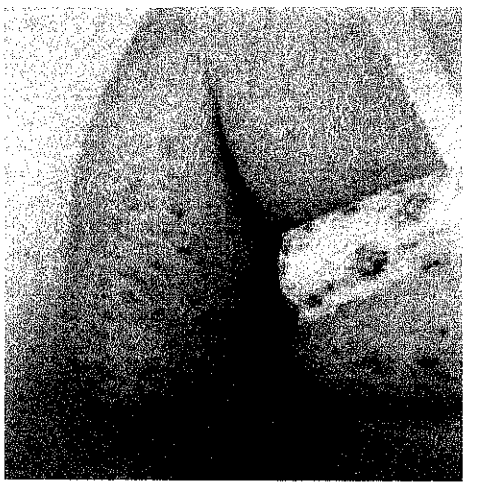


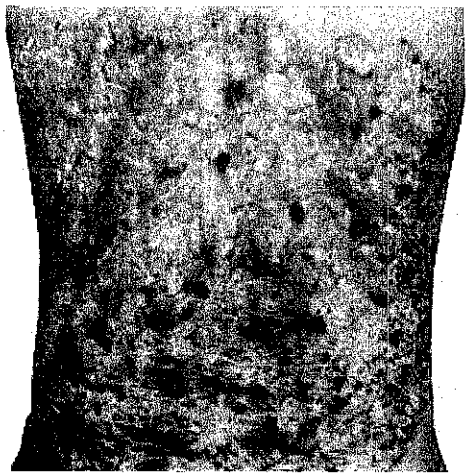


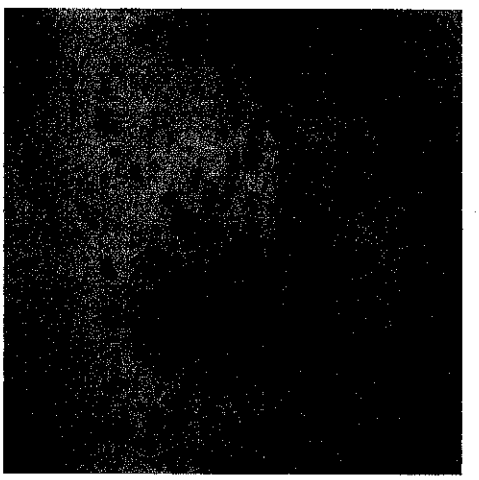


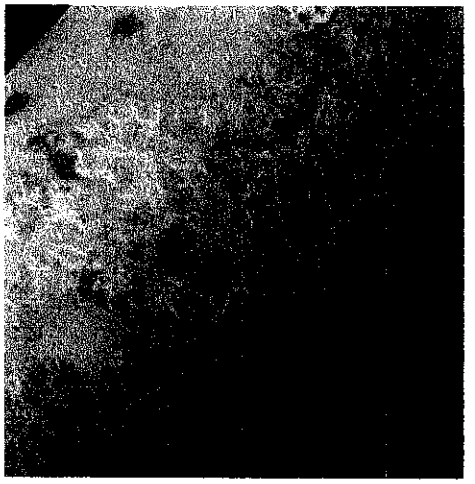


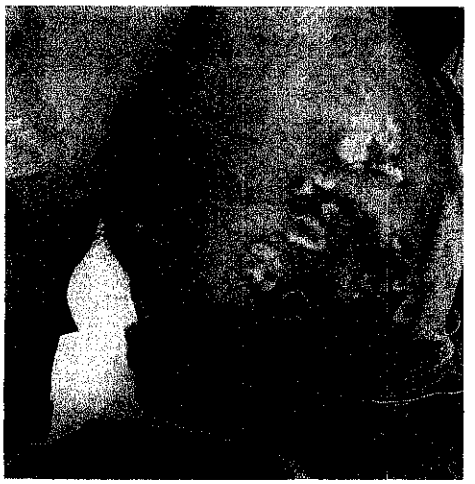


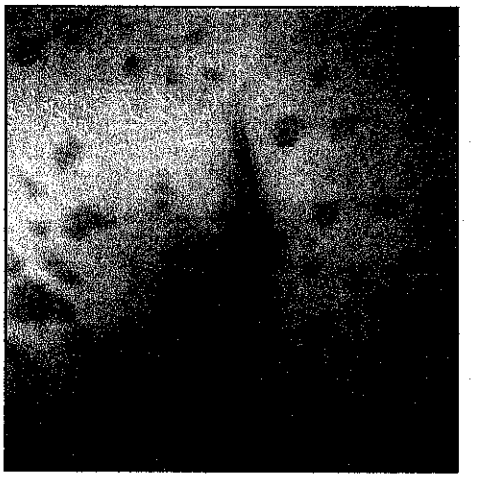




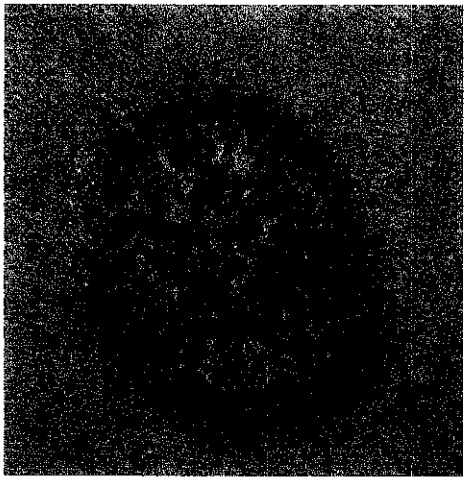
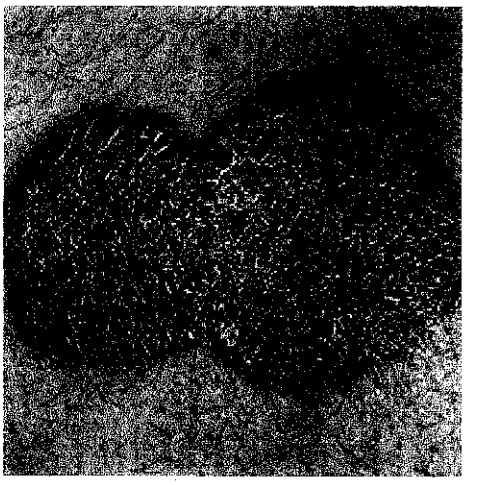
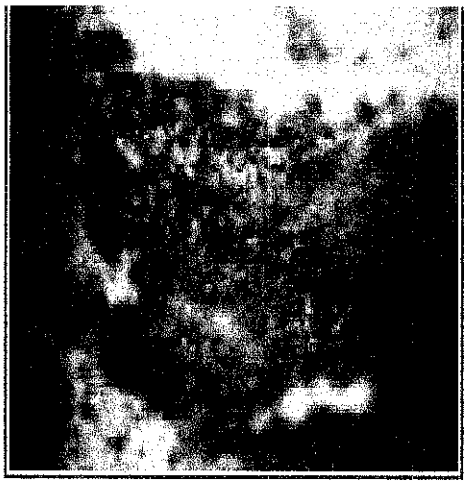
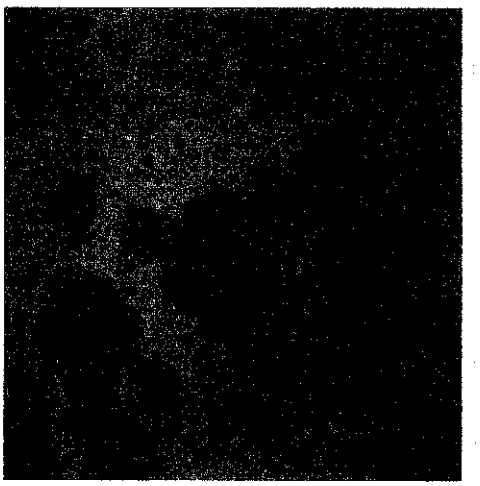


