

2007

A comparison of edge detection methods for segmentation of skin lesions in mobile-phone-quality images

Khanh Linh Hua
Edith Cowan University

Follow this and additional works at: https://ro.ecu.edu.au/theses_hons



Part of the [Software Engineering Commons](#)

Recommended Citation

Hua, K. L. (2007). *A comparison of edge detection methods for segmentation of skin lesions in mobile-phone-quality images*. https://ro.ecu.edu.au/theses_hons/1275

This Thesis is posted at Research Online.
https://ro.ecu.edu.au/theses_hons/1275

Edith Cowan University

Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study.

The University does not authorize you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following:

- Copyright owners are entitled to take legal action against persons who infringe their copyright.
- A reproduction of material that is protected by copyright may be a copyright infringement. Where the reproduction of such material is done without attribution of authorship, with false attribution of authorship or the authorship is treated in a derogatory manner, this may be a breach of the author's moral rights contained in Part IX of the Copyright Act 1968 (Cth).
- Courts have the power to impose a wide range of civil and criminal sanctions for infringement of copyright, infringement of moral rights and other offences under the Copyright Act 1968 (Cth). Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

USE OF THESIS

The Use of Thesis statement is not included in this version of the thesis.

Use of theses

This copy is the property of Edith Cowan University. However the literary rights of the author must also be respected. If any passage from this thesis is quoted or closely paraphrased in a paper or written work prepared by the user, the source of the passage must be acknowledged in the work. If the user desires to publish a paper or written work containing passages copied or closely paraphrased from this thesis, which passages would in total constitute an infringing copy for the purpose of the Copyright Act, he or she must first obtain the written permission of the author to do so.

Signature: _____

Date: _____

11/03/2008

A COMPARISON OF EDGE DETECTION METHODS FOR SEGMENTATION OF SKIN LESIONS IN MOBILE-PHONE- QUALITY IMAGES

A thesis submitted in partial fulfilment of the requirements for the degree of

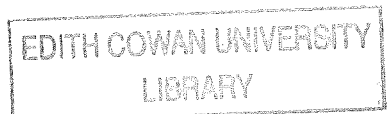
Bachelor of Science (Software Engineering) with Honours

By: KHANH LINH HUA
[REDACTED]

Faculty of Computing, Health and Science
Edith Cowan University

Supervisor(s): Dr ALFRED TAN
Dr MARTIN MASEK
Mr. MICHAEL COLLINS

Date of submission: Semester 2, 2007



Declaration

I certify that this thesis does not incorporate without acknowledgement any material previously submitted for a degree in any institution of higher education, and that, to the best of my knowledge and belief, it does not contain any material previously published or written by any other person except where due acknowledgement is made.

Signature: __

Date: 11/03/2008

TABLE OF CONTENTS

ABSTRACT	1
ACKNOWLEDGEMENTS	2
1 INTRODUCTION	3
1.1 BACKGROUND STUDY	3
1.2 RESEARCH QUESTIONS:	7
2 LITERATURE REVIEW	8
2.1 OVERVIEW OF EDGE DETECTORS	8
2.2 ROBERTS CROSS EDGE DETECTOR.....	8
2.3 SOBEL EDGE DETECTOR.....	9
2.4 PREWITT EDGE DETECTOR.....	10
2.5 LAPLACIAN OF GAUSSIAN.....	10
2.6 CANNY EDGE DETECTOR	11
2.7 REVIEW OF SEGMENTATION METHODS TO DETECT SKIN CANCER	13
2.8 EDGE DETECTORS AS PART OF THE PROCESS OF DETECT SKIN CANCER.....	14
3 THEORETICAL FRAMEWORK	16
3.1 STRENGTHS AND WEAKNESS OF EDGE DETECTOR.....	16
3.2 QUALITY OF VGA CAMERA PHONE IMAGES	16
4 MATERIALS AND METHODS	18
4.1 DATA AND EQUIPMENTS.....	18
4.2 EDGE DETECTION METHODS AND PROCEDURES	18
4.3 EVALUATION METHODS.....	19
5 RESULTS AND FINDINGS	21
5.1 PRE-PROCESSING PHASE.....	21
5.2 FINDING THE LESION BORDER	29
5.2.1 <i>Approach 1 – Automatic edge parameters</i>	29
5.2.2 <i>Approach 2 – Manual tuning of edge parameters</i>	38
6 CONCLUSION	42
7 FUTURE WORK	43
8 GLOSSARIES	44
8.1 IMAGE PROCESSING TERMINOLOGIES	44
8.2 SKIN CANCER TERMINOLOGIES.....	45
9 REFERENCES	48

10 APPENDIX.....53

10.1 APPENDIX A53
 10.1.1 SKIN LESSION IMAGES.....53
 10.1.2 Images take from camera phone.....58
10.2 APENDIX B (SOURCE CODE)64
 10.2.1 Function *pre-process.m*64
 10.2.2 Function *colorspace.m*.....65
 10.2.3 Function *gaussgradient2.m*.....74
 10.2.4 Function *Canny_detector.m*.....76
 10.2.5 Function *LoG_detector.m*.....76
 10.2.6 Function *Sobel_detector.m*77
 10.2.7 Function *Prewitt_detector.m*77
 10.2.8 Function *Roberts_detector.m*.....77

TABLE OF FIGURES

<i>Figure 1: Two typical Roberts Cross kernels.....</i>	<i>9</i>
<i>Figure 2: Two typical Sobel kernels.....</i>	<i>10</i>
<i>Figure 3: Prewitt kernels.</i>	<i>10</i>
<i>Figure 4: The procedure framework</i>	<i>20</i>
<i>Figure 5: Four corners of image.....</i>	<i>22</i>
<i>Figure 6: Intensity image that has been converted by using Euclidean distance.....</i>	<i>23</i>
<i>Figure 7: Image in figure 6 is smoothed by 2D Gaussian kernel of standard deviation 2 pixels.....</i>	<i>23</i>
<i>Figure 8: Image from Figure 7 after intensity clamping.....</i>	<i>24</i>
<i>Figure 9: Image 12.jpg. (a) Result from Canny . (b) Result from LoG. (c) Result from Sobel.....</i>	<i>34</i>
<i>Figure 10: Image 7.jpg. (a) Result from Canny. (b) Result from LoG. (c) Result from Sobel.....</i>	<i>37</i>
<i>Figure 11: (a) Image 8.jpg detect by Canny edge detector with default value.....</i>	<i>40</i>
<i>Figure 12: (a) Image 8.jpg- detect by Sobel edge detection with default value</i>	<i>41</i>

LIST OF TABLES

<i>Table 1:</i> 16 samples of normal and cancer skin (1-4: normal moles, 5-16: cancer moles).....	26
<i>Table 2:</i> Intensity images of 16 samples in Table 1 after pre-processing phase from Xu's method	27
<i>Table 3:</i> Result after intensity clamping images in Table 2.	28
<i>Table 4:</i> Default threshold value for Canny, LoG, Sobel, Prewitt and Roberts Cross associate with each image. Default sigma for Canny is 1 and for LoG is 2.....	30
<i>Table 5:</i> Result from Canny and Laplacian of Gaussian edge detection	38
<i>Table 6:</i> Result from Sobel, Prewitt and Roberts-Cross edge detection (using default value and the good range of threshold value)	39

Abstract

The world is faced with a rapidly increasing number of skin cancers every year. Melanoma is the most deadly type of skin cancer though it can be treated if it has been detected at an early stage. However, there is a shortage of dermatologists in rural areas. The increasing number of camera phones, together with improved coverage in rural areas gives some potential for tele-dermatology, whereby people with no local access to a dermatologist can send images of suspicious skin lesions to an expert for assessment. Merely relaying images to a human expert solves only part of the problem, there is still an acute shortage of experts whose time is limited. Computer Assisted Diagnosis (CADi) of lesions promises to reduce the workload of dermatologists by acting as an assistant. Current skin lesion CADi systems employ algorithms that are designed to run on a computer at a clinic. These clinic-based systems are limited when it comes to tele-dermatology as they rely on a suitable quality image being sent in, and need to process a large number of arriving images. An alternative to this process, afforded by the growing capabilities of mobile phones, is to do some of the CADi processing on the phone which was used to take the image. This has the potential advantage that images can be evaluated for quality on the patient's side, making it more convenient to take another image, rather than waiting for the clinic's assessment. Distributing the processing to the patient's phone also eases the workload on the clinic's machine.

The first step towards implementing skin lesion CADi is the segmentation of lesions from the image background; therefore for a mobile phone to perform CADi it is a pre-requisite that it would be able to perform this step. The study seeks to determine, for an existing skin lesion segmentation algorithm, whether it is practical to adapt for mobile phone use given the limitations of the mobile camera's low resolution. The chosen algorithm depends on an edge detection step, and so an investigation will be made into edge detectors. Edge detectors are sensitive to their parameters, pixel size and lighting conditions – thus the parameters published for clinic-based systems which rely on high resolution cameras and custom lighting can not be expected ideal for mobile phone use. Experiments have shown that the approach from Xu et. al. (1999) can only apply on some types of images, which have unique background colour and distinctive from the lesion (foreground) colour.

Acknowledgements

Firstly, I would like to express my gratitude to Mr. Michael Collins for his time and valuable support. Michael was my first supervisor, who introduced to me the world of imaging – a new and interesting research area. Michael also helped me to edit my dissertation and provided helpful comments that greatly improved the presentation and enhanced the clarity of this work.

Secondly, I most grateful to my two supervisors for their role in inspiring this project as well as for their ongoing guidance: Dr. Alfred Tan and Dr. Martin Masek. Alfred Tan has given me a lot of suggestions with his wide knowledge of new technology, essential for mobile phone images. He was the most important resource in my study. I am appreciative for his enthusiastic support during my research. I also would like to thank Dr. Martin, who has supervised me in short time but his support was invaluable. Dr. Martin has always given me helpful comments and also shared his knowledge of image processing in Matlab. He also assisted me in conducting the test functions in this research.

Thirdly, thanks to Cassandra Colvin for her support and suggestions. Thanks for her motivation when I struggle to find direction for my study. Thanks also go to Noleen Costello for her kindly help in solving the problem of my enrolment.

Thank you to the reviewers, who have spent time to provide valuable feedback. I also would like to thank my friend Hiroko Kato for her help and kindly lending me her camera phone for capturing images of my testing.

And finally, thanks to my family, my aunty and my friends for their support throughout my study. I could not finish the degree without their encouragement.

1 Introduction

1.1 Background study

Edge detection is very important to vision systems such as biological and computational system. Since edges are fundamental to determination of an image's properties. In image processing, the goals of edge detection are determining object boundaries of a still image. It is marking those points in a digital image where the luminous intensity changes sharply. These changes include surface orientation, depth or physical properties of materials, the points of tangent discontinuity in the luminance signal (Ziou and Tabbone, 1998).

In medical imaging applications, edge detection plays a crucial role by automating or facilitating the delineation of anatomical structures and other regions of interest. Some examples follow. Xuan, Adali, & Wang (1995) presented a sophisticated method that integrates region growing and edge detection for magnetic resonance image (MRI) segmentation of brain images. They used region growing to handle the complex image structure and edge detection to verify and/or correct region boundaries. Mezghani, Deschenes, Godbout, Branchaud and Guise (2006) applied successful anisotropic filtering and a local Canny-Decriche detector to detect spinal vertebrae edge detection from biplane X-ray radiographic image. Pathak, Haynor and Kim (2000) stated the use of edges detection for boundary delineation yields an accurate detection of prostate boundaries, which is required in many diagnostic and treatment procedures for prostate disease. Their experiment has been tested on 125 images from 16 patients and statistically evaluated by five expert observers, who found that the accuracy of edge detectors is as good as human observers. Moreover, edge detection can also be extended to other applications in medical imaging where poor contrast in the images and the complexity in the anatomy limit the clinical usability of fully automatic edge-detection techniques. Qi and Kuruganti (2006) used Canny's edge detector in the first step to detect breast cancer from infrared image by asymmetry analysis.

Skin cancer is one of the most common global diseases and Australians have the highest incidence (Lavelle, 2003). According to The Cancer Council of Australia -

TCCA (2004), skin cancer is a disease of the skin cells of body caused by over-exposure to ultraviolet radiation from the sun or from other sources like sun-beds but may also be caused by genetic factors and immune system deficiencies. The three most common types of skin cancer are: basal cell carcinoma (BCC), squamous cell carcinoma (SCC), and melanoma. Of these, the most dangerous and deadliest is melanoma where the cancer cells spread quickly to other parts of body. Statistics from the Cancer Council of Australia (2004) show that every year more than 374,000 Australians are diagnosed with non-melanoma (BCC and SCC) skin cancer, of whom 360 die; less than 1% deaths. However, of the 8,800 people diagnosed with melanoma, approximate 1000 cases die; 11.3% deaths. American Academy of Dermatology (AAD) (2006) data suggests that of 95,880 Americans diagnosed with melanoma in 2004, 7,910 will have died by 2006. The AAD also pointed out that skin cancer is curable if it is detected in the earlier stages. Detection may even result from a simple check.

When a mole on skin has been noticed one may apply the ABCD rule, as follows:

- **A**symmetry – check for mismatch of halves of the mole;
- **B**order irregularity, where edges are ragged, notched or blurred;
- **C**olour – where pigmentation is not uniform and shades of tan, brown or black are present; and
- **D**iameter – melanomas are usually greater than 6mm in diameter when diagnosed, but they may be smaller.
 - n.b. Where melanomas are smaller than 6mm, researchers recommended the “growing” of the image called “evolving” to be considered, then an E – (Evolving) is added into ABCD rule (Zwillich, 2004).

According to CIGNA (2006) several techniques have been developed to detect skin cancer, e.g.:

- Digital epiluminescence microscopy (DELM);
- Melanography;
- Dermatoscopy;
- Mole mapping;
- Incidence light microscopy;
- Dermatoscopic oil-immersion photography;

- Fluid free dermatoscopic photography or Cross-polarization, producing similar but possibly clearer result to oil-immersion photography ; and
- Epiluminescence imaging ELM.

Zouridakis, Doshi, Duvic and Mullani (2005) assert these methods rely on delineating the boundary of a lesion based on characteristics like shape, size, symmetry, colour and texture. On the other hand, side-transillumination ELM or TLM is a recent imaging technique which has an advantage of sensitivity to increased blood flow and vascularization and makes visible the subsurface pigmentation in a nevus that the above techniques may have missed.

Several brand names of dermatoscopy devices have been developed and used to detect skin cancer such as: MoleMax™, Nevoscope™, Dermascope™, Episcope™, Dermlite, and SolarScan. However, these diagnostic tools are mostly used in clinics or medical centres by such as general-practitioners, specialists or dermatologists to test accuracy (Dolianitis, Kelly, Wolfe and Simpson, 2005). Clearly, patients need to see dermatologists or clinicians for scanning moles and follow up diagnostic procedures. This is a significant problem for people who live in rural or remote areas, where there is little opportunity to see doctors or specialists face-to-face. Furthermore, the Australian Institute of Health and Welfare (AIHW) reported cancer rates are higher in rural and remote areas. Dr Mark Short of the Health Registers and Cancer Monitoring Unit of Institute emphasized the incidence of melanoma in rural and remote areas in 2001 to 2003 is higher than metropolitan areas because of lifestyle factors like excessive sun exposure. But while the number of skin cancer in rural Australia is increasing significantly, the numbers of dermatologists and specialists in remote area is not: the number of practicing Australian dermatologists has increased from 136 in 1976 to 350 in 2006. Accordingly, even in metropolitan areas people still need to wait for long periods to see dermatologists, so the access to dermatologists in rural and regional Australia is worse (Commens, 2006). Teledermatology has become a solution of interest to address this problem, holding great potential for revolutionizing the delivery of dermatology services to remote areas. Teledermatology can be applied in 2 ways: utilizing videoconferencing equipment conducted in real time or store-and-forward methods where digital images or photographs are transmitted with a clinical history (Kadurina

and Mitoff, 2005). The store-and-forward variant uses asynchronous data (digital images) transfer technology, e.g. via email or web-based utilities. Such systems enable data to be taken at one site, stored on computer and then transmitted to the other end to can be stored again pending review. This method is less expensive and more convenient for patients and dermatologists alike. Real time or synchronous data transfer technologies such as videoconferencing software require all communicating parties, e.g. dermatologist and patient, to be available at the same time (Eminovic, Keizer, Bindels and Hasman, 2007). Oakley (2005) stated that higher resolution of images and higher transmission rates make it easier to evaluate a skin image. The American Academy of Dermatology's Position Statement on Telemedicine (cited by Oaskley, 2005) recommends a connection speed of 384 kbps (using ISDN) and a minimum resolution of 800x600 pixels for diagnostic images, but lower speeds can be successful for selected patients providing there is access to freeze-frame or captured still images. Thus the Internet broadband is ideal for teledermatology but has not yet reached much of rural Australia. Minister for Queensland State Development, Employment and Industrial Relations-John Mickel asserts: "The problem for us in Australia is that although our broadband services in metropolitan areas are reasonable, our regional and rural areas are still missing out" (CAIRNS, 2007). On the other hand, the mobile phone network coverage in Australia almost covers the whole country, population-wise. Ericsson and Telstra have announced the new Telstra Next G network including the Third Generation Mobile System (3GSM) technology and High-Speed Packet Access (HSPA) coverage provides mobile broadband access to 98.8 percent of Australians. According to Ericsson (2007), this new technology will provide high-speed network from remote locations and off-shore of Australia geography.

Since the first camera phone released in 2000, introduced by SHARP and J-Phone in Japan (Sharp, 2000), the camera phone market has grown rapidly. According to O'Keefe (2004) the number of camera phone sales worldwide has reached nearly 150 million is expected to reach 656 million units in 2008. Moreover, the utility of the camera phone has become wider, not limited just to telecommunication or a camera to satisfy a personal hobby but it is also a useful tool in medical treatment. For example, there was an idea to use camera phones to help detect and verify the falls of elderly people (Hansen, Eklund, Sprinkle, Bajcsy and Sastry, 2005). Israeli doctor Nitzan Yaniv has

come up with a new idea by installing extra software and adding a basic IR camera, the camera-phone can be the tool to detect breast cancer, the image then could be immediately transmitted to a medical laboratory for analysis to see whether further checks are necessary (Cascio, 2005).

