

AUTO-LOGO-TAGGING SYSTEM FOR PHOTOGRAPHER

BY

LEONG KHEI HUA

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONS) INFORMATION SYSTEMS  
ENGINEERING

Faculty of Information and Communication Technology

(Perak Campus)

MAY 2015

## REPORT STATUS DECLARATION FORM

Title: \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

Academic Session: \_\_\_\_\_

I \_\_\_\_\_

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in  
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

\_\_\_\_\_  
(Author's signature)

\_\_\_\_\_  
(Supervisor's signature)

Address:

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_  
Supervisor's name

Date: \_\_\_\_\_

Date: \_\_\_\_\_

AUTO-LOGO-TAGGING SYSTEM FOR PHOTOGRAPHER

BY

LEONG KHEI HUA

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfilment of the requirements

for the degree of

BACHELOR OF INFORMATION SYSTEMS (HONS) INFORMATION SYSTEMS  
ENGINEERING

Faculty of Information and Communication Technology

(Perak Campus)

MAY 2015

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**Auto-Logo-Tagging System for Photographer**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : \_\_\_\_\_

Name : \_\_\_\_\_

Date : \_\_\_\_\_

## **ACKNOWLEDGEMENT**

First of all, I would like to take this opportunity to express my greatest gratitude to my supervisor for this project that is Dr. Kheng Cheng Wai who has been very patient and supportive throughout this project. I feel lucky to have met Dr. Kheng Cheng Wai as my supervisor to assist me in this project due to the challenging nature of this project. His guidance and encouragement have given me the motivation and confidence to solve many issues that I had faced in this project. He is willing to share his experience and knowledge with me, therefore I am able to expand my knowledge and broaden my experience in image processing which is the main domain in this project.

Other than that, I would also like to thank my friends and family who are always very supportive all the time. They always give many advises and recommendations to me throughout this project that can give me more innovative ideas so that I can improve this project and accomplish it successfully. Without their support, I might not be able to achieve this far today.

Lastly, I had learnt a lot regarding the image processing field by reading the related articles. I find that I have much enthusiasm in the image processing. I would like to participate in this project again if it is given a chance to me for re-picking the project topic.

## **ABSTRACT**

In this project, there is a system called Auto-Logo-Tagging System that will be developed in order to solve some problems faced by the users especially the photographers. It has significant effect for them as they need the logo to be tagged in photos or images so that they can protect their images from being exploited by other parties. This report also explains how the problem exists for the photographers and how the Auto-Logo-Tagging System solves the problem. There are information gain, colour space, feature point detection, and wavelet transform algorithms that were done by other researchers to be reviewed that can greatly help and be referred to implement the proposed system. In order to solve the problem stated, there will be few algorithms used in Auto-Logo-Tagging System. They are HSI colour space algorithm and Kullback-Leibler information gain algorithm. They can help the system to identify the main objects in an image and find the best position for the logo to be tagged in image. In other words, the background and foreground of the image can be identified through the algorithms used and the placement of logo can be easily done. Therefore the photographers do not have to tag or place the logo manually by using other image editing software.

## TABLE OF CONTENTS

TITLE .....	i
REPORT STATUS DECLARATION FORM .....	ii
TITLE .....	iii
DECLARATION OF ORIGINALITY .....	iv
ACKNOWLEDGEMENT .....	v
ABSTRACT.....	vi
TABLE OF CONTENTS.....	vii
LIST OF FIGURES .....	ix
LIST OF TABLES .....	xi
LIST OF ABBREVIATIONS.....	xii
Chapter 1 Introduction .....	1
1.1 Motivation and Problem Statement .....	1
1.2 Project Scope and Objectives.....	3
1.2.1 Project Title.....	3
1.2.2 Project Objective.....	3
1.2.3 Project Scope .....	3
1.2.4 Project Innovation / Contribution .....	4
Chapter 2 Literature Review .....	5
2.1 Literature Review on Information Gain.....	5
2.2 Literature Review on Skin Colour Tone Model .....	10
2.3 Literature Review on Feature Point Detection.....	17
2.4 Literature Review on Wavelet Transform .....	22
2.5 Summary of Literature Review on Information Gain.....	29
2.6 Summary of Literature Review on Skin Colour Tone Model.....	30
2.7 Summary of Literature Review on Feature Point Detection.....	31
2.8 Summary of Literature Review on Wavelet Transform .....	32