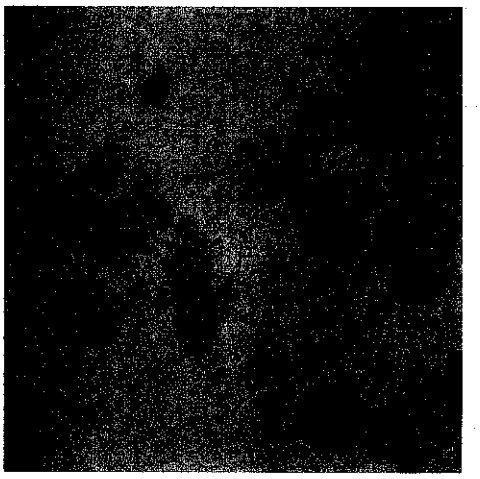


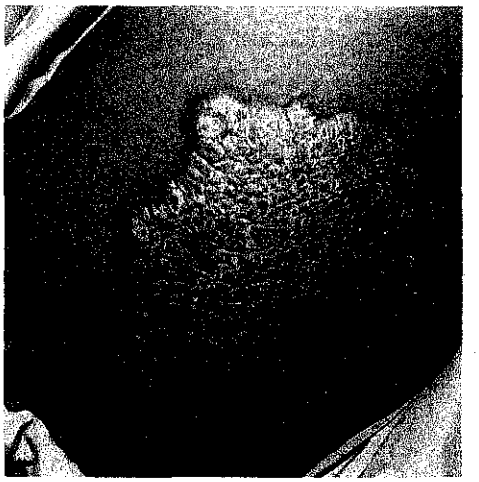


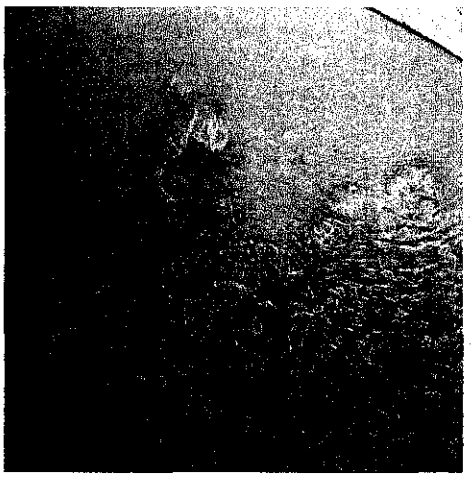


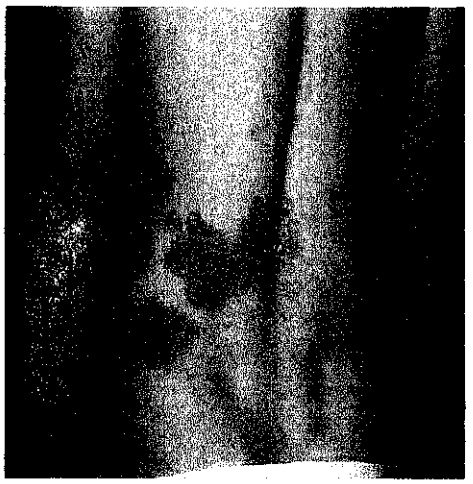
REDNESS SCORE = 3

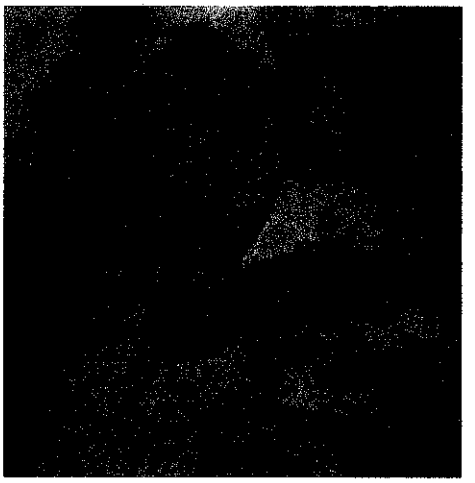





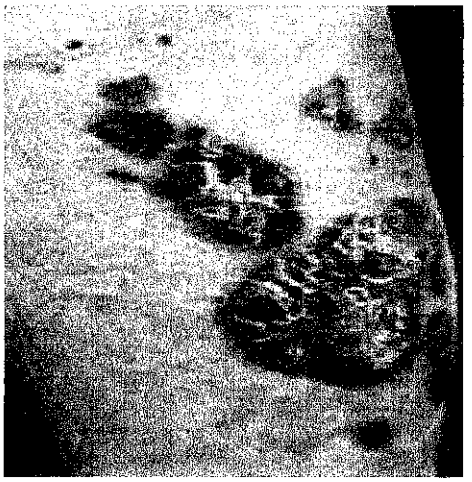


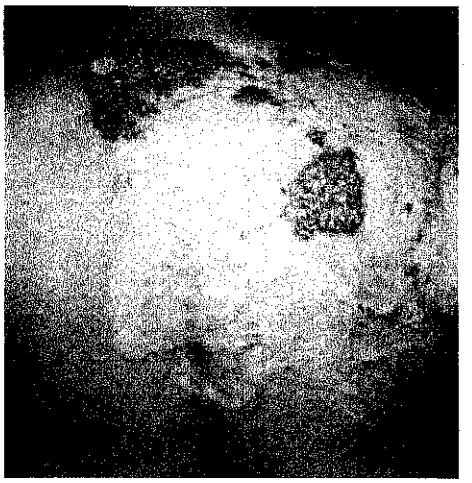


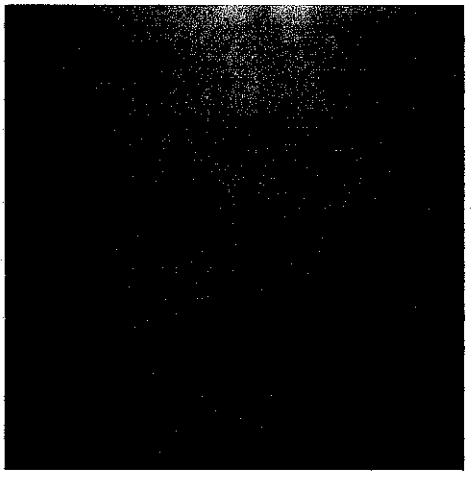