The increasing rate of skin cancer in rural Australia and the shortage of dermatologists have led to interest in teledermatology as a new technique and convenient tool for patients in remote areas. The increase of the camera phone market and the coverage rate of mobile network in rural Australia and the utility of camera phone in medical treatment are the basic premises of this study. In this study, we seek to combine the use of camera phone and edge detectors to aid detection of skin cancer in its earlier stages.

1.2 Research questions:

In this research, we would like to find answers for following questions:

1. Is it possible to adapt an existing lesion segmentation algorithm for use in mobile phone camera images?
2. Given the lower resolution of mobile phone camera images, which edge detection method is more suitable for the existing lesion segmentation algorithm?

2 Literature Review

2.1 Overview of edge detectors

According to Parker (1997, p.1) "an edge is the boundary between an object and the background, and indicates the boundary between overlapping objects". In a digital image, an edge is a set of connected pixels that lie on the boundary between two regions. Edge detection is one of most commonly used tools in image processing and, hence, object-recognition. Accordingly, edge detection is the process of locating the edge pixels on the boundaries of objects that fall within a digital image (Rubino, 2003).

Several ways to perform edge detection have been designed to date. Claypoole, Levis, Blashyam and Kelly (1997) classified edge detection operators into two major categories: gradient and Laplacian methods. The gradient method is detecting edges by finding the maxima and minima in the first derivative of the image. Examples of this method are Roberts Cross, Prewitt and Sobel. Alternatively, the Laplacian method is used to find the edges by searching for zero-crossings in the second derivative of the image.

Neoh and Hazanchuk (2005) defined the basis of an edge detection operator as the process of determining the level of variance between different pixels by a matrix area gradient operation. It is calculated by forming a matrix centred on a pixel chosen as the centre of the matrix area. The middle pixel in the matrix is classified as an edge when the value of this matrix area is above a given threshold.

2.2 Roberts Cross edge detector

The Roberts Cross operator was one of the first operators used to detect edges in an image. This particular edge detector consists of a pair of 2x2-convolution masks to compute a 2-dimensional spatial gradient of an incoming matrix. The G_x image yields diagonals that run from the top-left to the bottom-right of the matrix while the G_y image yields diagonals that run from the top-right to the bottom-left (Fisher, Perkins, Walker and Wolfart, 2003).

Fisher et. al. (2003) depicted Roberts Cross kernel as Figure 1 below.



Figure 1: Two typical Roberts Cross kernels.

The gradient components in each dimension of G_x and G_y are measured separately by applying the masks separately to the image. Then the combination of the two individual images G_x and G_y are used to find the absolute magnitude of the gradient at each point. According to Fisher et. al. (2003), the approximation equation (1) is used to calculate the absolute magnitude of the gradient and equation (2) to calculate the angle of orientation of the edge that associates to the spatial gradient:

$$|G| = |G_x| + |G_y| \tag{1}$$

$$\theta = \arctan(G_y/G_x) - 3\pi/4 \tag{2}$$

Roberts Cross operator is very quick to compute because only four input pixels need to be examined to determine the value of each output pixel and calculations are limited to those of addition and subtraction. However, as Roberts Cross kernels are relatively small, it is very sensitive to noise and is only good for images that have very sharp edges. (Fisher et. al, 2003).

2.3 Sobel edge detector

Sobel's algorithm is similar to Roberts's algorithm; both methods have same basis: examine the two axes edges individually and then combine them for the resulting edge detection. Roberts Cross kernels are used to find the edges that run along the vertical axis of 45 degrees and axis of 135 degree, whereas Sobel's kernel tries to detect edges along the horizontal axis and vertical axis (Figure 1). The algorithm of Sobel is less susceptible to noise than Roberts Cross' algorithm because it uses 3x3 gradient edge detectors, creating better averages of the neighbouring pixels (Rubino, 2003).

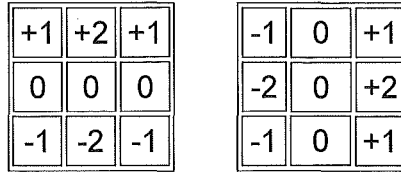


Figure 2: Two typical Sobel kernels.

According to Rubino (2003), the calculating of the absolute magnitude gradient and the angle of orientation of the edge are presented in equation (3) and (4):

$$|G| = |G_x| + |G_y| \tag{3}$$

$$\theta = \arctan(G_y/G_x) \tag{4}$$

2.4 Prewitt edge detector

Fisher et. al. (2003) assert that the Prewitt operator is similar to Sobel's but the mask coefficients are different. The kernel produces similar results to the Sobel but is not as isotropic. Isotropic in an image processing context means that applies equally well in all directions in an image. The masks of Prewitt operator are presented by Fisher et. al. (2003) shows in the Figure 3 bellow:



Horizontal mask

Vertical mask

Figure 3: Prewitt kernels.

2.5 Laplacian of Gaussian

An example of the second category is the Laplacian operator, which seeks an edge in the second derivative to increase the chances of detecting a weak edge that may not be detected in a first derivative, e.g. Sobel's, operator. However, Marshall (1994) stated that the Laplacian method is also sensitive to noise. Accordingly it is often used with

image smoothing methods, like Gaussian smoothing, in order to reduce such sensitivity by ignoring zero-crossings produced by small changes in image intensity. This compound method is called Laplacian of Gaussian (LoG).

Fisher et. al, (2003) calculate the Laplacian $L(x,y)$ of an image having pixel intensity values $I(x,y)$ by equation (5) and the 2 dimensions Laplacian of Gaussian function centered on zero and with Gaussian standard deviation Sigma (σ) is presented in equation (6):

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \quad (5)$$

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (6)$$

2.6 Canny edge detector

According to Owens (1997), Canny's edge detector also applies Gaussian smoothing and Green (2002) states that Canny's method is viewed by many as being optimal for edge detection. In his work, Canny (1986) identified three criteria that an edge detector must address, specifically:

1. *Error rate* - The response should find only edges and should find all of them without missing any, there be no responses to non-edges.
2. *Localisation* – The edge detector should be able to minimise as much as possible any distance between the edge pixels as found by the edge detector and the actual edge. Good localisation is close to centre of the true edge.
3. *Response* –Only one response to a single edge, multiple edge pixels should be eliminated.

Owens (1997) describes Canny's edge detector by the following steps:

1. Firstly, the image is smoothed by using Gaussian filter with a specified standard deviation to reduce noise. Wang (2004) illustrates standard deviation in formula below:

$$g(m, n) = G_{\sigma}(m, n) * f(m, n) \quad (7)$$

Where

$$G_{\sigma} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{m^2 + n^2}{2\sigma^2}\right] \quad (8)$$

Then look for maximum gradient in the first partial derivatives of the resulting signal using convolution similar to Sobel masks. The direction producing the largest result at each pixel point is marked. Maximum gradient and angle of direction are presented by Wang (2004) as:

$$M(m, n) = \sqrt{g_m^2(m, n) + g_n^2(m, n)} \quad (9)$$

And

$$\theta(m, n) = \tan^{-1}[g_n(m, n)/g_m(m, n)] \quad (10)$$

Non-maximal suppression is performed in the next step which is used to trace along the edge in the edge direction. Any gradient value is not a local peak will be set to zero. Threshold M is shown in equation below by Wang (2004):

$$M_T(m, n) = \begin{cases} M(m, n) & \text{if } M(m, n) > T \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Where T is chosen that all edge elements are kept while most of the noise is suppressed

2. Find connected sets of edge points and form into lists.
3. Relate the edge direction to a traceable direction to resolve orientation of edge.
4. Canny introduced thresholding hysteresis which is used as a means of eliminating streaking. Hysteresis combines 2 threshold values: low threshold T1 and high threshold T2. The higher threshold value is usually 3 times higher than

the lower. A valid edge point is classified if its gradient value greater than the higher threshold. In addition, any pixels that connected to these valid edge points and have gradient value within higher and lower threshold value are also classified as edge points.

2.7 Review of segmentation methods to detect skin cancer

Several approaches have been developed for detecting skin cancer over the past three decades from non computer aided diagnosis to computer aided diagnosis, and invasive biopsy to non-invasive biopsy. Various image segmentation methods have been designed to delineate lesion boundaries. Kreutz, Anschutz, Gehlen, Grunendick, and Hoffmann (2001) used digital image processing and mixture-of-experts to diagnose skin cancer. Theirs was a hybrid method combining statistical clustering of the colour space and a hierarchical region growing algorithm to detect boundary of lesion. The Polartechnic Solarscan melanoma diagnosis was developed over eight years by CSIRO and the Sydney Melanoma Unit, University of Sydney at Royal Prince Alfred Hospital. The method-segmentation of lesion used in this technique is initially based on seeded region growing. If the result is not satisfactory then the semi-automated procedure based on colour clustering will take place to extract the features of lesion (Talbot and Bischof, 2003). Guo and Aslandogan (2003) developed a system mining skin lesion data with Density-Based Spatial Clustering of Applications with Noise (DBSCAN) or Generalise Density-Based Spatial Clustering of Applications with Noise (DBSCAN/GDBSCAN) technique. This DBSCAN algorithm is a clustering method relying on a density-based notion of clusters, and it is designed to discover clusters of arbitrary shape and good efficiency on large databases. The program from Guo and Aslandogan consists of two major steps: the image is split into smaller regions until all the regions meet the homogeneity criteria set by the threshold for splitting. Then the small split regions are grouped by applying DBSCAN algorithm to form final regions of interest. During the splitting process, the Euclidean distance in RGB colour space is used to measure colour distance and test homogeneity between colour regions. This program does not require any colour transformation process. Thus the identified objects can be represented in the original colour space and ready for retrieval and manipulation of colour data. Segmentation image using DBSCAN will be applied by

iteration. The first iteration is used to identify the lesion, the second moves inside the lesion, identifying sub-regions. However, the DBSCAN will not iterate more than twice to avoid over segmentation. Ilea and Whelan (2006) present automatic segmentation of skin cancer image using adaptive colour clustering. In their research, the Adaptive Spatial K-means clustering technique was used to extract the colour features from skin cancer and the Local Binary Patterns was applied for texture analysis.

2.8 Edge detectors as part of the process of detect skin cancer

Denton, Duller and Fish (1995) proposed an approach using an edge preserving 7x7 median filters to reduce the noise in image then applied thresholding and subsequent morphological closing and identification by size and shape to locate the lesion. The diameter of the lesion is used to calculate the initial bounding box and the space constant sigma for the Laplacian of Gaussian (LoG) edge detector (minimum diameter equal two times the width of the sensitive region of the filter). After finding an initial boundary, a successive refinement of the LoG leads to the definition of the lesion or mole boundary. The researchers suppose that Canny or IIR filter may similarly be used to determine the boundary, but did not test these methods.

Xu, Jackowski, Goshtasby, Roseman, Bine and Yu, (1999) present an automatic method for segmentation of skin cancer images and other pigmented lesions. They first reduce a colour image into an intensity image by transforming RGB colour into CIELAB space and then transform the resulting image to intensity space. Here, intensity of a pixel is set as the Euclidian distance of a pixel LAB colour from the LAB colour of normal skin (as determined by a heuristic of taking the median colour from areas at the image corners) . Image segmentation uses an appropriate edge detector to detect the boundary of a lesion. Double thresholding is used to focus on an image area where a lesion boundary potentially exists. Image edges that are extracted from combining edge detector and double thresholding are then refined using an elastic curve model, which is based on the rational Gaussian formulation.

Zambanini, Langs, Sablanig, and Maier, (2007) presents the feature-based image registration method that consists of four main steps:

- 1) feature detection detect interest point in images,

- 2) feature matching where the interest points are matched by means of their feature descriptions,
- 3) transform model estimation to compute the parameters of the mapping function using matched interest points,
- 4) image re-sampling and transformation use the computed mapping function.

In the study, the comparison is performed of application of Difference of Gaussian (DoG) and Canny edge detectors for the first step of finding interest points. The result has shown that Canny interest points are more stable than DoG interest points.

The review has shown that although edge detection methods were not presented as a standalone method to detect skin cancer, they have been used as a vital part in the process of segmentation of skin lesions. It is intended that this study will determine whether such techniques may be applied using typical mobile phone camera quality images.

3 Theoretical Framework

3.1 Strengths and weakness of edge detector

Roushdy (2006) observed in his comparison of edge detection algorithms that the gradient edge detector such as Sobel, Prewitt are simple calculations to detect edges and their orientations but the results are unreliable in the presence of noise. His experiment has proven that although LoG operators can find the correct places of edges and testing a wider area around a pixel but may not find the orientation of an edge. In the other hand, Canny is seen as optimal for edge detection in the presence of noise.

Owens (1997) asserted there is number of problems with gradient-based edge detection schemes: a common complication for gradient based methods is that the threshold values and the width of convolution masks have to chosen by user, the width of the mask affects the positions of zero crossings and maximum intensity gradients in the image. Ideally, the estimated position of any edge should not be affected by the size of the convolution mask. Additionally, the corners are often missed because the 1D gradient at corners is usually small. First derivative operators will only find step-like features. To find the line, we need to use other operator of second derivative such as Canny operator for finding lines.

3.2 Quality of VGA camera phone images

Camera phones have become, arguably, the most prolific image capture device in the world. However, it has not yet replaced the dedicated digital camera in terms of quality of image. According to InfoTrends/CAP Ventures (InfoTrends, 2005) the quality of image from camera phone is still poorer than digital camera. Avecmobile (2004a) asserts the camera of high-end smart phones can be compared with low-end digital cameras which has two mega-pixels or less.

In this study, the VGA camera phone has been chosen for the following reasons: VGA is the most popular digital camera resolution in camera phones at the moment, and it is a standard resolution for capturing, viewing on mobile screen and PC desktop screen. The size of picture 640x480 pixel is only taken from 15-30Kb per VGA size photo so it

optimises use of available memory space (Avecmobile, 2004j). A VGA image is relatively small file and is quickly and easily transmitted over an existing mobile network. If the VGA image, which has poor quality compared to dedicated digital cameras, is able to be used to detect skin cancer, then it may be supposed that technological improvements may only render camera phone images more practicable for using in teledermatology for detecting, or aiding to detect, skin cancer.

4 Materials and methods

4.1 Data and equipments

There are two sets of images (refer to appendix A): normal skin moles and melanoma skin moles were derived from www.about.com website (Brannon, 2007). These photos are captured by Nokia 6600 camera phone with VGA resolution as the VGA camera is a popular and basic resolution camera phone. Then these images are transferred to a laptop to perform the experiments documented within this thesis.

4.2 Edge detection methods and procedures

This research is an implementation of the approach by Xu et. al.(1999). The testing is conducted in Matlab programming language and will be illustrated in Figure 4.

- **Pre-processing phase:** where the RGB colour images of normal and cancer lesions are first transform into standardised colour space - the CIE L*a*b* space, which has less redundancy than RGB space. The median colours at the four corners of an image are used to get the colour of image background. The Euclidean distance will then be used to calculate the colour difference between an image pixel and its background. It also transform colour image into an intensity image. This intensity image is then smoothed by a 2 dimension kernel of Gaussian function to reduce the gradient of the image belonging to background as well as those belonging to lesion. Hence, the gradient of intensities falling on a lesion's boundary will be increased. The aim of this colour mapping is to enhance the boundary of a lesion while suppressing details inside and outside a lesion because our interest is in finding the edge which defines a boundary of lesion.
- **Finding lesion border phase:** As the purpose of this study is to compare results from different edge detectors, therefore we will not further refine the image region using a closed elastic curve as presented in Xu's (1999) research. Instead, we will be focussing on experimentation with different edge detector methods like Roberts Cross, Sobel, Prewitt, Laplacian of Gaussian and Canny. The result of edge detections will be observed base on two approaches:
 - o **Approach 1:** These parametric edge detection methods will be tested using default parameters from the Matlab implementation.

- Approach 2: Performing edge detection methods with various values for its parameter such as threshold and sigma for Canny and Laplacian of Gaussian and threshold with direction for Sobel to find out the best result of each edge detector.
- **Evaluate result**: Compare the results of each method against each others as well as against the results from the original algorithm by Xu et. al. (1999).

4.3 Evaluation Methods

Roushdy (2006) asserts there are two methods to evaluate performance of edge detectors, subjective and objective. Subjective is based on psychology and human judgment which is limited by difference of human perspective and depends on the individual perspective. On the other hand, objective methods used to evaluate the performance of edge detectors using signal to noise ratio and mean square error compare between the edge detector images and the original. However, in this study the result will be compared based on human perception of image quality. Thus the method used to evaluate in this study is the subjective method, which according to Roushdy performed better than an objective method if the goal of evaluation is concerned with the quality of the images as defined by human vision perception.