2.9 Review of Existing Software .....	33
Chapter 3 Methodology .....	42
3.1 Block Diagram .....	42
3.2 System Methodology .....	43
3.2.1 Calculate Mean and Standard Deviation in Moving Kernel .....	43
3.2.2 Compute Information Loss from Adjacent Kernels.....	45
3.2.3 Find Suitable Location for Logo Placement .....	47
3.3 Experiment Setting.....	49
3.3.1 Identification of Background and Foreground Using Kullback-Leibler Information Gain.....	49
3.3.2 Identification of Human Skin Using HSI Colour Space Model .....	52
3.4 Actual System and Algorithms Walkthroughs .....	55
Chapter 4 Conclusion.....	64
Chapter 5 Reference.....	66
Appendix A Gaussian Mixture Model and Expectation-Maximization Algorithm...A-1	
A-1 Gaussian Mixture Model .....	A-1
A-2 Expectation (E Step) .....	A-3
A-3 Maximization (M Step).....	A-4
A-4 Termination.....	A-4
A-5 Kullback-Leibler Divergence Algorithm.....	A-5
Appendix B Determination of Skin Pixel in the Image .....	B-1
B-1 RGB Colour Space.....	B-1
B-2 Cb-Cr Subspace from YCbCr Colour Space .....	B-2
B-3 HSV Colour Space.....	B-3



## LIST OF FIGURES

Figure 1-1: Illustration for the logo placed at the frame of photo. ....	2
Figure 2-1: The structure of transform or image coder (Gracemann & Miikkulainen 2005). ....	23
Figure 2-2: The ESP algorithm which is applied to the wavelets (Gracemann & Miikkulainen 2005). ....	24
Figure 2-3: The evaluation function in the ESP algorithm (Gracemann & Miikkulainen 2005). ....	24
Figure 2-4: Basic model of compression system (Ruchika, Singh & Singh 2012). ....	26
Figure 2-5: Thee compression algorithm proposed in the study (Ruchika, Singh & Singh 2012). ....	27
Figure 2-6: Illustration for Step 1 (Smith 2012). ....	33
Figure 2-7: Illustration for Step 2 (Smith 2012). ....	34
Figure 2-8: Illustration for Step 3 (Smith 2012). ....	34
Figure 2-9: Illustration for Step 4 (Smith 2012). ....	35
Figure 2-10: Illustration for Step 1 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	36
Figure 2-11: Illustration for Step 2 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	36
Figure 2-12: Illustration for Step 3 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	37
Figure 2-13: Illustration for Step 4 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	37
Figure 2-14: Illustration for Step 5 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	38
Figure 2-15: Illustration for Step 6 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	38
Figure 2-16: Illustration for Step 7 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	39
Figure 2-17: Illustration for Step 8 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	39
Figure 2-18: Illustration for Step 9 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013). ....	40