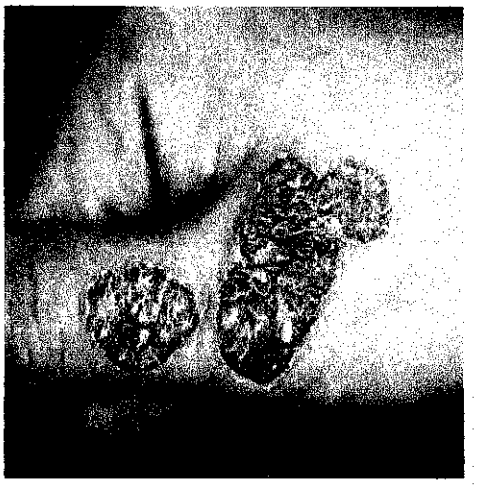


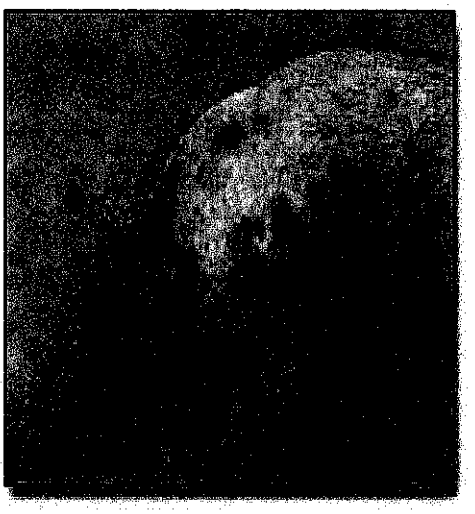


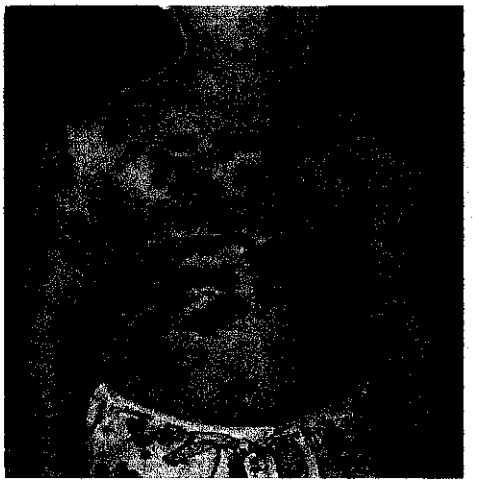


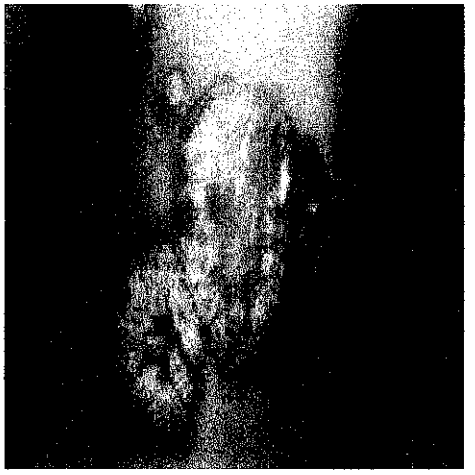


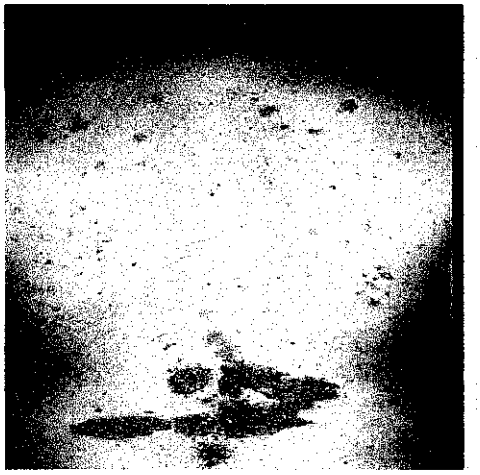


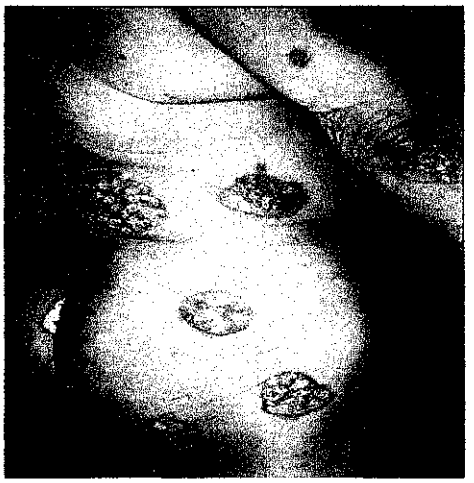


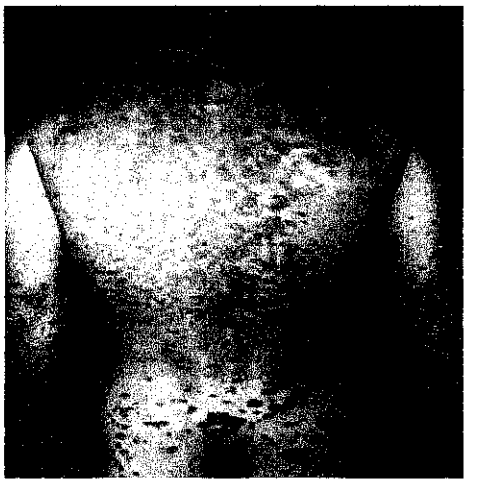


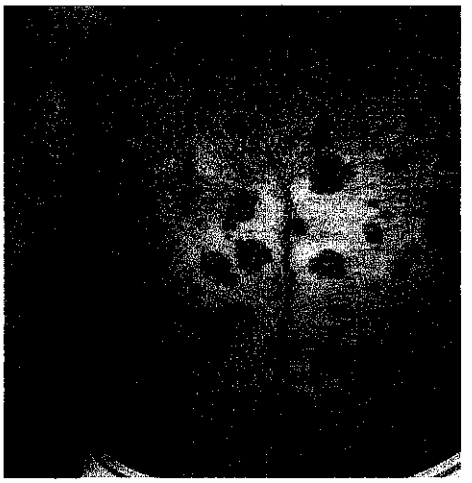