According to Roushdy (2006), there are three types of subjective method: 1) impairment tests where the test subject grades images that consider how bad they are, 2) quality tests, which rate images based on how good they are and 3) comparison tests, where images are compared side-by-side. This provides a relative measure and is the easiest metric to evaluate images by human vision. Comparison tests are the most useful of the subjective methods. In this experiment, the data collects from each edge detector filter will be compared side-by-side with the original image and with results from other edge detectors using a comparison test.

i/-Pre-processing phase

1/- Transform RGB image into colour space

2/- Colour mapping using Euclidean distance

3/- Smoothing image using 2 kernel of Gaussian smooth

4/- Finding intensity pixels which are greater than 1 and convert them to 1

ii/-Finding lesion border phrase

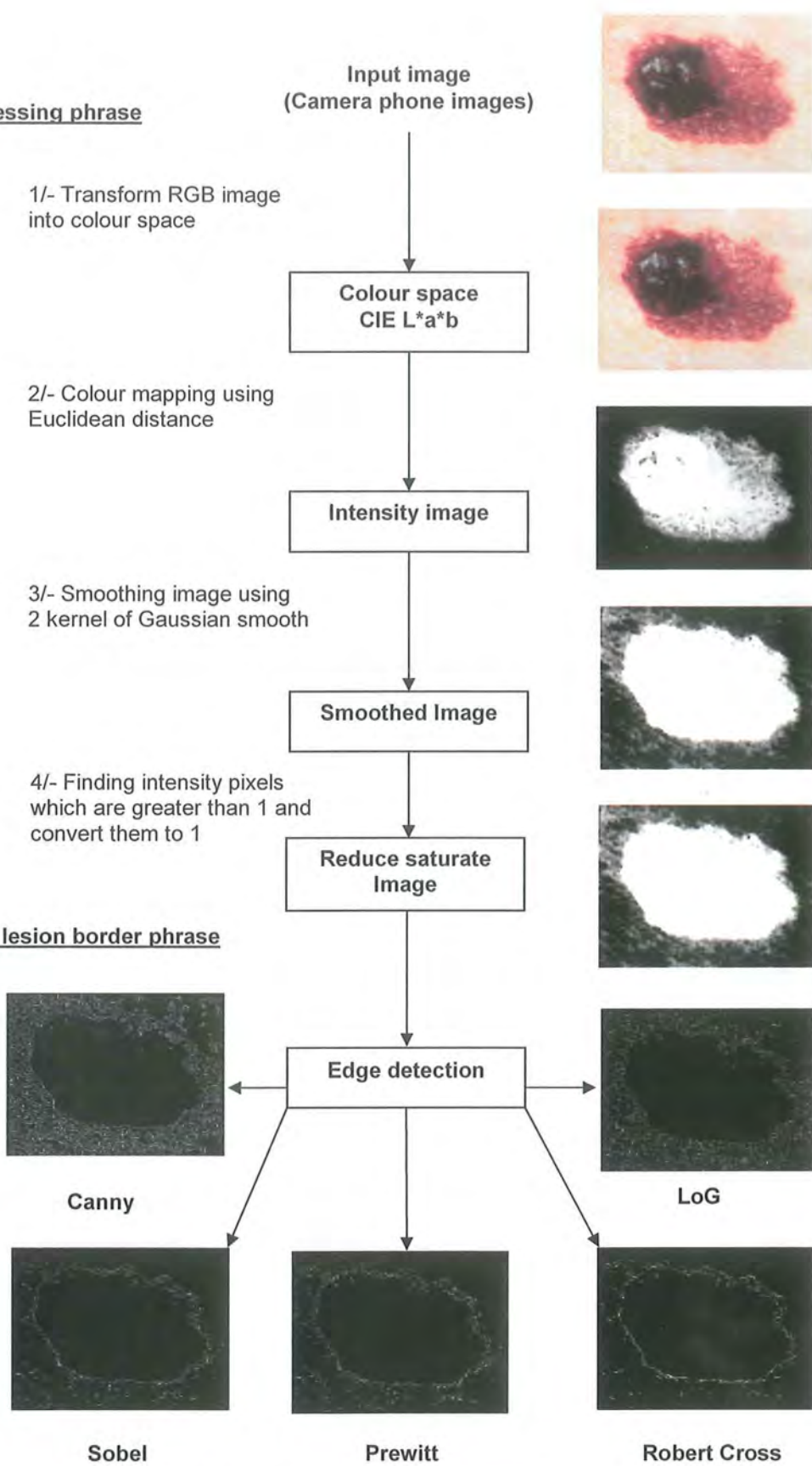


Figure 4: The procedure framework

5 Results and Findings

5.1 Pre-processing phase

Image captured by camera phone are normally in the RGB colour space. Edge detection in the Xu et. al. (1999) paper is applied after the conversion of colour images to intensity image. This is performed in two stages, first converting the RGB image to the CIELAB colour space, then to intensity.

Following the procedure documented by Xu et. al. (1999), the RGB image is first converted into a standardized colour space called CIE LAB. In our experiment, we convert RGB colour from the original image to XYZ colour space then convert XYZ to CIELAB. The result of this is demonstrated in Figure 4 from step 1 to step 2. This conversion is done using a function provided by Pascal Getreuer (Getreuer, 2005).

During this pre-processing phase, there were two significant observations mentioned in Xu's paper:

1. Colours of background and lesion in the image are usually different. Therefore colour changes between background and lesion can be used to effectively segment images.
2. On the other hand, when segmenting a skin image, some colours in the background may appear in a lesion and vice versa. Thus such variation should be ignored. Only the colours belonging to a lesion boundary are important in image segmentation.

The first observation from Xu et. al. (1999) has pointed out the change in colour of lesion and its background is similarly observed in all images; therefore they have conducted a method to determine the background colour by using median colour of four corners from the CIELAB image, each corner is sample with a window size of 10x10 pixels. This is depicted in Figure 5.



Figure 5: Four corners of image

The colour distance between the pixel and the determined image background (ΔE) is calculated using the Euclidean distance in LAB colour space:

$$\Delta E = \sqrt{\Delta L^2 + \Delta a^2 + \Delta b^2} \quad (12)$$

Where:

ΔL , Δa and Δb are the differences between the LAB colour space components of the pixel from the background found using the median of the four image corners.

An intensity image is constructed where each pixel represents the distance from its colour in the LAB image to the background colour. In this intensity image, moles are expected to appear bright due to their higher difference in colour to normal surrounding skin.

Applying this method, the colour image from Figure 5 has been converted into the intensity image that is depicted in Figure 6.

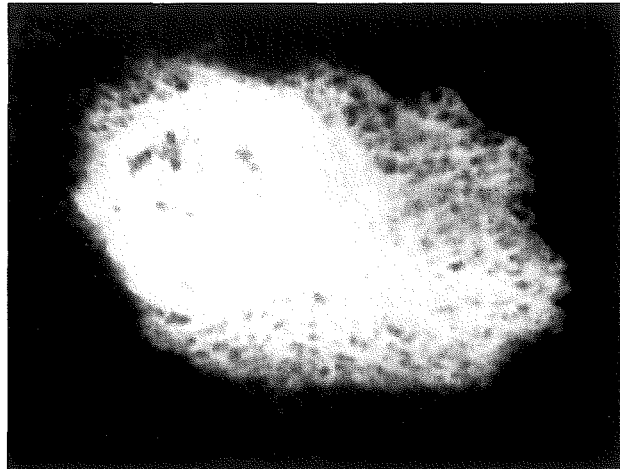


Figure 6: Intensity image that has been converted by using Euclidean distance

To account for the second observation from Xu et. al. (1999), we re-map the intensities using a sigmoid function. The effect of this is to increase the separation between low intensities (background) and high intensities (lesion), whilst reducing the separation within both the high and low intensity pixel groups. To complete this step, we modified a function contributed by Guanglei Xiong (Xiong, 2005) to match the function documented in Xu et. al. (1999). Figure 7 shows the result of remapping the image from Figure 6.

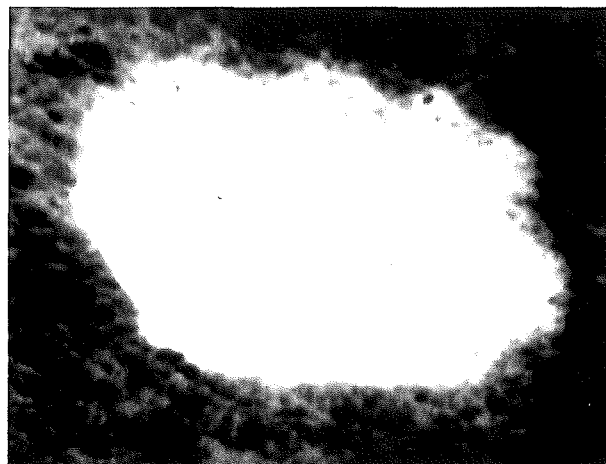


Figure 7: Image in figure 6 is smoothed by 2D Gaussian kernel of standard deviation 2 pixels

In Xu's paper they have applied the Sobel edge detector to the outcome of the previous stage. In our study, we continue one step further before applying edge detection. In this further step, we clamped all pixels with intensity greater than 1 to 1. This is to further eliminate intensity differences within the lesion, producing a uniformly bright lesion as shown in Figure 8.

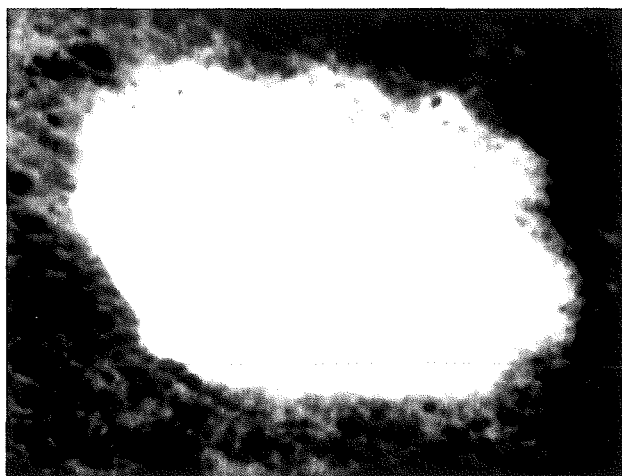


Figure 8: Image from Figure 7 after intensity clamping

After testing through 16 samples from this pre-processing phase, our findings are:

- The quality of the resulting intensity image is strongly dependent on the quality of the input image. Tables 1, 2 and 3 have shown that only input images with distinct background colour produce adequate results. The border of lesions in such images number 2, 4, 12 and 14 are distinctive from the image background. Meanwhile the border of lesion in other images is not quite well defined. This problem can be caused by: Image which is taken by camera phone has been mostly affected by noise, such as poor or non-uniform lighting. For example, Image number 1 is very noisy and the colour from background or from four corners is very much similar to colour around the boundary of lesion. The effects of lighting are shown in the centre of the lesion as the bright area, which created unexpected result. When segmentation by such method. When the sigmoid was used to increase the difference between low and high intensities, this amplified the effects of these bright regions within the lesion. In image number 3, the

shadow cast by the mole (in the lower right) makes the border of the lesion harder to define (as shown from the result in Tables 2 and 3).

- The quality of images used in Xu's paper are very clear, the background colour and the lesion colour are clearly distinctive. In our test, the input images are derived from the Internet and have been captured by VGA camera phone, which produced low resolution images. Even if the image is taken directly of a live subject, it is still prone to the low resolution and noise inherent to mobile camera phones. Thus, segmentation by this pre-processing is not a practical approach to use on mobile phone images.

Table 1: 16 samples of normal and cancer skin (1-4: normal moles, 5-16: cancer moles)



1



2



3



4



5



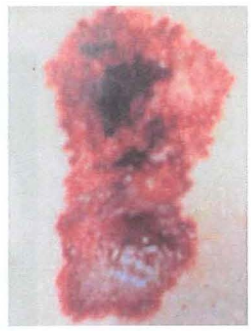
6



7



8



9



10



11



12



13



14



15



16

Images 1-6 reference from About.com (Brannon, 2007);

Images 7-9 from MyoClinic.com (MyoClinic.com, 2006);

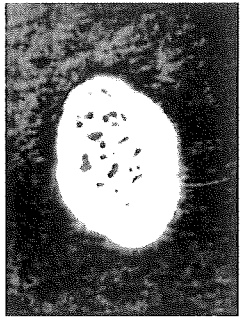
Images 10-13 retrieved from About.com (Frayer, 2006);

Images 14-16 retrieved from Dermnet website (Dermnet, 2007)

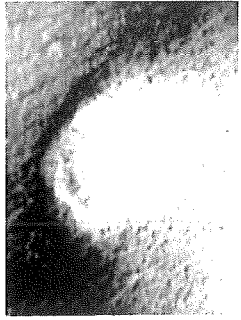
Table 2: Intensity images of 16 samples in Table 1 after pre-processing phase from Xu's method



1



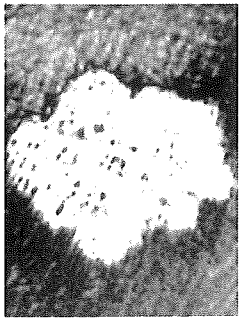
2



3



4



5



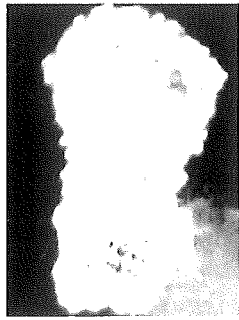
6



7



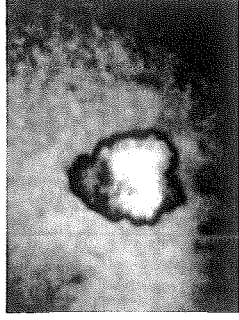
8



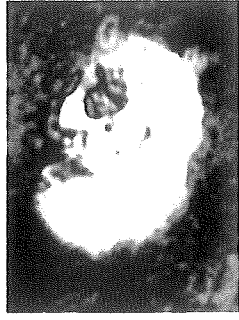
9



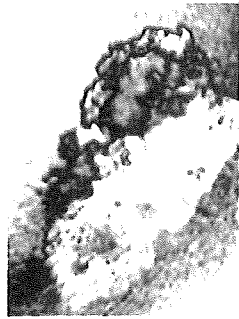
10



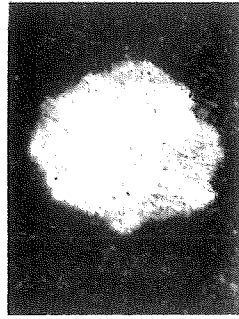
11



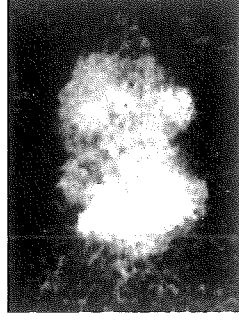
12



13



14

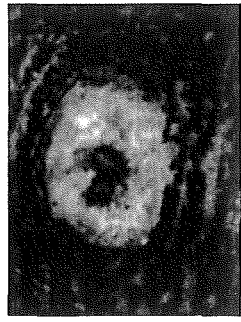


15

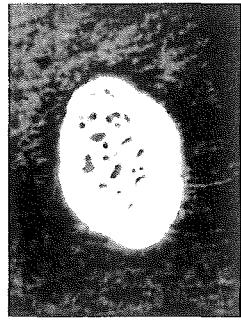


16

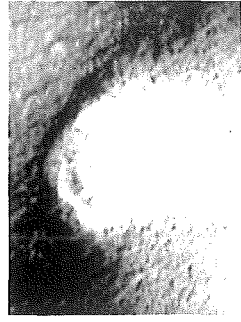
Table 3: Result after intensity clamping images in Table 2.



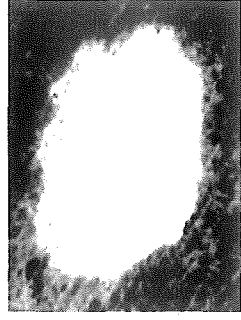
1



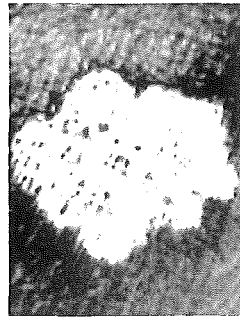
2



3



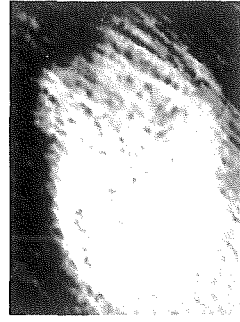
4



5



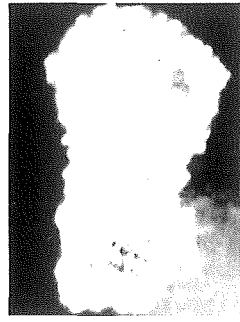
6



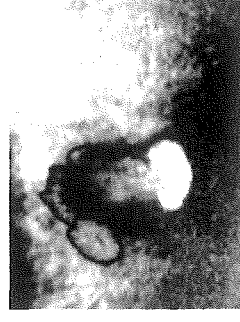
7



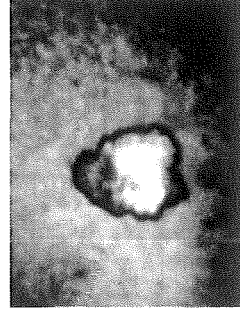
8



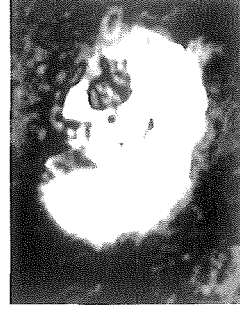
9



10



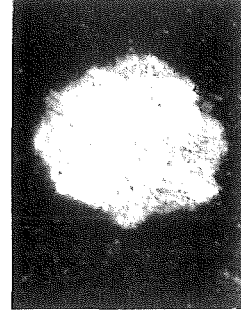
11



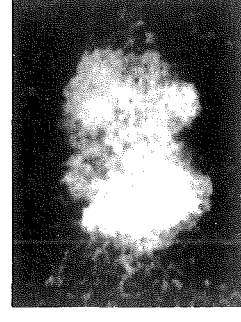
12



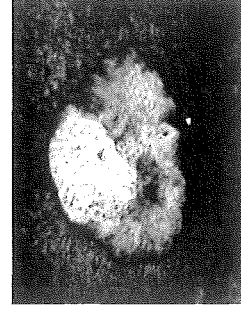
13



14



15



16

5.2 Finding the lesion border

Despite the setback in the previous phase, where it was determined that mobile camera phone images are largely unsuitable for the existing pre-processing approach we investigated further on the possibility to segment these images with the aid of edge detection. Thus in this phase, we applied the edge detectors: Canny, LoG, Prewitt, Sobel and Roberts Cross to define the border of lesions from the smoothed images that were yielded from the previous pre-processing phase above. We evaluated these edge detectors based on 2 approaches. The first approach uses parameters for the edge detectors as determined by Matlab's 'edge' function, which analyses the image to determine suitable parameters for the given edge detector. The second approach was to investigate the result of varying the edge detector parameters.

5.2.1 Approach 1 – Automatic edge parameters

The 16 samples in Table 3 have been tested with the five edge detectors using the automatically generated parameter values produced by Matlab. For Canny and LoG there are two parameters, the threshold and sigma value as documented in the review of edge detectors earlier. The only parameter for Sobel, Prewitt and Roberts Cross operators is the threshold value. Table 4 shows the threshold values produced by Matlab for the edge detectors on the tested images. The sigma value for the Canny edge detector in Matlab is 1.0 and sigma value for LoG is 2.0.