Figure 2-19: One of the ways to add the frame for an image (Ray 2012). .....	41
Figure 3-1: Block Diagram for illustrating overall process for Auto-Logo-Tagging System.....	42
Figure 3-2: The input-process-output flow chart for Calculate Mean and Standard Deviation in Moving Kernel Module.....	43
Figure 3-3: The input-process-output flow chart for Compute Information Loss from Adjacent Kernels Module. ....	45
Figure 3-4: The input-process-output flow chart for Find Suitable Location for Logo Placement Module. ....	47
Figure 3-5: The traversing of kernel in the image (Columbia University, n.d.). .....	49
Figure 3-6: The code for creating custom kernel based on size of the logo. ....	50
Figure 3-7: Expected result in the experiment (Columbia University, n.d.).....	51
Figure 3-8: The HSI colour space model (Maraqa, Al-Zboun, Dhyabat & Abu Zitar 2012). ....	52
Figure 3-9: The percentage of pixels in kernel that is occupied by skin pixels (Columbia University, n.d.). ....	53
Figure 3-10: Expected result that shows the possible location for tagging the logo in image (Columbia University, n.d.). ....	54
Figure 3-11: Logo is resized based on original photo.....	55
Figure 3-12: Kernel movement in photo image.....	56
Figure 3-13: Kernel will be moved to lower position after reaching right end of photo image.....	57
Figure 3-14: System will tag kernel position that has the lowest information loss from current kernel position. ....	58
Figure 3-15: Kernel position which has been tagged as having the least information loss by surrounding kernel positions.....	59
Figure 3-16: Potential positions for logo placement are identified by the system.....	61
Figure 3-17: The final result when the logo is placed at suitable location. ....	62
Figure 3-18: Another logo placement in other photo image by the system.....	63
Figure 3-19: Another logo placement in other photo image by the system.....	63

## LIST OF TABLES

Table 2-1: The summary of literature review on information gain. ....	29
Table 2-2: The summary of literature review on skin colour tone model. ....	30
Table 2-3: The summary of literature review on feature point detection. ....	31
Table 2-4: The summary of literature review on wavelet transform. ....	32
Table 2-5: The strengths and limitation for using Photoshop CS6 to tag the logo in images. ....	41

## LIST OF ABBREVIATIONS

<i>RGB</i>	Red, Blue and Green colour base model
<i>YCbCr</i>	Family of colour space model
<i>HSI</i>	Hue, Saturation and Intensity colour base model
<i>MIAS</i>	Mammographic Image Analysis Society
<i>RMSE</i>	Root Mean Square Error
<i>PSNR</i>	Peak-Signal-To-Noise Ratio
<i>MEC</i>	Minimum Compound Gain
<i>MEG</i>	Minimum Error Information Gain
<i>NVESD</i>	Night Vision and Electro-optic Sensors Directorate
<i>DISSTAF</i>	Distributed Interactive Simulation, Search And Target Acquisition Fidelity
<i>IITK</i>	Indian Institute of Technology, Kanpur
<i>Photoshop CS6</i>	Photoshop Creative Suite 6
<i>PNG</i>	Portable Network Graphics

## **Chapter 1 Introduction**

### **1.1 Motivation and Problem Statement**

In this project, the Auto-Logo-Tagging System for photographer will be going to be developed. Nowadays the number of people that are involved in photographic field keeps increasing from time to time. In order to protect their captured photos from being copied and transferred by other parties without the permission of original owners, the photographers need to put their own logo on the photos to show their ownership for those photos. Another main intention of putting a logo into the image is to advertise them to all audiences who view the photographers' captured images. There are some image editing software available for them to tag the logo on photos, this can solve the problem of violating the copyright for photos. Through some observations, there is a problem exists when the photographers use those image editing software, the problem is they need to find the best position for the logo to be placed in photos that cannot be done in those software. The best position for the logo means that the logo placed will not cover up the main objects in photos. This can cause the photographers need to add the logo manually after capturing the photos. It may consume lots of time for tagging the logo manually.

Instead of adding the logo manually, some of the photographers will let the software to place the logo automatically by fixing the position of logo placed in each photo, so that it can save the time for handling the process. The logo position can be fixed either at the edge of or in the photo. Although this can solve the problem of copyright violating, there are other problems exist. One of them is when the logo is placed at the edge of photo, this can be known as photo frame. It means that the logo will be placed at the frame of photos. By using photo frame the people can easily crop those photo by just removing logo which is tagged at the frame of photo. Then they can abuse them without the acknowledgement of photo's owner. There is another case where the photographers place the logo that is fixed in a particular location in photo. This can be done by using the image editing software that can tag the logo with many photos at the same time without doing it manually. This can lead to a problem in which the logo in some photos cannot be seemed obviously or clearly, or the logo is really fuzzy in the photos for some people.