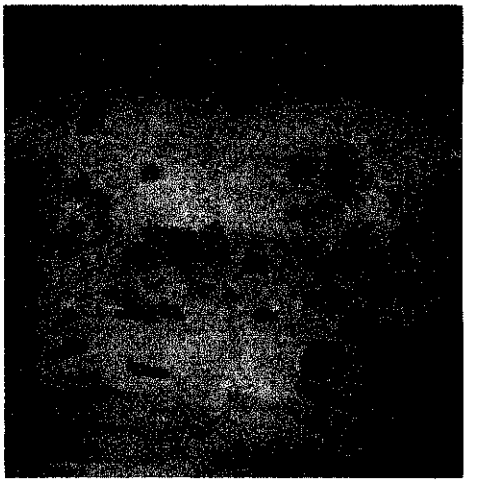


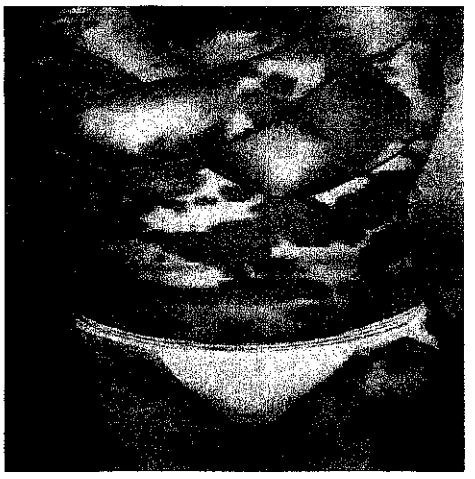


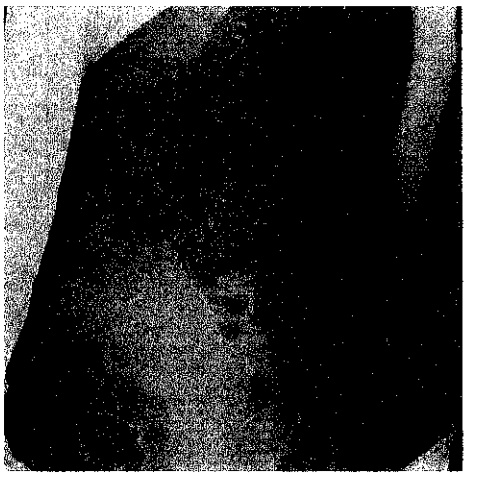




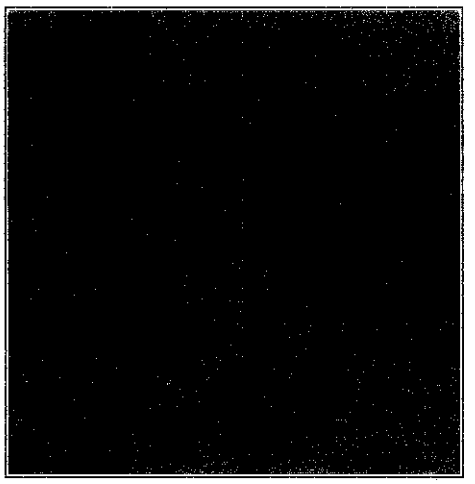
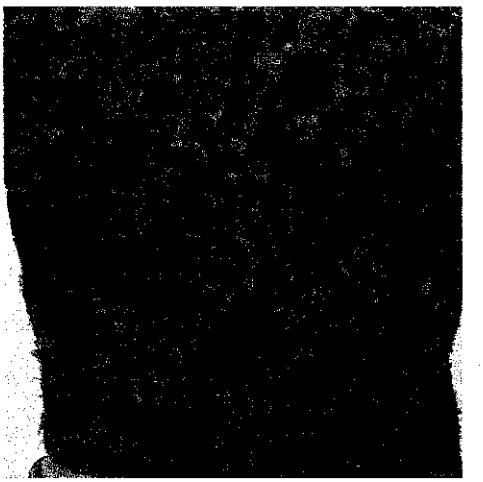
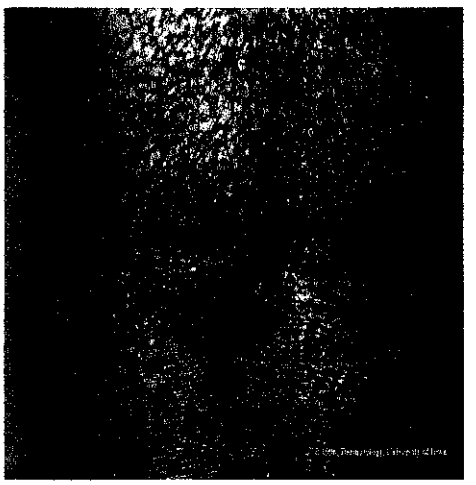
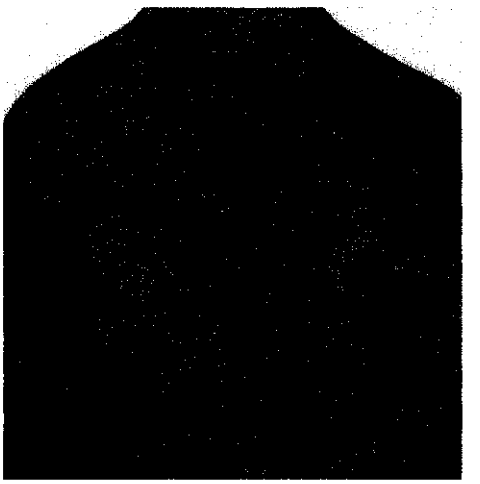
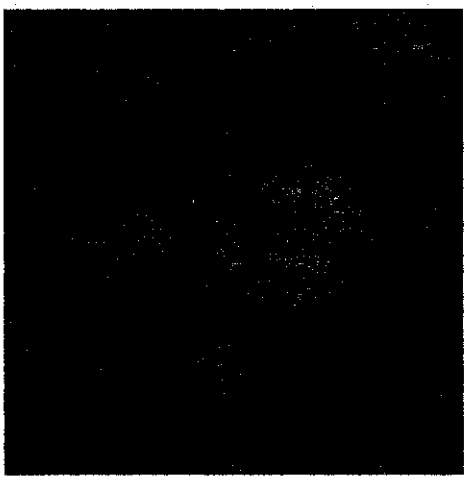
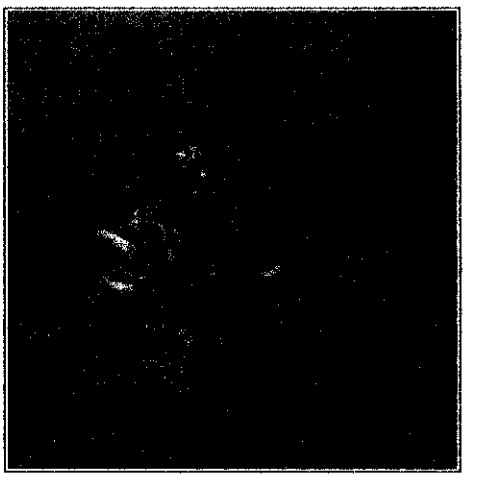