Table 4: Default threshold value for Canny, LoG, Sobel, Prewitt and Roberts Cross associate with each image. Default sigma for Canny is 1 and for LoG is 2

Image	Default threshold value				
	Canny	LoG	Sobel	Prewitt	Roberts Cross
1.jpg	0.09375 - 0.23438	0.0023803	0.033466	0.033294	0.044658
2.jpg	0.04375 - 0.10938	0.0029116	0.049552	0.049314	0.063969
3.jpg	0.06875 - 0.17188	0.0026552	0.036241	0.036027	0.047898
4.jpg	0.04375 - 0.10938	0.0020525	0.039829	0.039668	0.051095
5.jpg	0.05625 - 0.14063	0.0033865	0.055883	0.055579	0.072266
6.jpg	0.08125 - 0.20313	0.0054216	0.082948	0.082346	0.107280
7.jpg	0.06875 - 0.17188	0.0026174	0.042146	0.041930	0.054343
8.jpg	0.06587 - 0.17188	0.0026307	0.041224	0.041032	0.054042
9.jpg	0.01875 - 0.04687	0.0015055	0.042757	0.042621	0.054363
10.jpg	0.05625 - 0.14063	0.0023609	0.035561	0.035382	0.046866
11.jpg	0.05000 - 0.12500	0.0017181	0.00281	0.027996	0.037329
12.jpg	0.03750 - 0.09375	0.0020401	0.042448	0.042311	0.054390
13.jpg	0.06250 - 0.15625	0.0042179	0.064986	0.064713	0.083181
14.jpg	0.04375 - 0.10938	0.0021470	0.049908	0.049355	0.067555
15.jpg	0.06250 - 0.15625	0.0022701	0.035127	0.034959	0.046148
16.jpg	0.03750 - 0.09375	0.0033942	0.050166	0.049689	0.067907

The result from running these edge detection methods using these automatically determined values on the test images has shown Canny edge detection produces better result than the others. Upon examination, the border is derived via Canny's detector were clearer than the other four methods. The edge detection which produced the next best set of results is the LoG detector. This is followed by the Sobel and Prewitt detectors, which produces similar result. Finally, the edge from Roberts Cross frequently contains discontinuities and less edge has been detected reliably. These findings are depicted in Figures 9 and 10.

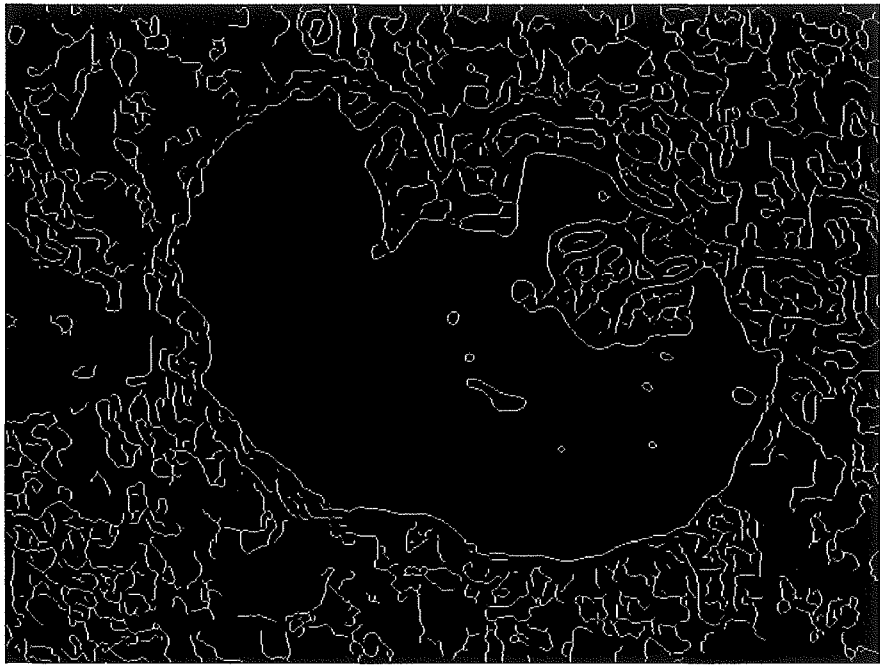
We have also tested these 16 samples using the program called Skinseg.exe. This program is an implementation of skin lesion segmentation using Xu's approach (Goshtaby, 1998). Unfortunately, only a few of our images (namely, image 3, 7, 12, 14 and 16) could be successfully analysed by this program to allow for comparison.

Comparing the different edge detectors to the results from Xu's shows that Canny and LoG edge detectors yield better results than the Skinseg program (which is using Sobel's edge detection method). In some cases the result for our approach is better than Skinseg. An example is shown in Figure 9. The border found by the Skinseg program lies further inside the actual border of the lesion than our Canny and LoG edge detector results, particularly in the upper left of the lesion border. Although the Skinseg is using the Sobel edge detector to find border of lesion in pre-processing step, it should be noted that the results from Skinseg show the results of post-processing of Xu et. al. (1999) produces a single clean lesion border and eliminates the edge artefacts seen in the images from our edge detector application.

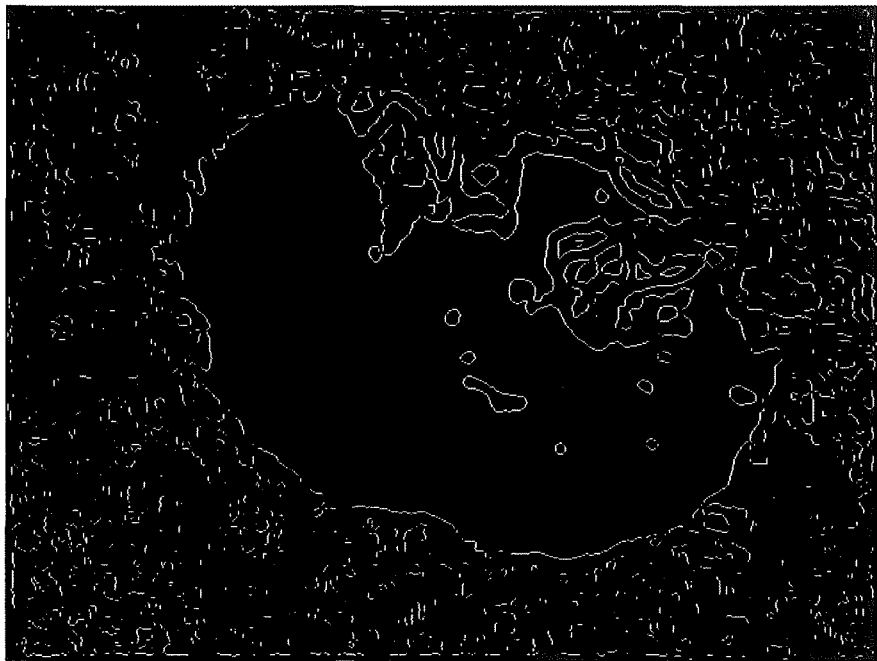
In other cases, the result from the Skinseg program seems better than our results, as in the images illustrated in Figure 10. The border found by Skinseg is closer to actual border from original image while our edge detectors are unable to yield that close border.

The findings that we have found in this approach are:

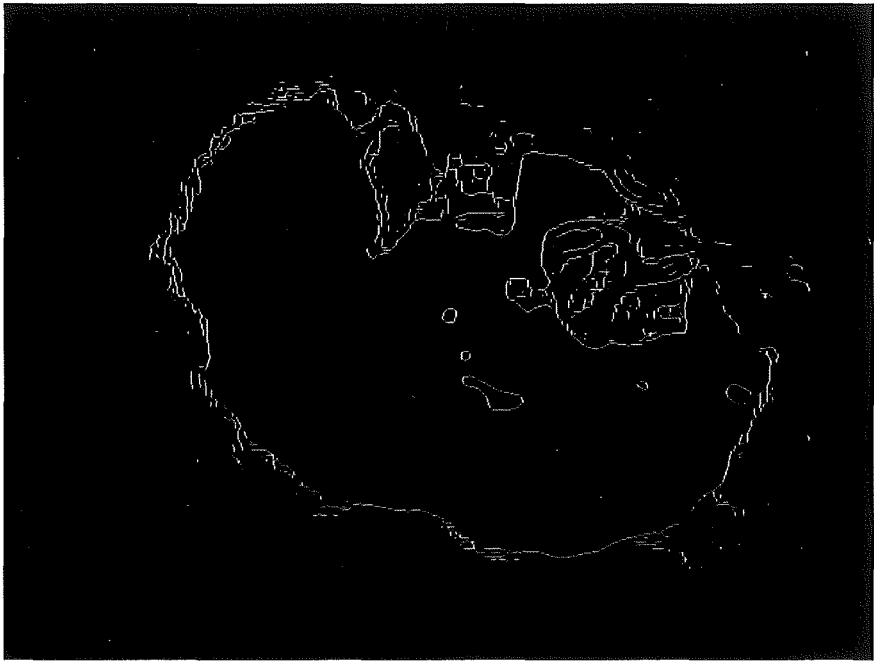
- Canny edge detector has offered the best result among these methods (given the automatically determined parameters from the Matlab implementation). Thus it should be considered to use to define boundary.
- The result of detecting border of lesion is very much dependant on how the result of image that has been done in pre-processing phase. If the pre-processing image does not perform well in terms of having clear intensity image. Then the border of lesion is very hard to find by only simply apply edge detection.



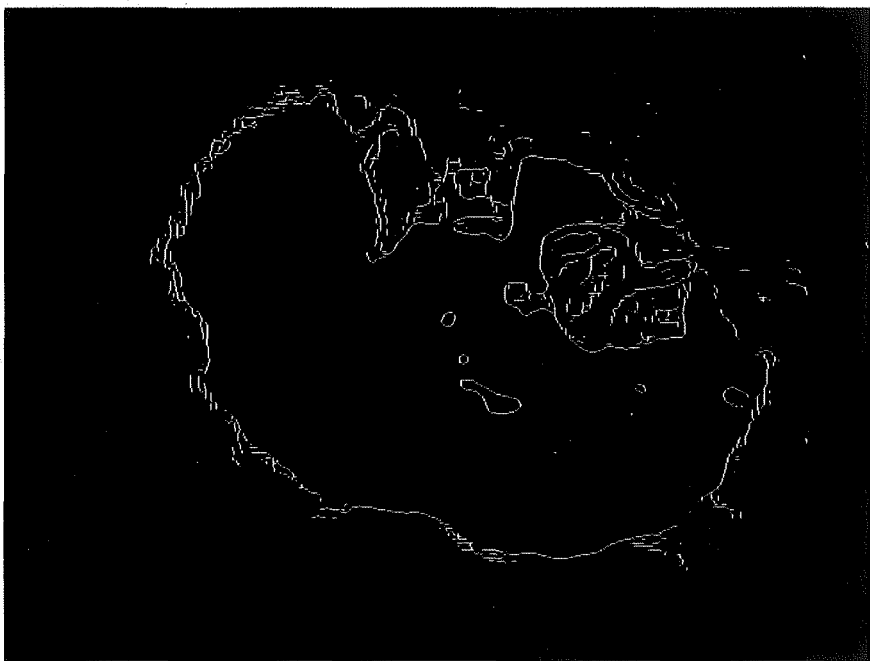
(a)



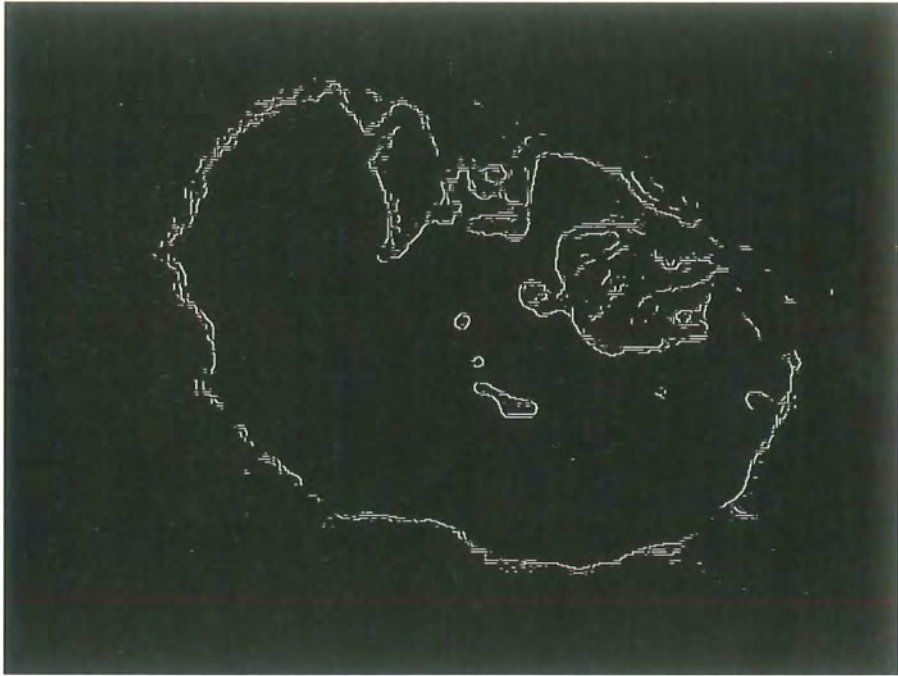
(b)



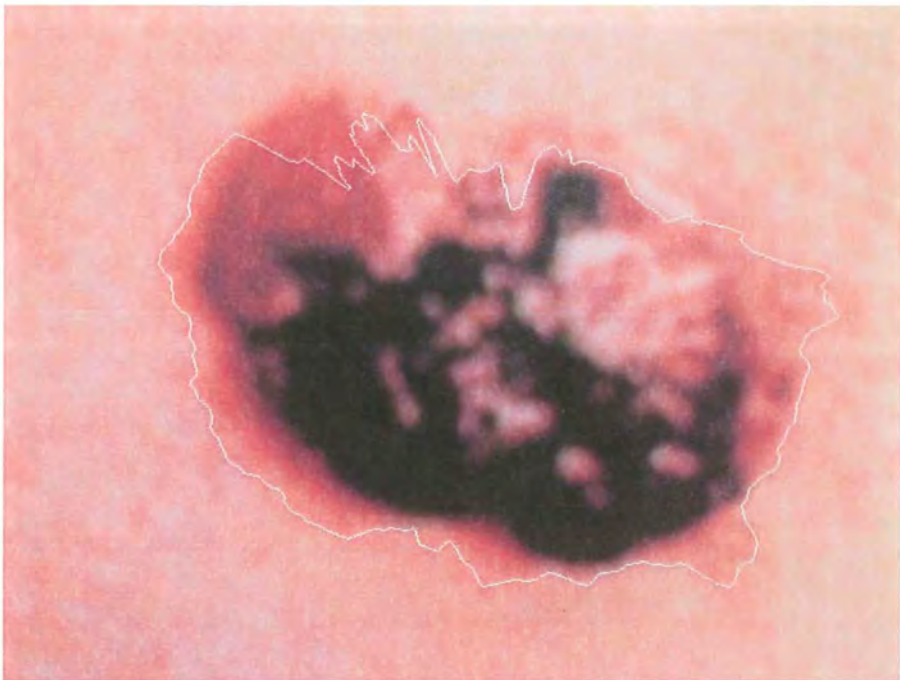
(c)



d)

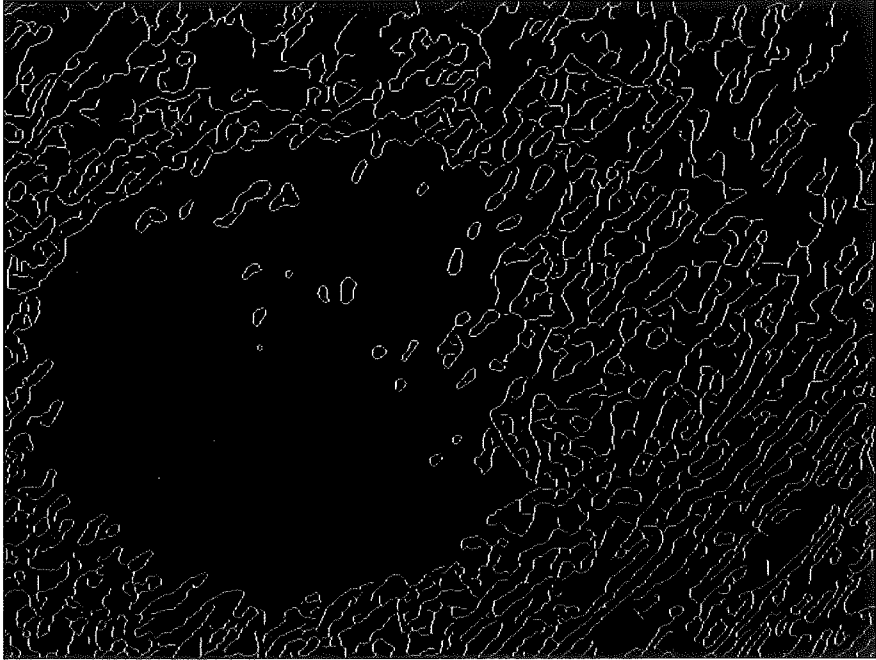


(e)

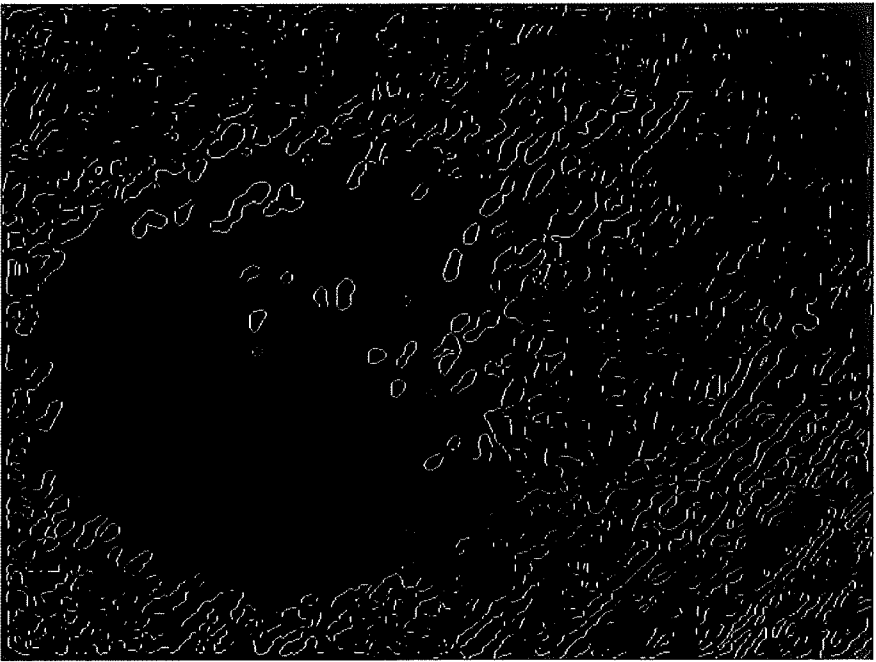


(f)