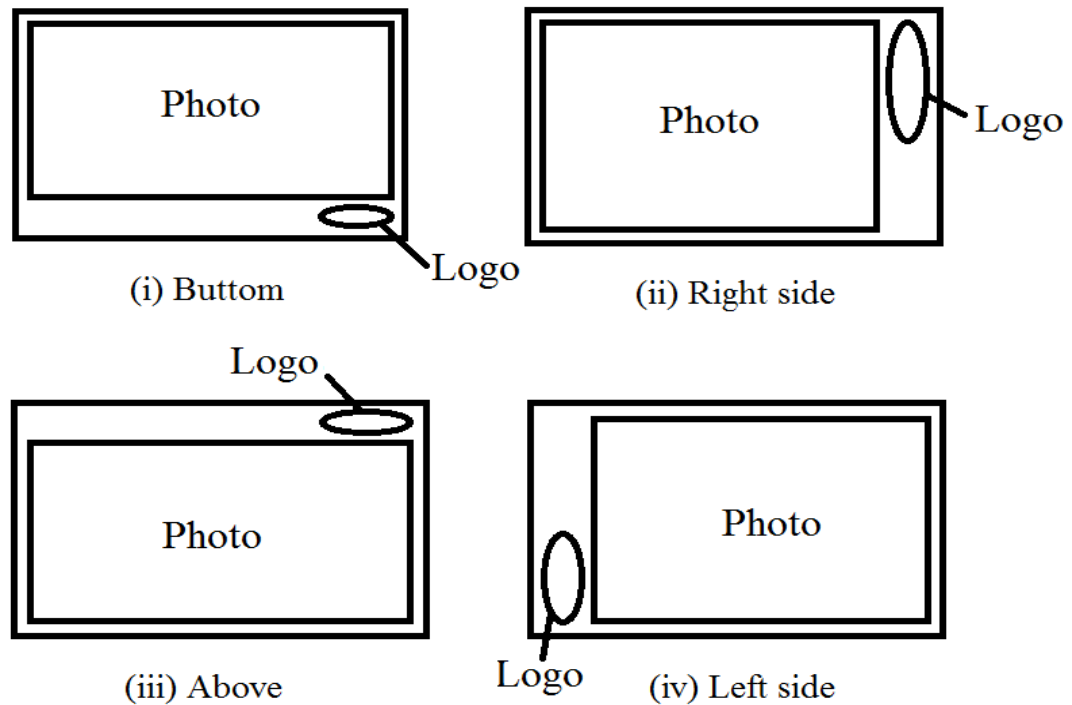


Figure 1-1: Illustration for the logo placed at the frame of photo.

The main problem is the existing software are unable to automatically add or tag the logo that is best fit in photos for the photographers so that the logo can be shown clearly to everyone. Therefore the main domain for this project is the image processing. This means that the knowledge related to image processing will be used to develop the Auto-Logo-Tagging System for photographers. Other than photographers, artists are also having the same problem mentioned before as they need to add their own logo on their drawings so that no one can counterfeit the ownership for those drawings. The Auto-Logo-Tagging System is important for them as it can help to automatically find the best position for logo to be added or tagged in the photos. In conclusion the problem mentioned is significant for them, the logo added or tagged in photos can help the photographers and artists to protect their copyright from being counterfeited by any party.

## **1.2 Project Scope and Objectives**

### **1.2.1 Project Title**

Auto-Logo-Tagging System for Photographers

### **1.2.2 Project Objective**

There are many objectives that need to be achieved in order to solve the problem stated in 1.1 Motivation and Problem Statement. The objectives are:

- To analyse how the Auto-Logo-Tagging System places the logo that is best fit in the photos so that the logo can be clearly shown with appropriate size and colour in photos.
- To design a system that can help the user to automatically find the best position for the logo to be placed in photos.
- To evaluate usefulness of the Auto-Logo-Tagging System for the purpose of assisting photographers.