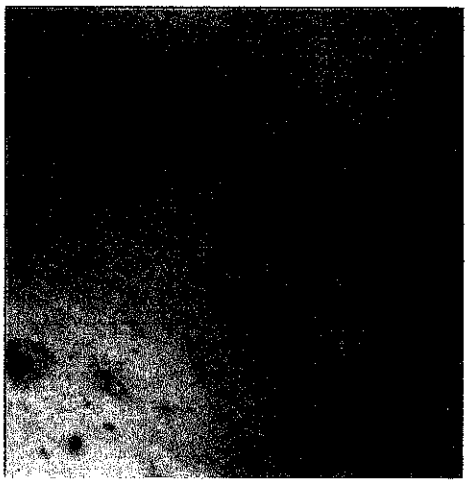


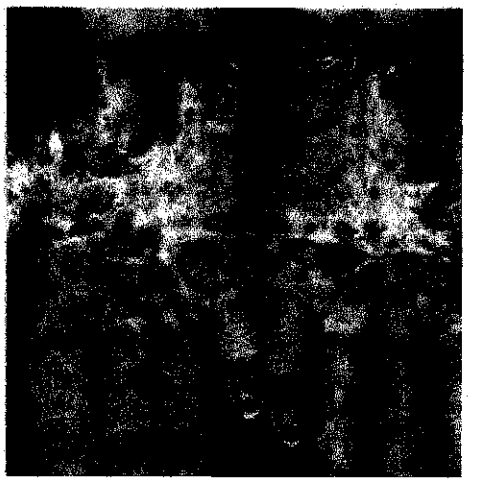


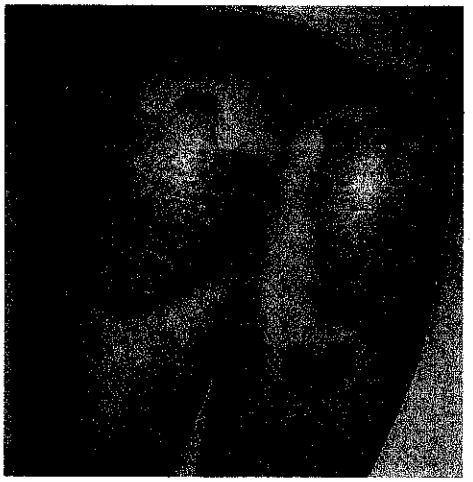


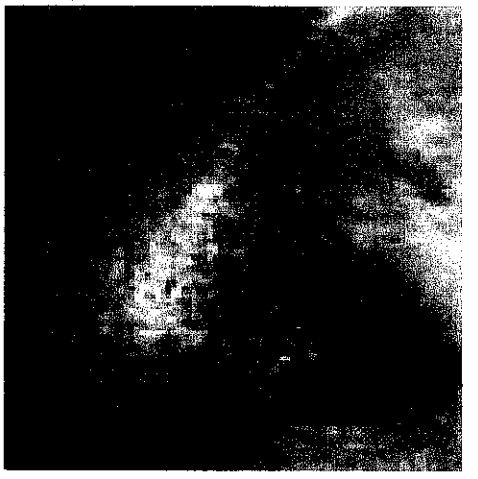
REDNESS SCORE = 4

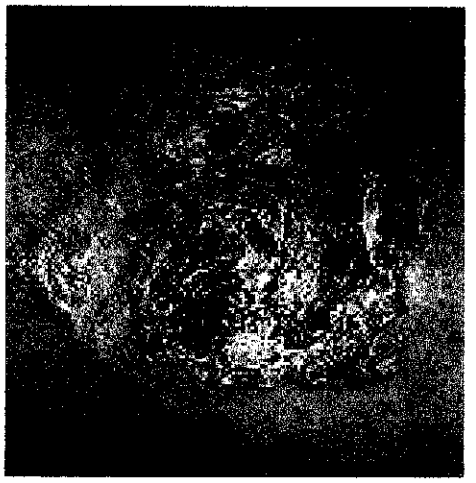







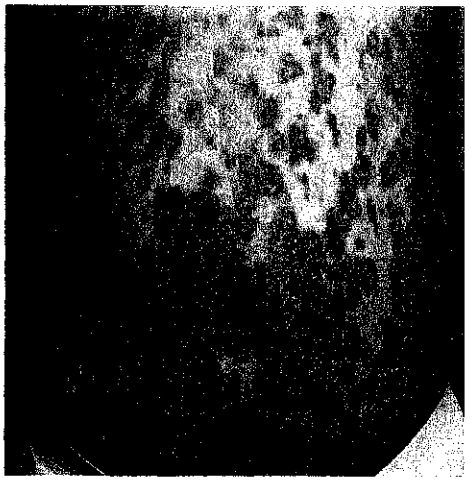


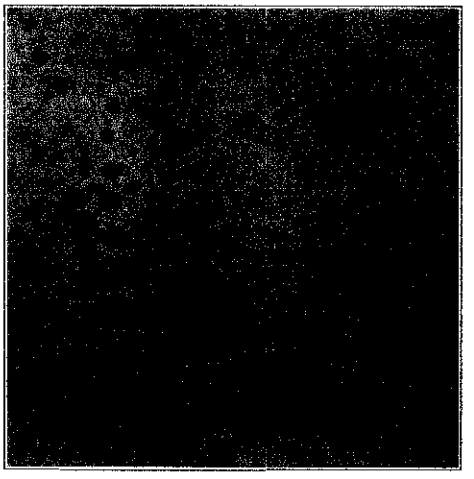


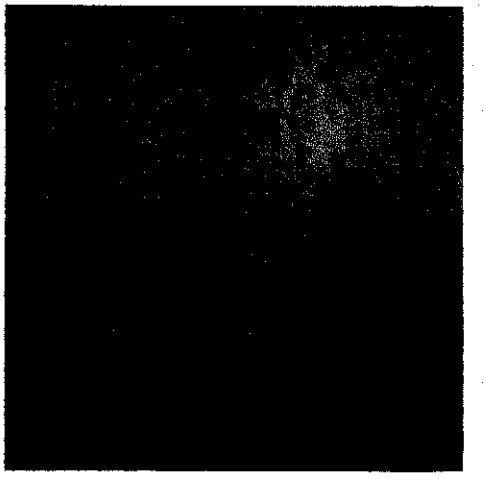


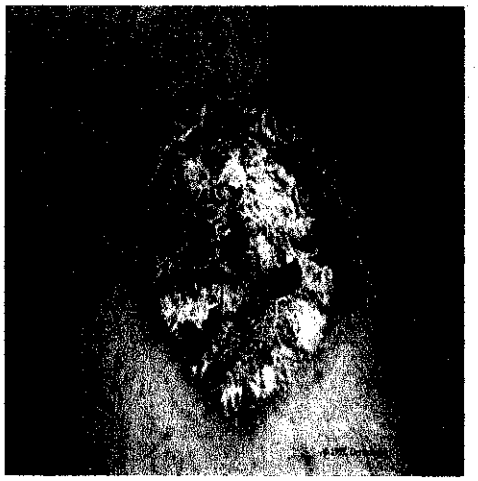


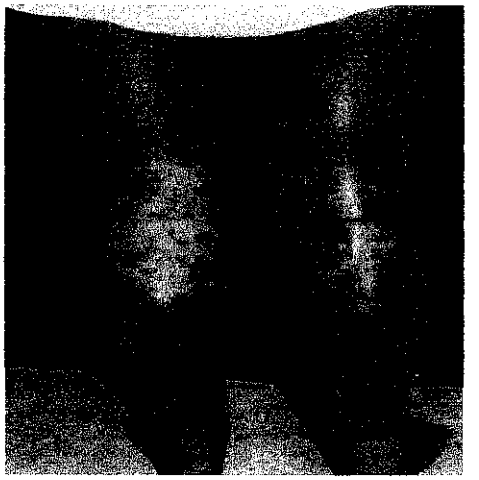


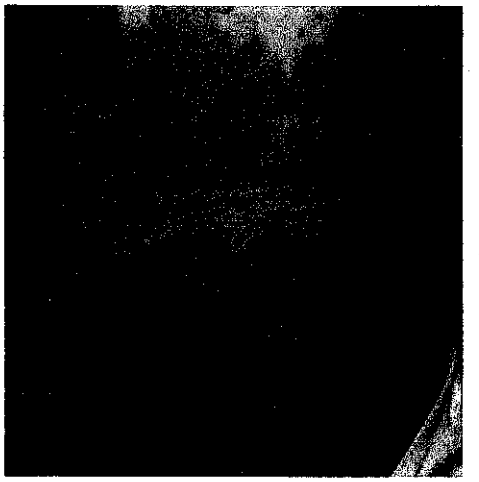


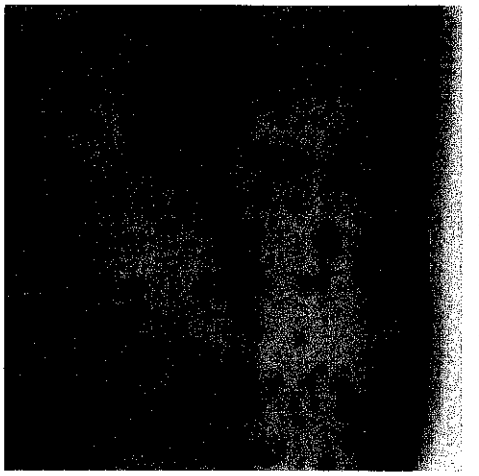


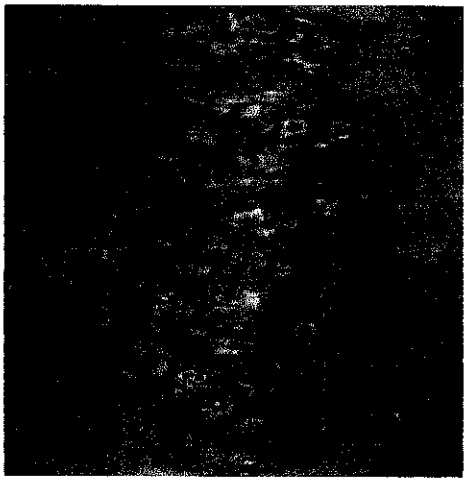


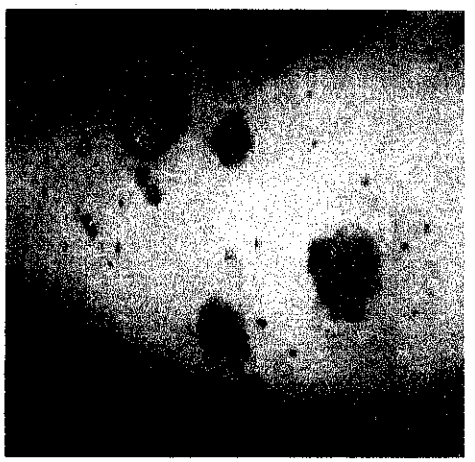


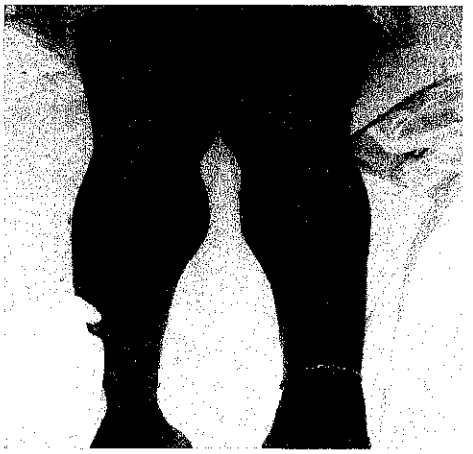












APPENDIX C

SOURCE CODE

```
function varargout = score(varargin)
% SCORE M-file for score.fig
%      SCORE, by itself, creates a new SCORE or raises the existing
%      singleton*.
%
%      H = SCORE returns the handle to a new SCORE or the handle to
%      the existing singleton*.
%
%      SCORE('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in SCORE.M with the given input arguments.
%
%      SCORE('Property','Value',...) creates a new SCORE or raises the
%      existing singleton*.  Starting from the left, property value pairs↙
are
%      applied to the GUI before score_OpeningFunction gets called.  An
%      unrecognized property name or invalid value makes property↙
application
%      stop.  All inputs are passed to score_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only↙
one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Copyright 2002-2003 The MathWorks, Inc.

% Edit the above text to modify the response to help score

% Last Modified by GUIDE v2.5 12-May-2006 01:13:31

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @score_OpeningFcn, ...
                   'gui_OutputFcn',  @score_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```
% --- Executes just before score is made visible.
function score_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to score (see VARARGIN)

% Choose default command line output for score
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes score wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = score_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


%-----------------------------------------------------------------
%           FUNCTION TO DISPLAY ORIGINAL IMAGE
%-----------------------------------------------------------------

% --- Executes on button press in original.
function original_Callback(hObject, eventdata, handles)
% hObject    handle to original (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% load psoriasis image from the graphics file
[filename, pathname] = uigetfile('*jpg');
img = [filename];
i = imread(img);

% display the original image
subplot(2,1,1),imshow(i),title('Original Image','fontsize',15,...
    'fontweight','bold','Color','r');

% save the image
handles.metricdata.i = i;
guidata(hObject,handles);
```

```
%--------------------------------------------------------------------
%            FUNCTION TO DISPLAY SEGMENTED IMAGE
%--------------------------------------------------------------------

% --- Executes on button press in segmented.
function segmented_Callback(hObject, eventdata, handles)
% hObject    handle to segmented (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% read the saved psoriasis image
I = handles.metricdata.i;

% image enhancement
p = imadjust(I,stretchlim(I),[]);

% Convert image from RGB color space to L*a*b* color space. The difference
% between two colors can be measured using the Euclidean distance metric.
% Makecform is used to create a color transformation structure, cform,↙
that
% defines the color space conversion specified by type. To perform the
% transformation, pass the color transformation structure as an argument
% to the applycform function
cform = makecform('srgb2lab'); %create a color transformation structure
lab_p = applycform(p,cform); %apply the color transformation structure

% Classify the colors in L*a*b* color space using K-means clustering
ab = double(lab_p(:,:,2:3)); %convert to double precision
nrows = size(ab,1); %size of array