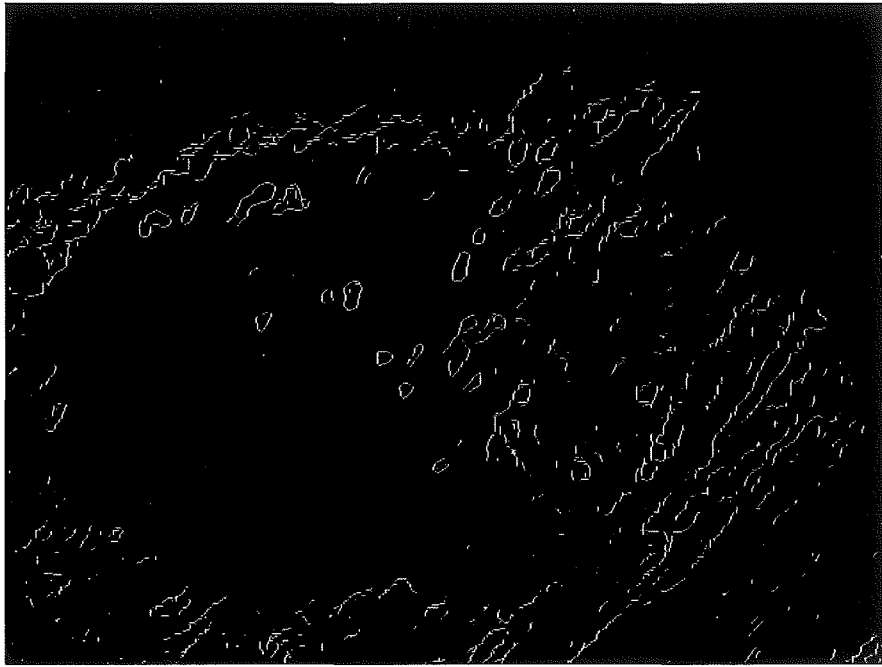
Figure 9: Image 12.jpg. (a) Result from Canny . (b) Result from LoG. (c) Result from Sobel. (d) Result from Prewitt. (e) Result from Roberts Cross. (f) Result from Skinseg.exe



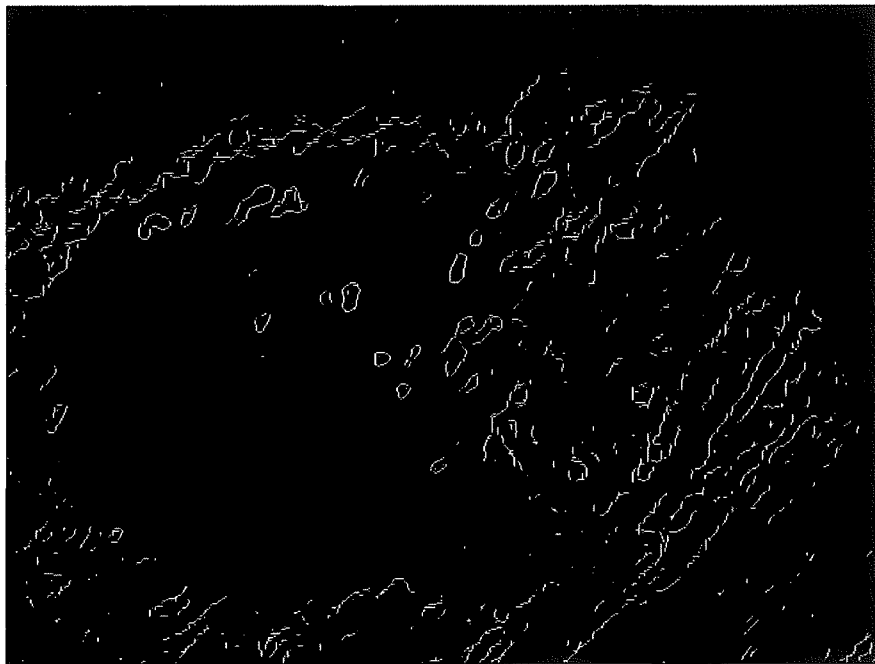
(a)



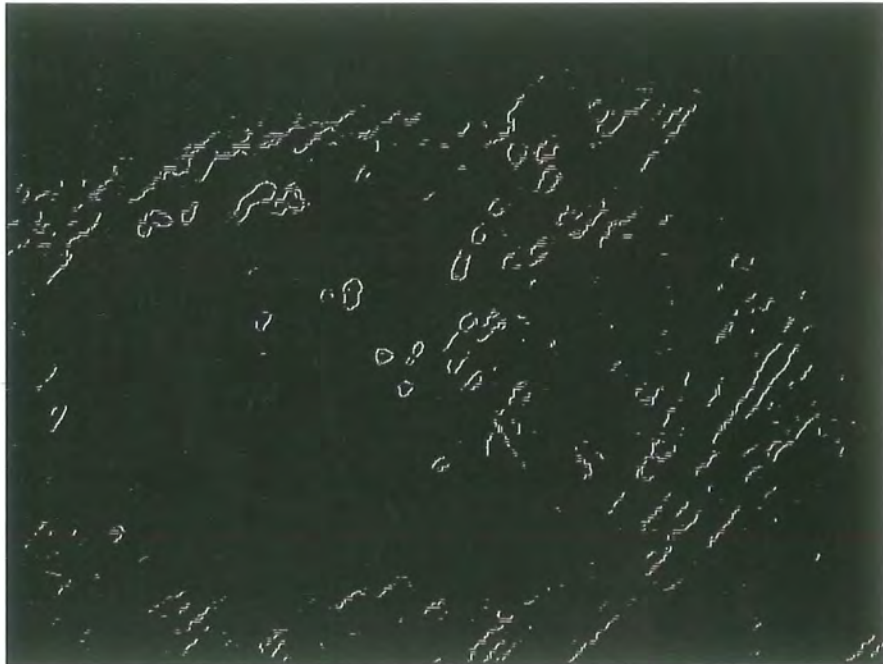
(b)



(c)



(d)



(e)



(f)

Figure 10: Image 7.jpg. (a) Result from Canny. (b) Result from LoG. (c) Result from Sobel. (d) Result from Prewitt. (e) Result from Roberts Cross. (f) Result from Skinseg.exe

5.2.2 Approach 2 – Manual tuning of edge parameters

In this approach, we apply edge detection with various range of threshold and sigma values in order to find which is the good range of each edge method that for determining a better lesion boundary.

The ideal range was determined by apply different thresholds for the edge detectors and using visual evaluation to determine the quality of the delineated edge. Various ranges of sigma values have also been applied for Canny and LoG operators in each image to determine a suitable range.

The resultant parameter range, compared to the automatically determined parameters is shown in Table 5 and Table 6.

Table 5: Result from Canny and Laplacian of Gaussian edge detection (default value and the good range of threshold and sigma value)

Image	CANNY EDGE DETECTOR				LAPLACIAN OF GAUSSIAN EDGE DETECTOR			
	Default value		Selected range		Default value		Selected range	
	Threshold	Sigma	Threshold	Sigma	Threshold	Sigma	Threshold	Sigma
1.jpg	0.09375 - 0.23438	1	0.430 - 0.447	1.72 - 1.78	0.0023803	2	0.0022 - 0.0025	2.1 - 2.2
2.jpg	0.04375 - 0.10938	1	0.0 10- 0.110	1.00 - 1.90	0.0029116	2	0.0010 - 0.0030	1.5 - 2.0
3.jpg	0.06875 - 0.17188	1	0.010 - 0.100	0.80 - 1.10	0.0026552	2	0.0020 - 0.0090	0.6 - 0.9
4.jpg	0.04375 - 0.10938	1	0.040 - 0.100	1.00 - 2.00	0.0020525	2	0.0010 - 0.0090	0.7 - 1.0
5.jpg	0.05625 - 0.14063	1	0.020 - 0.100	0.70 - 1.50	0.0033865	2	0.0010 - 0.0060	1.0 - 1.5
6.jpg	0.08125 - 0.20313	1	0.010 - 0.060	0.10 - 0.40	0.0054216	2	0.0010 - 0.0070	0.6 - 1.0
7.jpg	0.06875 - 0.17188	1	0.010 - 0.900	0.10 - 1.60	0.0026174	2	0.0010 - 0.0050	0.6 - 1.0
8.jpg	0.06587 - 0.17188	1	0.010 - 0.140	1.30 - 2.50	0.0026307	2	0.0010 - 0.0030	1.3 - 1.8
9.jpg	0.01875 - 0.04687	1	0.010 - 0.050	0.80 - 2.00	0.0015055	2	0.0010 - 0.0030	1.0 - 1.8
10.jpg	0.05625 - 0.14063	1	0.010 - 0.110	1.50 - 2.50	0.0023609	2	0.0020 - 0.0030	1.7 - 2.5
11.jpg	0.05000- 0.12500	1	0.010 - 0.050	2.30 - 2.70	0.0017181	2	0.0005 - 0.0009	2.3 -2.9
12.jpg	0.03750 - 0.09375	1	0.020 - 0.090	1.20 - 1.80	0.0020401	2	0.0020 - 0.0050	1.0 - 1.4
13.jpg	0.06250 - 0.15625	1	0.010 - 0.070	2.00 - 2.70	0.0042179	2	0.0040 - 0.0070	1.0 - 1.8
14.jpg	0.04375 - 0.10938	1	0.010 - 0.080	2.00 - 3.00	0.0021470	2	0.0020 - 0.0030	2.3 - 2.7
15.jpg	0.06250 - 0.15625	1	0.010 - 0.120	1.60 - 2.75	0.0022701	2	0.0200 - 0.0350	1.6 - 2.0
16.jpg	0.03750 - 0.09375	1	0.010 - 0.050	1.00 - 2.00	0.0033942	2	0.0010 - 0.0040	1.5 - 2.0

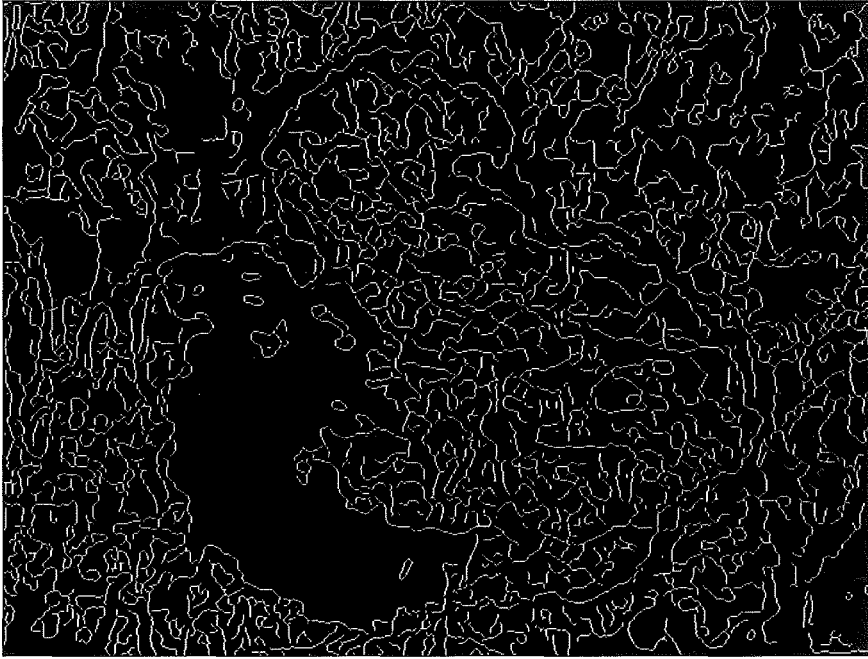
Table 6: Result from Sobel, Prewitt and Roberts-Cross edge detection (using default value and the good range of threshold value)

Image	SOBEL EDGE DETECTOR		PREWITT EDGE DETECTOR		ROBERTS CROSS EDGE DETECTOR	
	Default Threshold	Selected range	Default Threshold	Selected range	Default Threshold	Selected range
1.jpg	0.033466	0.029 - 0.032	0.033294	0.028 - 0.034	0.044658	0.028 - 0.035
2.jpg	0.049552	0.020 - 0.040	0.049314	0.020 - 0.040	0.063969	0.050 - 0.060
3.jpg	0.036241	0.010 - 0.030	0.036027	0.010 - 0.050	0.047898	0.010 - 0.030
4.jpg	0.039829	0.010 - 0.025	0.039668	0.010 - 0.030	0.051095	0.020 - 0.035
5.jpg	0.055883	0.030 - 0.060	0.055579	0.020 - 0.060	0.072266	0.050 - 0.070
6.jpg	0.082948	0.010 - 0.050	0.082346	0.010 - 0.040	0.107280	0.010 - 0.060
7.jpg	0.042146	0.010 - 0.029	0.041930	0.001 - 0.003	0.054343	0.001 - 0.020
8.jpg	0.041224	0.025 - 0.039	0.041032	0.025 - 0.040	0.054042	0.027 - 0.048
9.jpg	0.042757	0.010 - 0.020	0.042621	0.010 - 0.016	0.054363	0.010 - 0.030
10.jpg	0.035561	0.028 - 0.035	0.035382	0.025 - 0.035	0.046866	0.029 - 0.037
11.jpg	0.00281	0.024 - 0.027	0.027996	0.023 - 0.027	0.037329	0.025 - 0.035
12.jpg	0.042448	0.020 - 0.035	0.042311	0.010 - 0.032	0.054390	0.011 - 0.040
13.jpg	0.064986	0.030 - 0.060	0.064713	0.040 - 0.070	0.083181	0.050 - 0.070
14.jpg	0.049908	0.035 - 0.060	0.049355	0.035 - 0.050	0.067555	0.045 - 0.070
15.jpg	0.035127	0.026 - 0.037	0.034959	0.032 - 0.039	0.046148	0.027 - 0.037
16.jpg	0.050166	0.010 - 0.029	0.049689	0.010 - 0.030	0.067907	0.020 - 0.040

The results for one of the images (image 8) are shown in Figure 11 and 12. For each detection method, there is a range of threshold/sigma values which could produce a better edge compare to the default value. According to the result for Sobel, Prewitt and Roberts Cross the ideal range of threshold value is often smaller than the default value. In making this statement it is important to define our criterion for analysing the threshold result is the completeness and continuity of the lesion border. The effect of lowering the threshold from the automatically generated Matlab value is that weak edges in the lesion border are found. Conversely, more edges caused by background noise are also detected when the threshold is decreased. Our argument is that as long as the lesion border in the processed image is accurate and complete, background noise can be eliminated by post-processing.

On the other hand, Canny and LoG have a different effect in selecting the good range of threshold and sigma for different image input. The complication arises as there are two parameters to tune for both Canny and LoG. Results from Table 5 shows that there seem to be no rule to select a good range of threshold and

sigma value with respect to the default value generated by Matlab for Canny and LoG.

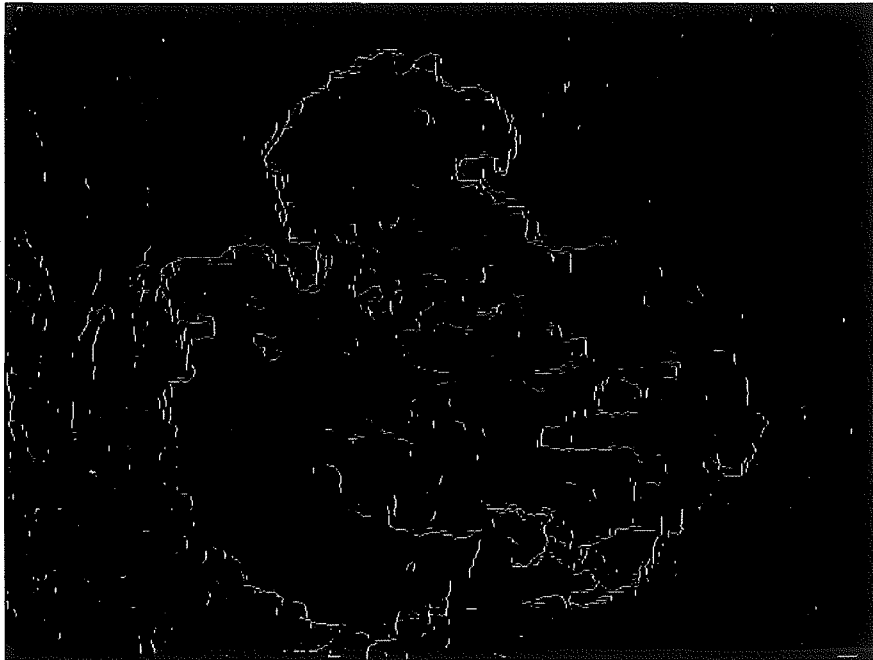


(a)

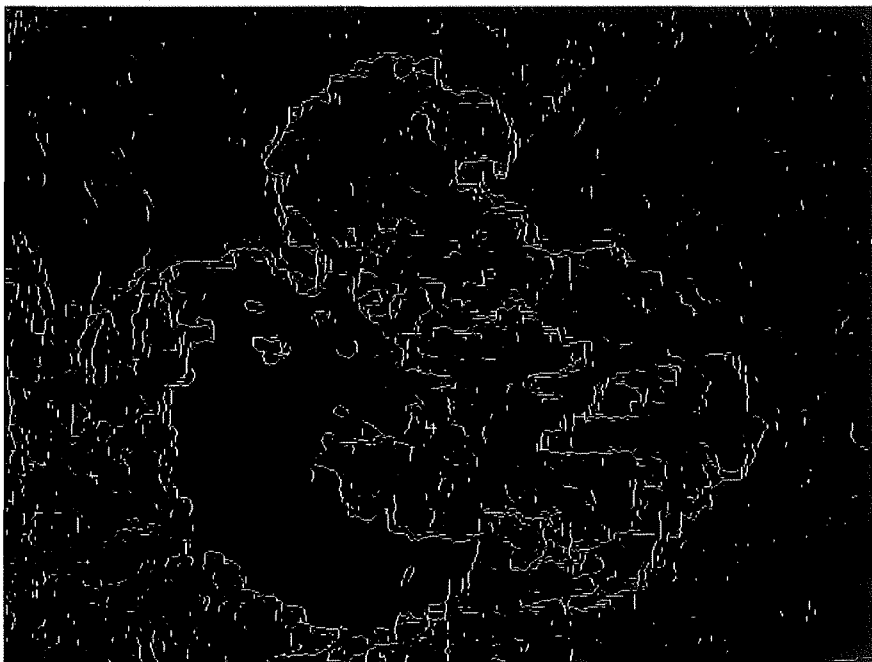


(b)

Figure 11: (a) Image 8.jpg detect by Canny edge detector with default value.
(b) Image 8.jpg detect by Canny method (threshold = 0.02 and sigma = 2.5).