### **1.2.3 Project Scope**

The project scope can help to define what the components will be included and what the things are needed to be done in the project. The project scope identified are:

- There are about 3 aspects that must be included in order to solve the problem of adding logo in photos. There are finding the best position for logo to be placed in photo, adjust the size of logo based on the best location found by system and change the colour of logo depends on the colour value computed in best position found.
- Some additional features will also be implemented to enhance the user experience for using the Auto-Logo-Tagging System. One of them is assist the users to manage their photos uploaded to the cloud storage available so that they can obtain those photos in anytime and anywhere. Simple logo editing tool is also enabled in Auto-Logo-Tagging system for the users who want to edit their own logo to be tagged in photos.

### **1.2.4 Project Innovation / Contribution**

In this project many innovation or contribution will be given that can help the users to eliminate the problem. The innovation in this project are:

- To employ the information gain approach to compare and find the best position for the logo to be tagged in photos. This can solve the problem and fulfil the objective No. 1 and No. 2.
- To employ the skin colour tone model to identify the main objects especially human beings in the photo so that the system can determine which areas that the logo can be placed. This can solve the problem and fulfil the objective No. 1, No. 2 and No. 3.



## Chapter 2 Literature Review

### 2.1 Literature Review on Information Gain

In order to find best position for placing the logo in some particular photos, there are many algorithms and techniques that need to be used so that the logo tagged will not cover up the main objects in photos. The information gain, skin colour tone model and feature point detection will be used to identify the main objects in photos or images and then the system will not place the logo in those objects identified. There are many experiments and studies (Garcia, Fdez-Valdivia, Rodriguez-S´anchez & Fdez-Vidal 2002; Garcia et al. 2001) that had been conducted in order to observe how information gain can be used in analysing the images which are contain more noise. Since these studies and experiments are mainly focusing on image processing, so the concept and knowledge of information gain can be greatly used in identifying the objects in photos or images.

According to the study conducted by Garcia, Fdez-Valdivia, Rodriguez-S´anchez and Fdez-Vidal (2002), there is an approach for using the information gain that is Kullback-Leibler information gain. It will be used for collecting and grouping the information from the compressed image that is relative to original image, specifically for the diagnose-purpose images in medical field. In this study the Kullback-Leibler information gain will also be tested for its performance in estimating the image fidelity. Image fidelity is the measure of accuracy for a particular image after it is reconstructed from the original one. This information gain approach is one of the useful algorithms used in image processing, this study let the researchers to take the opportunity to prove the usefulness of Kullback-Leibler information gain. This can greatly contribute for the image processing field. Therefore in order to test its performance, there were about three postulates to be proposed before the testing was started. Those postulates were (Garcia, Fdez-Valdivia, Rodriguez-S´anchez & Fdez-Vidal 2002):

1. The property of how a single event of a digital image was that is depending on its probability occurrence.
2. The extent which some probability distributions with the mean of mathematical expectation of how unexpected its single events were from this distribution can affect how unexpected a digital image was.

3. The estimation of how unexpected the reference image was from an estimated distribution and the estimate from true distribution.

So there were ten digitized mammographic images in JPEG form will be obtained from Mammographic Image Analysis Society (MIAS) and the Department of radiology at the University of Nijmegen. They will be used in this testing which also mainly analyses the consequences of lousy compression. Those images were originally used for identification of breast cancer micro-calcification. By using them through testing they were evaluated whether those images can be used for the specific problem of detection in individual micro-calcifications at the different compression ratios of images (Garcia, Fdez-Valdivia, Rodriguez-S´anchez & Fdez-Vidal 2002). There was one term called false positive (FP) to true positive (TP) that will be used as a method to group the compression losses through images reconstructing under many extent of lousy compression. Its value can be obtained by reviewing those reconstructed images and used to be evaluated in terms of diagnose usefulness.

Through the terms of diagnose usefulness the Kullback-Leibler information gain will use the false positive to true positive value in order to predict the image fidelity and measure the amount of image distortion between the original image and reconstructed image after compression. And its performance will be assessed. To compare its performance there were additional two quantitative measures will be included in the evaluation, they are Root Mean Square Error (RMSE) and Peak-Signal-To-Noise Ratio (PSNR). After the testing the result showed that Kullback-Leibler information gain had the best performance among the three qualitative measures.