ncols = size(ab,2); %size of array
ab = reshape(ab,nrows*ncols,2); %change the size
nColors = 3; %number of clusters

% Repeat the clustering four times to avoid local minima.
[cluster_idx cluster_center] = kmeans(ab,nColors,'distance',...
    'sqEuclidean','Replicates',3);

% For every object in the input, kmeans returns an index corresponding to↙
a cluster.
% Label every pixel in the image with its cluster_index.
pixel_labels = reshape(cluster_idx,nrows,ncols);

% Using pixel_labels, the objects in psoriasis.jpg can be separated by↙
color,
% which will result in three images.
segmented_images = cell(1,3); %create cell array
rgb_label = repmat(pixel_labels,[1 1 3]); %replicate and tile an array
for k = 1:nColors
color = p;
color(rgb_label ~= k) = 0;
segmented_images{k} = color;
```

```
end

% display the different images in the three different clusters
subplot(2,3,1), imshow(segmented_images{1}), title('Cluster 1',...
    'fontsize',15,'fontweight','bold','Color','r');
subplot(2,3,2), imshow(segmented_images{2}), title('Cluster 2',...
    'fontsize',15,'fontweight','bold','Color','r');
subplot(2,3,3), imshow(segmented_images{3}), title('Cluster 3',...
    'fontsize',15,'fontweight','bold','Color','r');

cluster1 = segmented_images{1};
cluster2 = segmented_images{2};
cluster3 = segmented_images{3};

% save the images in the three different clusters
handles.metricdata.cluster1 = cluster1;
handles.metricdata.cluster2 = cluster2;
handles.metricdata.cluster3 = cluster3;
guidata(hObject,handles)


%--------------------------------------------------------------------
%           FUNCTION TO DISPLAY LESION IMAGE
%--------------------------------------------------------------------

% --- Executes on button press in cluster1.
function cluster1_Callback(hObject, eventdata, handles)
% hObject    handle to cluster1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of cluster1

% read the lesion image if the image is in cluster 1
lesion = handles.metricdata.cluster1;

% display the lesion image
subplot(2,1,1),imshow(lesion),title('Lesion Image',...
    'fontsize',15,'fontweight','bold','Color','r');

% save the lesion image
handles.metricdata.lesion = lesion;
guidata(hObject,handles);


% --- Executes on button press in cluster2.
function cluster2_Callback(hObject, eventdata, handles)
% hObject    handle to cluster2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of cluster2
```

```
% read the lesion image if the image is in cluster 2
lesion = handles.metricdata.cluster2;

% display the lesion image
subplot(2,1,1),imshow(lesion),title('Lesion Image',...
    'fontsize',15,'fontweight','bold','Color','r');

% save the lesion image
handles.metricdata.lesion = lesion;
guidata(hObject,handles);


% --- Executes on button press in cluster3.
function cluster3_Callback(hObject, eventdata, handles)
% hObject    handle to cluster3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of cluster3

% read the lesion image if the image is in cluster 3
lesion = handles.metricdata.cluster3;

% display the lesion image
subplot(2,1,1),imshow(lesion),title('Lesion Image',...
    'fontsize',15,'fontweight','bold','Color','r');

% save the lesion image
handles.metricdata.lesion = lesion;
guidata(hObject,handles);


%-------------------------------------------------------------
%           FUNCTION TO SCORE THE REDNESS
%-------------------------------------------------------------

% --- Executes on button press in redness.
function redness_Callback(hObject, eventdata, handles)
% hObject    handle to redness (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% read the saved lesion image
image = handles.metricdata.lesion;

% convert the lesion image to grayscale
g_image = rgb2gray(image);

% analyse the lesion data from the image and histogram
[y x] = imhist(g_image); % set the x-axis and y-axis
N = y(x>0); % identify the size of the image
```

```
bw = roicolor(g_image,10,255); % select the region of interest
M = bwarea(bw); % find out the total pixels of the lesion image
b = (1:255); % set the range of intensity

% determine the mean of the lesion image
num = N;