(a)



(b)

Figure 12: (a) Image 8.jpg- detect by Sobel edge detection with default value
(b) Image 8.jpg – detect by Sobel edge detection with threshold = 0.032

6 Conclusion

In this Thesis, we have demonstrated method of segmentation for skin cancer that an implementation based on the Xu et. al. (1999) method. Results were generated from different edge detectors based on the default value and selected range of threshold and sigma values have been compared. We also compared these results to its original image and also the results that are obtained by the program Skinseg.exe from Xu. Experiments have shown that the approach from Xu et. al. (1999) can only apply to some types of images, which have unique background colour and distinctive from the lesion (foreground) colour. This is not suitable for application to an image that does not have a uniform background and foreground colour. Thus, images captured by VGA camera phone are not suited for lesion detection using the Xu's method.

Result from this experiment shows that it is not possible to adapt an existing lesion segmentation algorithm for use in mobile phone camera images. The method is derived from Xu et. al. (1999) is not sufficiently to detect skin lesion which have been taken by mobile camera, especially using VGA camera. Some works need to be done to reduce noise in image.

Given the lower resolution of mobile phone camera images, the Canny edge detection method is more suitable for the existing lesion segmentation algorithm compare to other methods. Canny method has detected clear edge for lesion than other. Sobel is the next method should also be taken into account for detecting skin cancer. LoG has given a fair result. Prewitt is not as good as LoG but still better than Robert which contain too much noise.

7 Future work

From the result of this research, there are a few things for future investigations: Firstly, we would like to find a better way for pre-processing the image. One approach can be using foreground colour to calculate distance colour of pixel in image. This is done by:

Find the top 5% values of pixels in the image (since the colour of foreground (the mole) is normally has higher value than in the background (the surrounding skin)).

Then running a scan through whole image to find any pixel that is in range of top 5% and mark it as '1', while others in lower range will be marked as '0'. This mapping method will create a binary image.

Finally, apply the 'regionprops' function in Matlab to find biggest continuous area of 1s in the resultant binary image. That should be the binary map of the lesion.

After further work we would like to achieve is implement a robustness mobile application that can run these tests on the camera phone. This will allow user to analyse their moles immediately after capturing them without sending the image to a laboratory.

8 Glossaries

8.1 Image processing terminologies

These image processing definitions have retrieved from Imagelabs website (Imagelabs, 2006).

Algorithm	A set of well-defined rules or procedures for solving a problem or providing an output from a specific set of inputs.
Boundary	The line formed by the joining of two image regions, each having a different light intensity. The edge of a region or object.
Convolution	Superimposing a $m \times n$ operator (usually a 3×3 or 5×5 mask) over an area of the image, multiplying the points together, summing the results to replace the original pixel with the new value. This operation is often performed on the entire image to enhance edges, features, remove noise and other filtering operations.
Edge	A change in pixel values exceeding some threshold amount. Edges represent borders between regions on an object or in a scene.
Filter	A device or process that selectively transmits frequencies. In optics, the material either reflects or absorbs certain wavelengths of light, while passing others
Gradient	The rate of change of pixel intensity (first derivative).
Gray level	A quantized measurement of image irradiance (brightness), or other pixel property typically in the range between pure white and black.
Grayscale Image	An image consisting of an array of pixels which can have more than two values. Typically, up to 256 levels (8 bits) are used for each pixel
Intensity	The relative brightness of a portion of the image or illumination source.
Kernel/mask	1) Setting portions of an image are neighbors to a constant value; 2) A filter matrix used as a convolution operator; 3) A logical or physical structure placed in an optical system to prevent viewing or passing of information in a certain spatial or

frequency region.

Median filter	A method of image smoothing which replaces each pixel value with the median grayscale value of its immediate neighbors.
Noise	Irrelevant or meaningless data resulting from various causes unrelated to the source. Random, undesired video signals.
Orientation	The angle or degree of difference between the object coordinate system major axis relative to a reference axis as defined in a 3D measurement space.
Pixel	An acronym for "picture element." The smallest distinguishable and resolvable area in an image. The discrete location of an individual photo-sensor in a solid state camera.
Region	Area of an image. Also called a region of interest for image processing operations
Resolution, Image	The number of rows and columns of pixels in an image.
RGB	An acronym for the Red-Green-Blue color space. This three primary color system is used for video color representation.
Segmentation	The process of dividing a scene into a number of individual objects or contiguous regions, differentiating them from each other and the image background.
Shape	An object characteristic, often referring to its spatial contour.
Thresholding	The process of converting gray scale image into a binary image. If the pixel's value is above the threshold, it is converted to white. If below the threshold, the pixel value is converted to black.
VGA	An acronym for Video Graphics Array. The IBM video display standard of 16 colors.

8.2 Skin cancer terminologies

Definitions retrieved from Medicinenet.com (Medicinenet.com , 2007)

Basal cell carcinoma	The most common type of skin cancer, a disease in which the cancer cells resemble the basal cells of the epidermis, the outer layer of the skin.
-----------------------------	--

Benign	Not cancer. Not malignant. A benign tumor does not invade surrounding tissue or spread to other parts of the body. A benign tumor may grow but it stays put (in the same place).
Biopsy	The removal of a sample of tissue for purposes of diagnosis. (Many definitions of "biopsy" stipulate that the sample of tissue is removed for examination under a microscope. This may or may not be the case. The diagnosis may be achieved by other means such as by analysis of chromosomes or genes).
Cancer	An abnormal growth of cells which tend to proliferate in an uncontrolled way and, in some cases, to metastasize (spread).
Curable	Amenable to a cure, capable of being cured, to being healed and made well. Most skin cancers, fortunately, are curable. From the word cure, from the Latin cura meaning care, concern or attention.
Dermatology	<ol style="list-style-type: none"> 1. The branch of medicine concerned with the diagnosis, treatment, and prevention of diseases of the skin, hair, nails, oral cavity and genitals. 2. Sometimes also, cosmetic care and enhancement. Dermatology is literally the study of the skin.
Lesion	Pronounced "lee-sion" with the emphasis on the "lee," a lesion can be almost any abnormality involving any tissue or organ due to any disease or any injury.
Melanin	A skin pigment (substance that gives the skin its color). Dark-skinned people have more melanin than light-skinned people. Melanin also acts as a sunscreen and protects the skin from ultraviolet light.
Melanoma	The most dangerous form of skin cancer, a malignancy of the melanocyte, the cell that produces pigment in the skin. Melanoma is most common in people with fair skin, but can occur in people with all skin colors. Most melanomas present as a dark, mole-like spot that spreads and, unlike a mole, has an irregular border. The tendency toward melanoma may be inherited, and the risk increases with overexposure to the sun and sunburn.
Mole	<ol style="list-style-type: none"> 1. In dermatology, a pigmented spot on the skin, Also called a nevus. 2. In gynecology, a mass within the uterus formed of partly developed products of conception. 3. In biochemistry, the molecular weight of a substance. In more technical terms, a mole is the number of atoms in exactly 12 g of carbon-12. Mole in this sense is short for molecular weight.

Nonmelanoma skin cancer

Skin cancer that does not involve melanocytes. Basal cell cancer and squamous cell cancer are nonmelanoma skin cancers.

Pigment

A substance that gives color to tissue. Pigments are responsible for the color of skin, eyes, and hair.

Squamous cell carcinoma

Cancer that begins in squamous cells -- thin, flat cells that look under the microscope like fish scales. Squamous cells are found in the tissue that forms the surface of the skin, the lining of hollow organs of the body, and the passages of the respiratory and digestive tracts. Squamous cell carcinomas may arise in any of these tissues.

9 References

- AAD. (2006). Melanoma in African Americans [Electronic Version]. Retrieved 21st February from http://www.aad.org/public/News/NewsReleases/Press+Release+Archives/Skin+Cancer+and+Sun+Safety/Melanoma_AfricanAmericans.htm.
- Avecmobile. (2004a). Digital camera or camera phone? Retrieved 20th July, 2007, from <http://www.avecmobile.com/index.php?id=218>
- Avecmobile. (2004J). Eight things you can do with a camera phone. Retrieved 20th July, 2007, from <http://www.avecmobile.com/index.php?id=281>
- Brannon, H. (2007). The ABCD's of Melanoma. Retrieved 20th July, 2007, from http://dermatology.about.com/od/melanoma/ss/ABCD_SBS.htm
- CAIRNS. (2007, 28th August). Broadband critical to Australia's economic development: Mickel. Retrieved 31st August, 2007, from http://www.sd.qld.gov.au/dsdweb/v3/guis/templates/content/gui_cue_cntnlist.cfm?id=1068
- Canny, J. F. (1986). A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6), 679 - 698
- Cascio, J. (2005). Cellular Telemedicine, Revisited. Retrieved 15th July, 2007, from <http://www.worldchanging.com/archives//003852.html>
- CIGNA. (2006). Malignant Melanoma Surveillance Technologies [Electronic Version]. *Medical Necessity Guidelines*, 0240, 7. Retrieved 15th November 2006 from www.cigna.com/.../medical/mm_0240_coveragepositioncriteria_photo_surveil_early_detect_melanoma.pdf.
- Claypoole, R., Levis, J., Blashyam, S., & Kelly, K. (1997). Other methods of edge detection [Electronic Version]. Retrieved 15th October 2006 from <http://www.owl.net.rice.edu/~elec539/Projects97/morphjrks/>.
- Commens, A. C. (2007). Skin cancer: changing paradigms of practice and medical education [Electronic Version]. Retrieved 10th August 2007 from http://www.mja.com.au/public/issues/187_04_200807/com10732_fm.html
- Denton, W. E., Duller, A. W. G., & Fish, P. J. (1995). Boundary detection for skin lesions: an edge focusing algorithm [Electronic Version]. *Image Processing and its Application, 1995. Fifth International Conference*, 399-403. Retrieved 10th August 2007 from IEEE.
- Dermnet. (2007). Melanoma (Skin Cancer), Nevi, and Moles. Retrieved 20th July, 2007, from <http://www.dermnet.com/moduleIndex.cfm?moduleID=14>

- Dolianitis, C., Kelly, J., Wolfe, R., & Simpson, P. (2005). Comparative performance of 4 dermoscopic algorithms by nonexperts for the diagnosis of melanocytic lesions. [Electronic Version]. *Arch Dermatol*, 141, 1008-1014. Retrieved 20th July 2007 from http://www.ncbi.nlm.nih.gov/sites/entrez?Db=PubMed&Cmd=ShowDetailView&TermToSearch=16103330&ordinalpos=1&itool=EntrezSystem2.PEntrez.Pubmed.Pubmed_ResultsPanel.Pubmed_RVAbstractPlus.
- Eminovic, N., Keizer, N. F. d., Bindels, P. J. E., & Hasman, A. (2007). Maturity of Teledermatology Evaluation Research: A Systematic Literature Review [Electronic Version]. *The British Journal of Dermatology*, 156, 412-419. Retrieved 21th June 2007 from <http://www.medscape.com/viewarticle/553881>.
- Ericsson. (2007). Ericsson and Telstra achieve world first 200 km cell range in nationwide network with up to 14.4 Mbps. Retrieved 20th June, 2007, from <http://www.ericsson.com/ericsson/press/releases/20070212-1104433.shtml>
- Fayed, L. (2006). Skin Cancer/Melanoma Photo Gallery. Retrieved 20th July, 2007, from <http://cancer.about.com/od/skincancermelanoma/ig/Skin-Cancer-Photo-Gallery/index.htm>
- Fisher, R., Perkins, S., Walker, A., & Wolfart, E. (2003). Image Processing Operator Worksheets [Electronic Version]. Retrieved 20th October 2006 from <http://homepages.inf.ed.ac.uk/rbf/HIPR2/wksheets.htm>
- Getreuer, P. (2005). Color Space Converter. Retrieved 20th August, 2007, from <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=7744&objectType=file>
- Goshtasby, A. (1998). Downloading Instructions. Retrieved 25th August, 2007, from <http://www.cs.wright.edu/~agoshtas/skinseg.html>
- Green, B. (2002). Canny Edge Detection Tutorial. Retrieved 12th August, 2006, from http://www.pages.drexel.edu/~weg22/can_tut.html
- Guo, W., & Aslandogan, Y. A. (2003). Mining Skin Lesion Images with Spatial Data Mining Methods [Electronic Version], 21. Retrieved 20th June 2007 from www.cse.uta.edu/research/publications/Downloads/CSE-2003-20.pdf
- Hansen, T. R., Eklund, J. M., Sprinkle, J., Bajcsy, R., & Sastry, S. (2005). *Using Smart Sensors and a Camera Phone to Detect and Verify The Fall of Elderly Persons*. Retrieved 25th November 2006, from www.eecs.berkeley.edu/~eklund/Papers/EMBEC2005_FallPhone.pdf
- Ilea, D. E., & Whelan, P. F. (2006). Automatic Segmentation of Skin Cancer Images using Adaptive Color Clustering [Electronic Version], 4. Retrieved 20th June 2007 from <http://www.vsg.dcu.ie/papers/>.

- Imagelabs. (2006). Welcome to Image Labs International's Glossary! Retrieved 10th July, 2007, from <http://www.imagelabs.com/library/glossary.php?requestUrl=/gl.shtml#E>
- InfoTrends. (2005). Worldwide Mobile Imaging: Stimulating Demand and Usage. Retrieved 20th August, 2007, from <http://www.infotrends-rgi.com/public/Content/Multiclients/mobileimaging05.html>
- Kadurina, M. I., & Mitoff, K. K. (2005). Teledermatology [Electronic Version]. *medicine Virtual Communications*, 170-174. Retrieved 12th June 2007 from http://www.diagnosisp.com/dp/journals/view_pdf.php?
- Kreutz, M., Anschutz, M., Gehlen, S., Grunendick, T., & Hoffmann, K. (2001). Automated Diagnosis of Skin Cancer Using Digital Image Processing and Mixture-of-Experts [Electronic Version], 357-361. Retrieved 12th June 2007 from <http://citeseer.ist.psu.edu/631919.html>.
- Lavelle, P. (2003). Skin cancer fact file. Retrieved 20th July, 2007, from http://abc.net.au/health/library/skincancer_ff.htm
- Marshall, D. (1994). Second Order Methods. *Vision Systems* Retrieved 15th October, 2006, from http://www.cs.cf.ac.uk/Dave/Vision_lecture/node29.html#SECTION00153200000000000000
- Medicinenet.com. (2007). Skin Cancer Glossary of Terms. Retrieved 20th August, 2007, from http://www.medicinenet.com/skin_cancer/glossary.htm
- Mezghani, N., Deschenes, S., Godbout, B., Branchaud, D., & Guise, J. A. d. (2006). Spinal Vertebrae Edge Detection by Anisotropic Filtering and A Local Canny-Deriche Edge Detector [Electronic Version]. *IEEE International symposium on Image/video communications*, 4. Retrieved 10th January 2007 from <http://www.etsmtl.ca/zone2/recherche/labo/lio/fiche/neilamezghani/>.
- MyoClinic.com. (2006). Slide show: Melanoma pictures to help identify skin cancer. Retrieved 20th July, 2007, from <http://www.mayoclinic.com/health/melanoma/DS00575/RETURNTOOBJID=9B73D513-F3AC-4443-AB0F2DBF14B54664&RETURNLINK=1&slide=1>
- Neoh, H. S., & Hazanchuk, A. (2004). Adaptive Edge Detection for Real-Time Video Processing using FPGAs [Electronic Version], 7. Retrieved 20th September 2006 from <http://www.altera.com/literature/cp/gspix/edge-detection.pdf>.
- O'Keefe, M. (2004). Worldwide Camera Phone Sales to Reach Nearly 150 Million in 2004, Capturing 29 Billion Digital Images. Retrieved 15th November, 2006, from <http://www.infotrends-rgi.com/home/Press/itPress/2004/3.11.04.html>

- Oakley, A. (2005). Teledermatology Today. *Health Care and Informatics Review Online* Retrieved 26th June, 2007, from <http://hcro.enigma.co.nz/website/index.cfm?fuseaction=articledisplay&featureid=030605>
- Owens, R. (1997). Index of /rbf/CVonline/LOCAL_COPIES/OWENS [Electronic Version]. Retrieved 25th October 2006 from http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/.
- Pathak, S. D., Haynor, D. R., & Kim, Y. (2000). Edge-guided boundary delineation in prostate ultrasound images [Electronic Version]. *Medical Imaging, IEEE Transactions on*, 19, 1211-1219. Retrieved 15th February 2007.
- Qi, H., & Kuruganti, P. T. (2006). Detecting Breast Cancer from Thermal Infrared Images by Asymmetry Analysis [Electronic Version], 22. Retrieved 20th January 2007 from http://www.aicip.ece.utk.edu/publication/06bme_asymmetry.pdf
- Roushdy, M. (2006). Comparative Study of Edge Detection Algorithms Applying on the Grayscale Noisy Image Using Morphological Filter [Electronic Version]. *GVIP*, 6, 17-23. Retrieved December from <http://www.icgst.com/gvip/Volume6/Issue4/>.
- Rubino, M. T. (2003). Edge Detection Algorithms [Electronic Version]. Retrieved 20th September 2006 from <http://www.ccs.neu.edu/home/mtrubs/html/EdgeDetection.html>.
- Sharp. (2000). Sharp History. Retrieved 22nd January, 2007, from http://www.sharp-world.com/corporate/info/his/h_company/2000/index.html
- Talbot, H., & Bischof, L. (2003). An overview of the Polartechncs SolarScan melanoma diagnosis alorithms [Electronic Version], 33-38. Retrieved 12th June 2007 from <http://www.aprs.org.au/wdic2003/CDROM/Program.html>.
- TCCA (2004). All about skin cancer [Electronic Version]. Retrieved 21st February 2007 from <http://www.cancer.org.au/content.cfm?randid=960742>.
- Wang, R. (2004). Canny Edge Detection. Retrieved 20th January, 2007, from <http://fourier.eng.hmc.edu/e161/lectures/canny/node1.html>
- Xiong, G. (2005). Gradient using first order derivative of Gaussian. Retrieved 25th August, 2007, from <http://www.mathworks.com/matlabcentral/fileexchange/loadFile.do?objectId=8060&objectType=file>
- Xu, L., Jackowski, M., Goshtasby, A., Roseman, D., Bine, S., Yu, C., et al. (1999). Segmentation of skin cancer images [Electronic Version]. *Image and Vision Computing*, 17, 65-74. Retrieved 12th June 2007 from <http://noodle.med.yale.edu/~mjack/papers/>.