den = M;
p = [N/M];
mean = b*p;

v0 = 191.9326; % reference value for score 0
v1 = 152.2451; % reference value for score 1
v2 = 130.3933; % reference value for score 2
v3 = 103.9350; % reference value for score 3
v4 = 87.1560;  % reference value for score 4

% set the threshold values
T1 = (v0+v1)/2;
T2 = (v1+v2)/2;
T3 = (v2+v3)/2;
T4 = (v3+v4)/2;

% display the redness score
if (mean >= T1)
    set(handles.redness_score,'String','0');
else if (mean < T1 && mean >= T2)
        set(handles.redness_score,'String','1');
    else if (mean < T2 && mean >= T3)
            set(handles.redness_score,'String','2');
        else if (mean < T3 && mean >= T4)
                set(handles.redness_score,'String','3');
            else if (mean < T4)
                    set(handles.redness_score,'String','4');
                end
            end
        end
    end
end


%-----------------------------------------------------------------
%           FUNCTION TO SCORE THE AREA
%-----------------------------------------------------------------

% calculate the area of the lesion image
im_area = M/1444; % total pixels divide by pixels value for 1cm2 ↙
(reference)

% save the formula for area of the lesion
handles.metricdata.im_area = im_area;
guidata(hObject,handles)
```

```
% --- Executes during object creation, after setting all properties.
function scale_CreateFcn(hObject, eventdata, handles)
% hObject    handle to scale (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get↙
(0,'defaultUicontrolBackgroundColor'));
end

function scale_Callback(hObject, eventdata, handles)
% hObject    handle to scale (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of scale as text
%        str2double(get(hObject,'String')) returns contents of scale as a↙
double

scale = str2double(get(hObject,'String'));
if isnan(scale)
    set(hObject,'String',0);
    errordlg('Input must be a number','Error');
end

% save the new scale value
handles.metricdata.scale = scale;
guidata(hObject,handles)


% --- Executes during object creation, after setting all properties.
function height_CreateFcn(hObject, eventdata, handles)
% hObject    handle to height (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get↙
(0,'defaultUicontrolBackgroundColor'));
end

function height_Callback(hObject, eventdata, handles)
```

```
% hObject    handle to height (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of height as text
%        str2double(get(hObject,'String')) returns contents of height as a↙
double

height = str2double(get(hObject,'String'));
if isnan(height)
    set(hObject,'String',0);
    errordlg('Input must be a number','Error');
end

% save the new height value
handles.metricdata.height = height;
guidata(hObject,handles)


% --- Executes during object creation, after setting all properties.
function weight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to weight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get↙
(0,'defaultUicontrolBackgroundColor'));
end

function weight_Callback(hObject, eventdata, handles)
% hObject    handle to weight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of weight as text
%        str2double(get(hObject,'String')) returns contents of weight as a↙
double

weight = str2double(get(hObject,'String'));
if isnan(weight)
    set(hObject,'String'),0;
    errordlg('Input must be a number','Error');
end

% save the new weight value
handles.metricdata.weight = weight;
guidata(hObject,handles)
```

```
% calculate area of the body covered by lesion
b_area = handles.metricdata.im_area * handles.metricdata.scale;
handles.metricdata.b_area = b_area;
guidata(hObject,handles)

%calculate body surface area (AREA) in m2 and cm2
BSAm = ((handles.metricdata.height*handles.metricdata.weight)/3600)^0.5;
BSA = BSAm*10000; %convert the bsa from m2 to cm2

% save the new AREA value
handles.metricdata.BSA = BSA;
guidata(hObject,handles)


% --- Executes on button press in head.
function head_Callback(hObject, eventdata, handles)
% hObject    handle to head (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of head

% calculate and save percent area of lesion on the head