- Xuan, J., Adali, T., & Wang, Y. (1995). Segmentation of magnetic resonance brain image: integrating region growing and edge detection [Electronic Version]. *Image Processing, 1995. Proceedings, International Conference on*, 3, 544-547. Retrieved 10th September 2006 from IEEE.
- Zambanini, S., Langs, G., Sablanig, R., & Maier, H. (2007). Automatic Robust Registration of Cutaneous Hemangiomas for Follow-up Examinations [Electronic Version], 8. Retrieved 10th September 2007 from <http://www.prip.tuwien.ac.at/~zamba/pubs/>
- Ziou, D., & Tabbone, S. (1998). Edge Detection Techniques - An Overview [Electronic Version]. *International Journal of Pattern Recognition and Image Analysis*, 8, 537-559. Retrieved 12th November 2006 from <http://citeseer.ist.psu.edu/ziou98edge.html>.
- Zouridakis, G., Doshi, M., Duvic, M., & Mullani, N. A. (2005). Transillumination imaging for early skin cancer detection [Electronic Version], 18. Retrieved 23rd March 2007 from <http://www.cs.uh.edu/docs/preprint/uh-cs-05-05.pdf>.

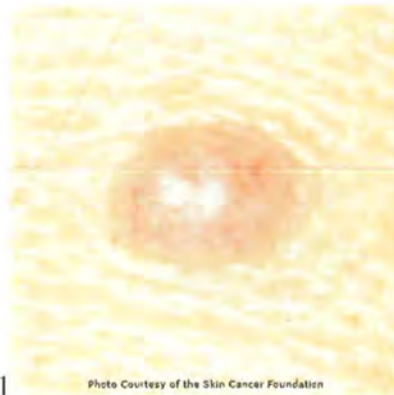
10 Appendix

10.1 Appendix A

10.1.1 SKIN LESSION IMAGES

1) These pictures of normal and cancer moles are retrieved from About.com website (Brannon, 2007). These original images have been scale down fifty percent for displaying.

NORMAL MOLES





3

©2007 Heather Brannon, MD licensed to About.com, inc.

CANCER MOLES



4

Photo Courtesy of the Skin Cancer Foundation



5

Photo Courtesy of the Skin Cancer Foundation



6

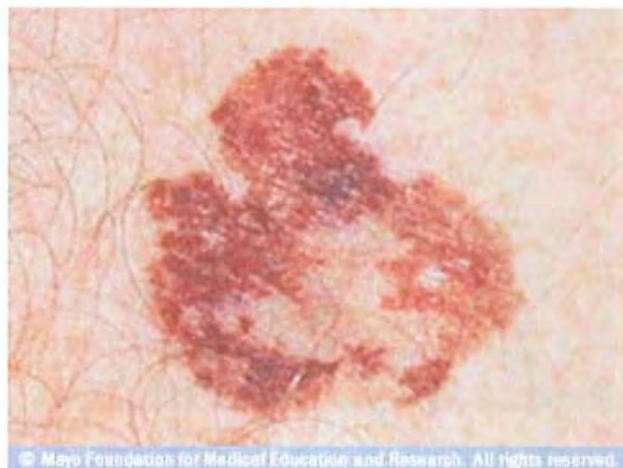
Photo Courtesy of the Skin Cancer Foundation

2) Skin cancer images retrieved from Myoclinic.com (MyoClinic.com, 2006) illustrate ABCD rule



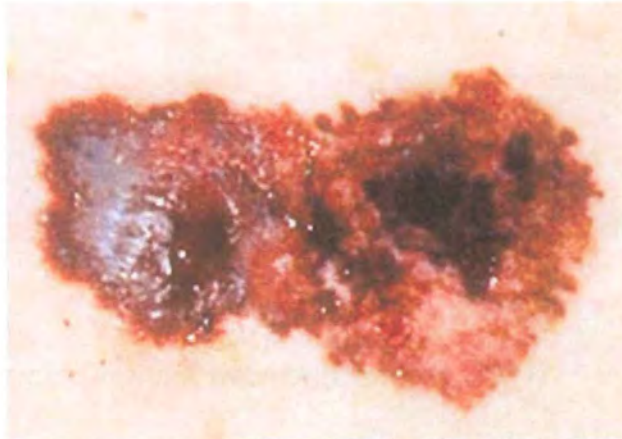
7

© Mayo Foundation for Medical Education and Research. All rights reserved.



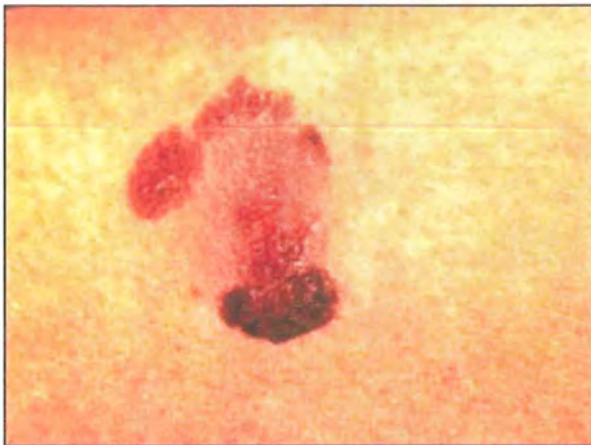
8

© Mayo Foundation for Medical Education and Research. All rights reserved.



9 © Mayo Foundation for Medical Education and Research. All rights reserved.

3) Photo of Melanoma moles – retrieved from About.com (Fayer, 2006).



10



11



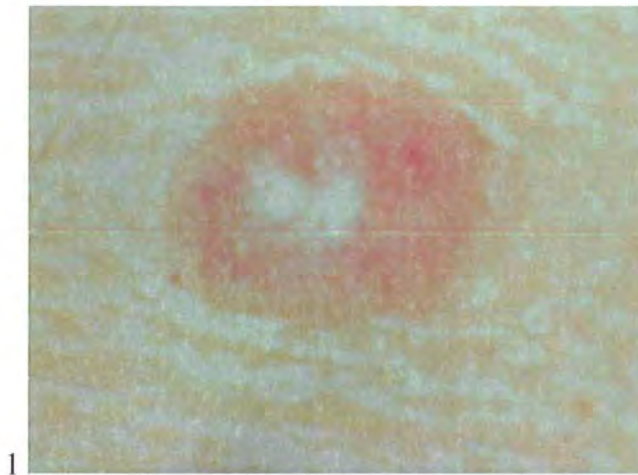
4) Images are retrieved from Dermnet website (Dermnet, 2007)



10.1.2 Images take from camera phone

These images above were printed by Lanier colour printer then the printed outs are captured by Nokia 6066 with VGA resolution (640x480) without flash. The images are scale down at fifty percent (320 x 240 pixel, 8.43cm x 11.29cm) for display. However, the images for testing will remained at original size (640 x 480).

NORMAL MOLES





3

CANCER MOLES



4



5



6



7



8



9



10



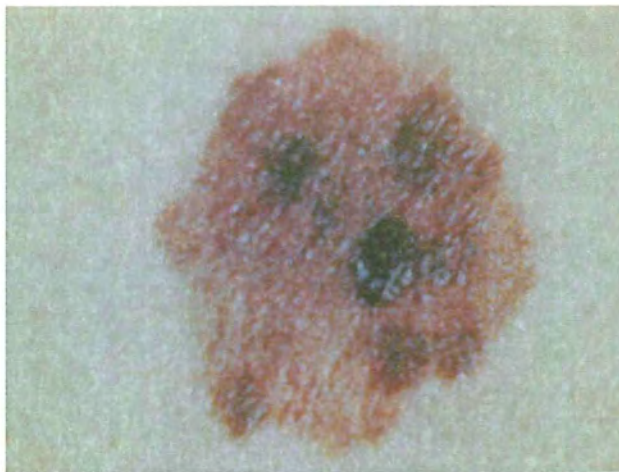
11



12



13

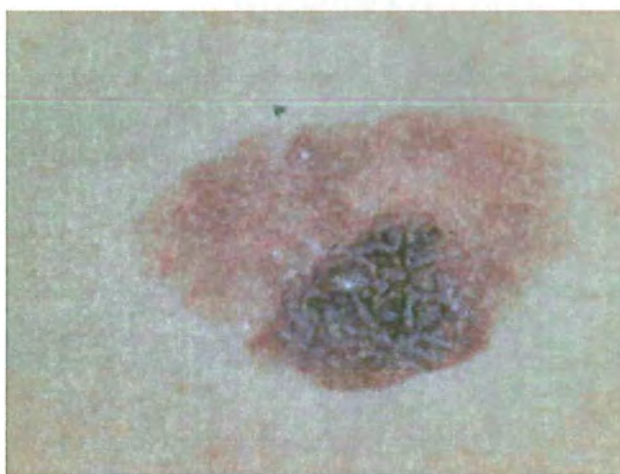


14

15



16



10.2 Appendix B (Source code)

10.2.1 Function pre-process.m

```
function u = pre-process(I)

% Function pre-processing image
% =====
% TRANSFORM COLOR FROM RGB -> XYZ -> CIELAB
% This function is derived from colorspace created by Getreuer, P. (2005).
% =====
Ixyz=colorspace('xyz<-rgb',I);           % convert RGB image into xyz
Icie=colorspace('cie<-xyz',Ixyz);       % convert xyz into CIELAB

% =====
% CROPPING FOUR CORNERS TO GENERATE COLOR OF BACKGROUND
% WINDOW SIZE 10x10 PIXELS
% =====
I1=imcrop(Icie,[1,1,9,9]);
I2=imcrop(Icie,[630,1,9,9]);
I3=imcrop(Icie,[1,470,9,9]);
I4=imcrop(Icie,[630,470,9,9]);

% =====
% CALCULATE MEDIAN OF FOUR CORNER
% =====
M1=median(median(I1));
M2=median(median(I2));
M3=median(median(I3));
M4=median(median(I4));
Mn = [M1; M2; M3; M4];
Mm1 = median(Mn);

% =====
% APPLY EUCLIDEAN DISTANCE COLOR TO TRANSFORM COLOR
% IMAGE INTO INTENSITY IMAGE
% =====
L = Icie(:,:,1);
a = Icie(:,:,2);
b = Icie(:,:,3);
delL = L - Mm1(:,:,1);
dela = a - Mm1(:,:,2);
delb = b - Mm1(:,:,3);
N = sqrt(power(delL,2) + power(dela,2) + power(delb,2));

% =====
```

```

% CALCULATE STANDARD DEVIATION FOR 2D KERNEL GAUSSIAN
% =====
S1 = std(std(I1));
S2 = std(std(I2));
S3 = std(std(I3));
S4 = std(std(I4));
Sm = std([S1;S2;S3;S4]);
S = std(Sm);

%=====
% APPLY GAUSSIAN FILTER
% gaussgradient2 is a modified of gaussgradient function created by Xiong, G. (2005).
% =====
Ng2 = gaussgradient2(N,S);

% =====
% CLAMPING INTENSITY IMAGE
% =====
coords = find(Ng2>1);    %finding intensity value > 1, convert it into 1
Ng2(coords) = 1;

```

10.2.2 Function colorspace.m

```

function varargout = colorspace(Conversion,varargin)
%COLORSPACE Convert a color image between color representations.
% B = COLORSPACE(S,A) converts the color representation of image A
% where S is a string specifying the conversion. S tells the source and destination color
% spaces, S = 'dest<-src', or alternatively, S = 'src->dest'. Supported color spaces are
%
% 'RGB'          R'G'B' Red Green Blue (ITU-R BT.709 gamma-corrected)
% 'YPbPr'        Luma (ITU-R BT.601) + Chroma
% 'YCbCr'/'YCC'  Luma + Chroma ("digitized" version of Y'PbPr)
% 'YUV'          NTSC PAL Y'UV Luma + Chroma
% 'YIQ'          NTSC Y'IQ Luma + Chroma
% 'YDbDr'        SECAM Y'DbDr Luma + Chroma
% 'JPEGYCbCr'    JPEG-Y'CbCr Luma + Chroma
% 'HSV'/'HSB'    Hue Saturation Value/Brightness
% 'HSL'/'HLS'/'HSI' Hue Saturation Luminance/Intensity
% 'XYZ'          CIE XYZ
% 'Lab'          CIE L*a*b* (CIELAB)
% 'Luv'          CIE L*u*v* (CIELUV)
% 'Lch'          CIE L*ch (CIELCH)
%
% All conversions assume 2 degree observer and D65 illuminant. Color

```

```

% space names are case insensitive. When R'G'B' is the source or destination, it can be
% omitted. For example 'yuv<-' is short for 'yuv<-rgb'.
%
% MATLAB uses two standard data formats for R'G'B': double data with
% intensities in the range 0 to 1, and uint8 data with integer-valued
% intensities from 0 to 255. As MATLAB's native datatype, double data is
% the natural choice, and the R'G'B' format used by colorspace. However,
% for memory and computational performance, some functions also operate
% with uint8 R'G'B'. Given uint8 R'G'B' color data, colorspace will
% first cast it to double R'G'B' before processing.
%
% If A is an Mx3 array, like a colormap, B will also have size Mx3.
%
% [B1,B2,B3] = COLORSPACE(S,A) specifies separate output channels.
% COLORSPACE(S,A1,A2,A3) specifies separate input channels.

% Pascal Getreuer 2005-2006

%% Input parsing %%
if nargin < 2, error('Not enough input arguments.');
```

```

[SrcSpace, DestSpace] = parse(Conversion);

if nargin == 2
    Image = varargin{1};
elseif nargin >= 3
    Image = cat(3, varargin{:});
else
    error('Invalid number of input arguments.');
```

```

end

FlipDims = (size(Image,3) == 1);

if FlipDims, Image = permute(Image,[1,3,2]); end
if ~isa(Image,'double'), Image = double(Image)/255; end
if size(Image,3) ~= 3, error('Invalid input size.');
```

```

end

SrcT = gettransform(SrcSpace);
DestT = gettransform(DestSpace);

if ~ischar(SrcT) & ~ischar(DestT)
    % Both source and destination transforms are affine, so they
    % can be composed into one affine operation
    T = [DestT(:,1:3)*SrcT(:,1:3), DestT(:,1:3)*SrcT(:,4)+DestT(:,4)];
    Temp = zeros(size(Image));
    Temp(:,,1) = T(1)*Image(:,,1) + T(4)*Image(:,,2) + T(7)*Image(:,,3) + T(10);
    Temp(:,,2) = T(2)*Image(:,,1) + T(5)*Image(:,,2) + T(8)*Image(:,,3) + T(11);
    Temp(:,,3) = T(3)*Image(:,,1) + T(6)*Image(:,,2) + T(9)*Image(:,,3) + T(12);
    Image = Temp;
end

```

```

elseif ~ischar(DestT)
    Image = rgb(Image,SrcSpace);
    Temp = zeros(size(Image));
    Temp(:,:,1) = DestT(1)*Image(:,:,1) + DestT(4)*Image(:,:,2) + DestT(7)*Image(:,:,3) +
    DestT(10);
    Temp(:,:,2) = DestT(2)*Image(:,:,1) + DestT(5)*Image(:,:,2) + DestT(8)*Image(:,:,3) +
    DestT(11);
    Temp(:,:,3) = DestT(3)*Image(:,:,1) + DestT(6)*Image(:,:,2) + DestT(9)*Image(:,:,3) +
    DestT(12);
    Image = Temp;
else
    Image = feval(DestT,Image,SrcSpace);
end

%%%% Output format %%%
if nargin > 1
    varargout = {Image(:,:,1),Image(:,:,2),Image(:,:,3)};
else
    if FlipDims, Image = permute(Image,[1,3,2]); end
    varargout = {Image};
end

return;

function [SrcSpace, DestSpace] = parse(Str)
% Parse conversion argument

if isstr(Str)
    Str = lower(strep(strep(Str,'-','),' ',''));
    k = find(Str == '>');

    if length(k) == 1 % Interpret the form 'src->dest'
        SrcSpace = Str(1:k-1);
        DestSpace = Str(k+1:end);
    else
        k = find(Str == '<');

        if length(k) == 1 % Interpret the form 'dest<-src'
            DestSpace = Str(1:k-1);
            SrcSpace = Str(k+1:end);
        else
            error(['Invalid conversion, "',Str,'"']);
        end
    end
end

SrcSpace = alias(SrcSpace);
DestSpace = alias(DestSpace);