p_area = (handles.metricdata.b_area/(0.1*handles.metricdata.BSA))*100;
handles.metricdata.p_area = p_area;
guidata(hObject,handles)


% --- Executes on button press in up_ext.
function up_ext_Callback(hObject, eventdata, handles)
% hObject    handle to up_ext (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of up_ext

% calculate and save percent area of lesion on the upper extremities
p_area = (handles.metricdata.b_area/(0.2*handles.metricdata.BSA))*100;
handles.metricdata.p_area = p_area;
guidata(hObject,handles)


% --- Executes on button press in trunk.
function trunk_Callback(hObject, eventdata, handles)
% hObject    handle to trunk (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of trunk
```

```
% calculate and save percent area of lesion on the trunk
p_area = (handles.metricdata.b_area/(0.3*handles.metricdata.BSA))*100;
handles.metricdata.p_area = p_area;
guidata(hObject,handles)


% --- Executes on button press in low_ext.
function low_ext_Callback(hObject, eventdata, handles)
% hObject    handle to low_ext (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of low_ext

% calculate and save percent area of lesion on the lower extremities
p_area = (handles.metricdata.b_area/(0.4*handles.metricdata.BSA))*100;
handles.metricdata.p_area = p_area;
guidata(hObject,handles)


% --- Executes on button press in area.
function area_Callback(hObject, eventdata, handles)
% hObject    handle to area (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% read the saved percent area of lesion
percent = handles.metricdata.p_area;

% display the area score
if (percent == 0)
set(handles.area_score,'String','0');
else if (percent > 0 && percent < 10)
        set(handles.area_score,'String','1');
    else if (percent >= 10 && percent < 30)
            set(handles.area_score,'String','2');
        else if (percent >= 30 && percent < 50)
                set(handles.area_score,'String','3');
            else if (percent >= 50 && percent < 70)
                    set(handles.area_score,'String','4');
                else if (percent >= 70 && percent < 90)
                        set(handles.area_score,'String','5');
                    else if (percent >= 90 && percent <= 100)
                            set(handles.area_score,'String','6');
                        end
                    end
                end
            end
        end
    end
end
```

```
% --- Executes on button press in reset.
function reset_Callback(hObject, eventdata, handles)
% hObject    handle to reset (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

initialize_gui(gcbf,handles,true);

function initialize_gui(fig_handle,handles,isreset)
% If the metricdata field is present and the reset flag is false, it means
% that we just re-initializing a GUI by calling it from the cmd line while
% it is up. So, bail out as we do not want to reset the data.

if isfield(handles,'metricdata') && ~isreset
    return;
end

set(handles.redness_score, 'String','');
set(handles.area_score, 'String','');
```

APPENDIX D

GANTT CHART

Gantt Chart for Final Year Project 1

No.	Detail / Week	1	2	3	4	5	6	7	8	9	10	11	12	13
1	Selection of Project Topics													
2	Preliminary Research Work													
3	Submission of Preliminary Report													
4	Brainstorming and Planning													
5	Research, Literature, and Software Learning													
6	Submission of Progress Report													
7	Learning and Training the Software													
8	Submission of Final Draft													
9	Submission of Interim Report													
10	Oral Presentation													

Gantt Chart for Final Year Project 2

No.	Detail / Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	Project Work	■	■	■	■	■	■	■	■	■	■	■	■									
2	Submission of Progress Report 1				■																	
3	Submission of Progress Report 2								■													
4	Submission of Draft Report													■								
5	Submission of Final Report (Soft Cover)															■						
6	Oral Presentations																			■		
7	Submission of Technical Report																				■	
8	Submission of Final Report (Hard Cover)																					■