```

```

else
    SrcSpace = 1;          % No source pre-transform
    DestSpace = Conversion;
    if any(size(Conversion) ~= 3), error('Transformation matrix must be 3x3.');
```

```

end
return;

function Space = alias(Space)
Space = strrep(Space,'cie','');
```

```

if isempty(Space)
    Space = 'rgb';
end
```

```

switch Space
case {'ycbcr','ycc'}
    Space = 'ycbcr';
case {'hsv','hsb'}
    Space = 'hsv';
case {'hsl','hsi','hls'}
    Space = 'hsl';
case {'rgb','yuv','yiq','ydbdr','ycbcr','jpegycbcr','xyz','lab','luv','lch'}
    return;
end
return;
```

```

function T = gettransform(Space)
% Get a colorspace transform: either a matrix describing an affine transform,
% or a string referring to a conversion subroutine
switch Space
case 'ypbpr'
    T = [0.299,0.587,0.114,0;-0.1687367,-0.331264,0.5,0;0.5,-0.418688,-0.081312,0];
case 'yuv'
    % R'G'B' to NTSC/PAL YUV
    % Wikipedia: http://en.wikipedia.org/wiki/YUV
    T = [0.299,0.587,0.114,0;-0.147,-0.289,0.436,0;0.615,-0.515,-0.100,0];
case 'ydbdr'
    % R'G'B' to SECAM YDbDr
    % Wikipedia: http://en.wikipedia.org/wiki/YDbDr
    T = [0.299,0.587,0.114,0;-0.450,-0.883,1.333,0;-1.333,1.116,0.217,0];
case 'yiq'
    % R'G'B' in [0,1] to NTSC YIQ in [0,1];[-0.595716,0.595716];[-0.522591,0.522591];
    % Wikipedia: http://en.wikipedia.org/wiki/YIQ
    T = [0.299,0.587,0.114,0;0.595716,-0.274453,-0.321263,0;0.211456,-
0.522591,0.311135,0];
case 'ycbcr'
```

```

% R'G'B' (range [0,1]) to ITU-R BRT.601 (CCIR 601) Y'CbCr
% Wikipedia: http://en.wikipedia.org/wiki/YCbCr
% Poynton, Equation 3, scaling of R'G'B' to Y'PbPr conversion
T = [65.481,128.553,24.966,16;-37.797,-74.203,112.0,128;112.0,-93.786,-18.214,128];
case 'jpegybcr'
% Wikipedia: http://en.wikipedia.org/wiki/YCbCr
T = [0.299,0.587,0.114,0;-0.168736,-0.331264,0.5,0.5;0.5,-0.418688,-
0.081312,0.5]*255;
case {'rgb','xyz','hsv','hsl','lab','luv','lch'}
T = Space;
otherwise
error(['Unknown color space, "',Space,'"']);
end
return;

```

```

function Image = rgb(Image,SrcSpace)
% Convert to Rec. 709 R'G'B' from 'SrcSpace'
switch SrcSpace
case 'rgb'
return;
case 'hsv'
% Convert HSV to R'G'B'
Image = huetorgb((1 - Image(:, :, 2)).*Image(:, :, 3),Image(:, :, 3),Image(:, :, 1));
case 'hsl'
% Convert HSL to R'G'B'
L = Image(:, :, 3);
Delta = Image(:, :, 2).*min(L,1-L);
Image = huetorgb(L-Delta,L+Delta,Image(:, :, 1));
case {'xyz','lab','luv','lch'}
% Convert to CIE XYZ
Image = xyz(Image,SrcSpace);
% Convert XYZ to RGB
T = [3.240479,-1.53715,-0.498535;-0.969256,1.875992,0.041556;0.055648,-
0.204043,1.057311];
R = T(1)*Image(:, :, 1) + T(4)*Image(:, :, 2) + T(7)*Image(:, :, 3); % R
G = T(2)*Image(:, :, 1) + T(5)*Image(:, :, 2) + T(8)*Image(:, :, 3); % G
B = T(3)*Image(:, :, 1) + T(6)*Image(:, :, 2) + T(9)*Image(:, :, 3); % B
% Desaturate and rescale to constrain resulting RGB values to [0,1]
AddWhite = -min(min(min(R,G),B),0);
Scale = max(max(max(R,G),B)+AddWhite,1);
R = (R + AddWhite)./Scale;
G = (G + AddWhite)./Scale;
B = (B + AddWhite)./Scale;
% Apply gamma correction to convert RGB to Rec. 709 R'G'B'
Image(:, :, 1) = gammacorrection(R); % R'
Image(:, :, 2) = gammacorrection(G); % G'
Image(:, :, 3) = gammacorrection(B); % B'

```

```

otherwise % Conversion is through an affine transform
    T = gettransform(SrcSpace);
    temp = inv(T(:,1:3));
    T = [temp,-temp*T(:,4)];
    R = T(1)*Image(:,:,1) + T(4)*Image(:,:,2) + T(7)*Image(:,:,3) + T(10);
    G = T(2)*Image(:,:,1) + T(5)*Image(:,:,2) + T(8)*Image(:,:,3) + T(11);
    B = T(3)*Image(:,:,1) + T(6)*Image(:,:,2) + T(9)*Image(:,:,3) + T(12);
    AddWhite = -min(min(min(R,G),B),0);
    Scale = max(max(max(R,G),B)+AddWhite,1);
    R = (R + AddWhite)./Scale;
    G = (G + AddWhite)./Scale;
    B = (B + AddWhite)./Scale;
    Image(:,:,1) = R;
    Image(:,:,2) = G;
    Image(:,:,3) = B;
end

% Clip to [0,1]
Image = min(max(Image,0),1);
return;

function Image = xyz(Image,SrcSpace)
% Convert to CIE XYZ from 'SrcSpace'
WhitePoint = [0.950456,1,1.088754];

switch SrcSpace
case 'xyz'
    return;
case 'luv'
    % Convert CIE L*uv to XYZ
    WhitePointU = (4*WhitePoint(1))./(WhitePoint(1) + 15*WhitePoint(2) +
3*WhitePoint(3));
    WhitePointV = (9*WhitePoint(2))./(WhitePoint(1) + 15*WhitePoint(2) +
3*WhitePoint(3));
    L = Image(:,:,1);
    Y = (L + 16)/116;
    Y = invf(Y)*WhitePoint(2);
    U = Image(:,:,2)./(13*L + 1e-6*(L==0)) + WhitePointU;
    V = Image(:,:,3)./(13*L + 1e-6*(L==0)) + WhitePointV;
    Image(:,:,1) = -(9*Y.*U)./(U-4).*V - U.*V; % X
    Image(:,:,2) = Y; % Y
    Image(:,:,3) = (9*Y - (15*V.*Y) - (V.*Image(:,:,1)))./(3*V); % Z
case {'lab','lch'}
    Image = lab(Image,SrcSpace);
    % Convert CIE L*ab to XYZ
    fY = (Image(:,:,1) + 16)/116;
    fX = fY + Image(:,:,2)/500;

```



```

fZ = fY - Image(:,:,3)/200;
Image(:,:,1) = WhitePoint(1)*invf(fX); % X
Image(:,:,2) = WhitePoint(2)*invf(fY); % Y
Image(:,:,3) = WhitePoint(3)*invf(fZ); % Z
otherwise % Convert from some gamma-corrected space
% Convert to Rec. 701 R'G'B'
Image = rgb(Image,SrcSpace);
% Undo gamma correction
R = invgammacorrection(Image(:,:,1));
G = invgammacorrection(Image(:,:,2));
B = invgammacorrection(Image(:,:,3));

```

```

% Convert RGB to XYZ
T = inv([3.240479,-1.53715,-0.498535;-0.969256,1.875992,0.041556;0.055648,-
0.204043,1.057311]);
Image(:,:,1) = T(1)*R + T(4)*G + T(7)*B; % X
Image(:,:,2) = T(2)*R + T(5)*G + T(8)*B; % Y
Image(:,:,3) = T(3)*R + T(6)*G + T(9)*B; % Z
end
return;

```

```

function Image = hsv(Image,SrcSpace)
% Convert to HSV
Image = rgb(Image,SrcSpace);
V = max(Image,[],3);
S = (V - min(Image,[],3))./(V + (V == 0));
Image(:,:,1) = rgbtohue(Image);
Image(:,:,2) = S;
Image(:,:,3) = V;
return;

```

```

function Image = hsl(Image,SrcSpace)
% Convert to HSL
switch SrcSpace
case 'hsv'
% Convert HSV to HSL
MaxVal = Image(:,:,3);
MinVal = (1 - Image(:,:,2)).*MaxVal;
L = 0.5*(MaxVal + MinVal);
temp = min(L,1-L);
Image(:,:,2) = 0.5*(MaxVal - MinVal)./(temp + (temp == 0));
Image(:,:,3) = L;
otherwise
Image = rgb(Image,SrcSpace); % Convert to Rec. 701 R'G'B'
% Convert R'G'B' to HSL
MinVal = min(Image,[],3);

```

```

MaxVal = max(Image,[],3);
L = 0.5*(MaxVal + MinVal);
temp = min(L,1-L);
S = 0.5*(MaxVal - MinVal)./(temp + (temp == 0));
Image(:,:,1) = rgbtohue(Image);
Image(:,:,2) = S;
Image(:,:,3) = L;
end
return;

```

```

function Image = lab(Image,SrcSpace)
% Convert to CIE L*a*b* (CIELAB)
WhitePoint = [0.950456,1,1.088754];

switch SrcSpace
case 'lab'
    return;
case 'lch'
    % Convert CIE L*CH to CIE L*ab
    C = Image(:,:,2);
    Image(:,:,2) = cos(Image(:,:,3)*pi/180).*C; % a*
    Image(:,:,3) = sin(Image(:,:,3)*pi/180).*C; % b*
otherwise
    Image = xyz(Image,SrcSpace); % Convert to XYZ

```

```

    % Convert XYZ to CIE L*a*b*
    X = Image(:,:,1)/WhitePoint(1);
    Y = Image(:,:,2)/WhitePoint(2);
    Z = Image(:,:,3)/WhitePoint(3);
    fX = f(X);
    fY = f(Y);
    fZ = f(Z);
    Image(:,:,1) = 116*fY - 16; % L*
    Image(:,:,2) = 500*(fX - fY); % a*
    Image(:,:,3) = 200*(fY - fZ); % b*
end
return;

```

```

function Image = luv(Image,SrcSpace)
% Convert to CIE L*u*v* (CIELUV)
WhitePoint = [0.950456,1,1.088754];
WhitePointU = (4*WhitePoint(1))./(WhitePoint(1) + 15*WhitePoint(2) +
3*WhitePoint(3));
WhitePointV = (9*WhitePoint(2))./(WhitePoint(1) + 15*WhitePoint(2) +
3*WhitePoint(3));

```

```

Image = xyz(Image,SrcSpace); % Convert to XYZ
U = (4*Image(:,:,1))./(Image(:,:,1) + 15*Image(:,:,2) + 3*Image(:,:,3));
V = (9*Image(:,:,2))./(Image(:,:,1) + 15*Image(:,:,2) + 3*Image(:,:,3));
Y = Image(:,:,2)/WhitePoint(2);
L = 116*f(Y) - 16;
Image(:,:,1) = L; % L*
Image(:,:,2) = 13*L.*(U - WhitePointU); % u*
Image(:,:,3) = 13*L.*(V - WhitePointV); % v*
return;

```

```

function Image = lch(Image,SrcSpace)
% Convert to CIE L*ch
Image = lab(Image,SrcSpace); % Convert to CIE L*ab
H = atan2(Image(:,:,3),Image(:,:,2));
H = H*180/pi + 360*(H < 0);
Image(:,:,2) = sqrt(Image(:,:,2).^2 + Image(:,:,3).^2); % C
Image(:,:,3) = H; % H
return;

```

```

function Image = huetorgb(m0,m2,H)
% Convert HSV or HSL hue to RGB
N = size(H);
H = min(max(H(:),0),360)/60;
m0 = m0(:);
m2 = m2(:);
F = H - round(H/2)*2;
M = [m0, m0 + (m2-m0).*abs(F), m2];
Num = length(m0);
j = [2 1 0;1 2 0;0 2 1;0 1 2;1 0 2;2 0 1;2 1 0]*Num;
k = floor(H) + 1;
Image = reshape([M(j(k,1)+(1:Num).'),M(j(k,2)+(1:Num).'),M(j(k,3)+(1:Num).')],[N,3]);
return;

```

```

function H = rgbtohue(Image)
% Convert RGB to HSV or HSL hue
[M,i] = sort(Image,3);
i = i(:,:,3);
Delta = M(:,:,3) - M(:,:,1);
Delta = Delta + (Delta == 0);
R = Image(:,:,1);
G = Image(:,:,2);
B = Image(:,:,3);
H = zeros(size(R));
k = (i == 1);
H(k) = (G(k) - B(k))./Delta(k);

```

```

k = (i == 2);
H(k) = 2 + (B(k) - R(k))./Delta(k);
k = (i == 3);
H(k) = 4 + (R(k) - G(k))./Delta(k);
H = 60*H + 360*(H < 0);
H(Delta == 0) = nan;
return;

```

```

function Rp = gammacorrection(R)
Rp = real(1.099*R.^0.45 - 0.099);
i = (R < 0.018);
Rp(i) = 4.5138*R(i);
return;

```

```

function R = invgammacorrection(Rp)
R = real(((Rp + 0.099)/1.099).^(1/0.45));
i = (R < 0.018);
R(i) = Rp(i)/4.5138;
return;

```

```

function fY = f(Y)
fY = real(Y.^(1/3));
i = (Y < 0.008856);
fY(i) = Y(i)*(841/108) + (4/29);
return;

```

```

function Y = invf(fY)
Y = fY.^3;
i = (Y < 0.008856);
Y(i) = (fY(i) - 4/29)*(108/841);
return;

```

10.2.3 Function gaussgradient2.m

This function from our thesis is modified from Gaussgradient.m function contributed by Guanglei Xiong (Xiong, G., 2005).

```
function [gx,gy]=gaussgradient2(IM,sigma)
```

```

%GAUSSGRADIENT Gradient using first order derivative of Gaussian.
% [gx,gy]=gaussgradient(IM,sigma) outputs the gradient image gx and gy of
% image IM using a 2-D Gaussian kernel. Sigma is the standard deviation of
% this kernel along both directions.

```

```

%
% Contributed by Guanglei Xiong (xgl99@mails.tsinghua.edu.cn)
% at Tsinghua University, Beijing, China.

%determine the appropriate size of kernel. The smaller epsilon, the larger
%size.
epsilon=1e-2;
halfsize=ceil(sigma*sqrt(-2*log(sqrt(2*pi))*sigma*epsilon)));
size=2*halfsize+1;
%generate a 2-D Gaussian kernel along x direction
for i=1:size
    for j=1:size
        u=[i-halfsize-1 j-halfsize-1];
        %This equation is  $f(x) = 1/\sqrt{2\pi\sigma} - g(x)$  in Xu's paper
        hx(i,j)=1/sqrt(2*pi*sigma) - gauss(u(1),sigma);
    end
end
hx=hx/sqrt(sum(sum(abs(hx).*abs(hx))));
%generate a 2-D Gaussian kernel along y direction
hy=hx';
%2-D filtering
gx=imfilter(IM,hx,'replicate','conv');
gy=imfilter(IM,hy,'replicate','conv');

function y = gauss(x,sigma)
%Gaussian
% $y = \exp(-x^2/(2*\sigma^2)) / (\sigma*\sqrt{2*\pi})$ ;

```

**% This has been modified according to equation used in Xu's paper, sigma is inside
% square root. So it has been edited to:**

```
y = exp(-x^2/(2*sigma^2)) / (sqrt(2*pi*sigma));
```

```

function canny_detector(I,i1,i2,j1,j2,thresh_step,sigma_step)
% I = image, i1 = lower threshold, i2 = higher threshold
% 0<i1<i2<1 and sigma must be positive number
% j1 = lower sigma value, j2 = higher sigma value
% thresh_step: step for threshold, sigma_step: step for sigma
for i = i1:thresh_step:i2
    for j = j1:sigma_step:j2
        c = edge(I,'canny',i,j);
        method = 'canny_';
    end
end

```

```

        ext = '.jpg';
        imwrite (c,strcat(method,num2str(i),'_',num2str(j),ext));
    end
end

```

10.2.4 Function Canny_detector.m

```

function canny_detector(I,i1,i2,j1,j2,thresh_step,sigma_step)

```

```

% This function is used to test Canny edge detector with various of threshold and sigma
% value
% I = image, i1 = lower threshold, i2 = higher threshold
% 0<i1<i2<1 and sigma j must be positive number
% j1 = lower sigma value, j2 = higher sigma value
% thresh_step: step for threshold, sigma_step: step for sigma
for i = i1:thresh_step:i2
    for j = j1:sigma_step:j2
        c = edge(I,'canny',i,j);
        method = 'canny_';
        ext = '.jpg';
        imwrite (c,strcat(method,num2str(i),'_',num2str(j),ext));
    end
end
end

```

10.2.5 Function LoG_detector.m

```

function LoG_detector(I,i1,i2,j1,j2,thresh_step,sigma_step)

```

```

% This function is used to test LoG edge detector with various of threshold and sigma
% value
% I = image, i1 = lower threshold, i2 = higher threshold
% i1<i2 and sigma must be positive number
% j1 = lower sigma value, j2 = higher sigma value
% thresh_step: step for threshold, sigma_step: step for sigma
for i = i1:thresh_step:i2
    for j = j1:sigma_step:j2
        l = edge(I,'log',i,j);
        method = 'log_';
        ext = '.jpg';
        imwrite (l,strcat(method,num2str(i),'_',num2str(j),ext));
    end
end
end

```

10.2.6 Function Sobel_detector.m

```
function sobel_detector(I,i1,i2,thresh_step)
```

```
% This function is used to test Sobel edge detector with various of threshold  
% I = image, i1 = lower threshold, i2 = higher threshold  
% i1 < i2  
% thresh_step: step for threshold  
for i = i1:thresh_step:i2  
    s = edge(I,'sobel',i);  
    imwrite (s,strcat('sobel_',num2str(i),'.jpg'));  
end
```

10.2.7 Function Prewitt_detector.m

```
function prewitt_detector(I,i1,i2,thresh_step)
```

```
% This function is used to test Prewitt edge detector with various of threshold  
% I = image, i1 = lower threshold, i2 = higher threshold  
% 0 <= i1 < i2 < 1  
% thresh_step: step for threshold  
for i = i1:thresh_step:i2  
    p = edge(I,'prewitt',i);  
    imwrite (p,strcat('prewitt_',num2str(i),'.jpg'));  
end
```

10.2.8 Function Roberts_detector.m

```
function roberts_detector(I,i1,i2,thresh_step)
```

```
% This function is used to test Roberts Cross edge detector with various of threshold  
% I = image, i1 = lower threshold, i2 = higher threshold  
% 0 <= i1 < i2 < 1  
% thresh_step: step for threshold  
for i = i1:thresh_step:i2  
    r = edge(I,'robert',i);  
    imwrite (r,strcat('robert_',num2str(i),'.jpg'));  
end
```