This study shows that Kullback-Leibler information gain is one of the best image processing algorithm used. It can also be used for finding the main objects in a particular image. In order to measure the accuracy of the results, this study used bootstrap sampling for statistical reasoning (Garcia, Fdez-Valdivia, Rodriguez-S´anchez & Fdez-Vidal 2002). The sampling result showed that the testing has 95% of confidence interval for the difference of means. This means that the result for testing was not having big difference when the size of testing is increasing. Therefore the result of this testing in the study is reliable and can be used to show the usefulness of Kullback-Leibler information gain. The only weakness of this study is the testing of Kullback-Leibler information gain performance is only for predicting the images fidelity in medical field.

## Chapter 2 Literature Review

If there are more resources available for the researchers, they can conduct the studies other than predicting the image fidelity, such as testing the information gain accuracy and determining the objects in images with greater noise surrounding them. From this study the Kullback-Leibler information gain algorithm used can be greatly applied in the Auto-Logo-Tagging System because this algorithm detects the objects in images more effectively with lower error of probability.

There is another study that is related to information gain (Garcia et al. 2001), but the intent of study is different with the previous one, it is using minimum error gain to predict the visual target distinctness. This study is quite similar to the previous one as the same authors did also researches or experiments for the past study. Therefore both of them have some similarities in terms of methods used and the area of study. This study was more focusing on how the information gain can determine the distinctness of visual target between target and non-target images. The target image here means an image that has visual target, or it has the main object that can be identified as a target. The goal of this study is to show that visual target can be determined with minimum error probability by using minimum error information gain. Compared with the previous study, this study used more images data set and more algorithms in order to archive its objectives and satisfy the hypotheses proposed. Due to this reason the study became much more complex and may involve lots of experiments.

There are two hypotheses proposed in this study that are first, through the representation of the image content based on orientation, spatial frequency, scale and spatial location it can determine exactly the structure of target and non-target images. And second, since it has some possibilities that the target and non-target image structures cannot be determined precisely, so they need to be characterized statistically by using discrete probability distributions (Garcia et al. 2001). In order to support a large number of measures for its availability there are seven postulates to be presented (Garcia et al. 2001), they are:

1. The measure of how unexpected a single event of target image was depends only on its probability.
2. The mathematical expectation of how unexpected the target image is from some probability distribution will define the estimation of how unexpected its single events were from this distribution.

3. From an estimated distribution the target image is more unexpected than from the true distribution.
4. In the comparison between the target and non-target image, its main concern is how to reduce the data without major loss of information if the target image contains relevant structures, unwanted details and noise.
5. If the target image's information at significant points is some constraint for non-target image, there are three steps included for determining error between them. They are firstly finding interest points in target image, secondly modifying scale parameter of interest point and lastly compressing non-target image locally with target image.
6. The selective information gain will be obtained from a discrepancy function between the interest points signification in target and non-target image when the target image has its interest points and their significance are some given constraint in non-target image.
7. The interest point mentioned is a spatial place or location of partially non-changing behaviour that can be used for minimizing the error probability between the target and non-target images or scenes in order to test whether the target is present.

In this study, the testing for predicting the visual target distinctness was using many target and non-target images. Those images was obtained from distributed interactive simulation, search and target acquisition fidelity (DISSTAF) field test, which was organized by Night Vision and Electro-optic Sensors Directorate (NVESD). The images obtained were captured in the complex environment and had much more noise surrounding the target or object. They were also divided into two sets of data and tested by using many information gain algorithms or prediction models. The algorithms used in this test were Root Mean Square Error (RMSE), selective gain, compound gain, Minimum Error Information Gain (MEG) and Minimum Compound Gain (MEC). This test included three experiment, the first one was using those prediction models to find the distinctness of the target and non-target images in the first set of data images. Another experiment was also the same with the first one, but it was using different set of data images. The third experiment was using both set of data in order to find what the impact is when altering the both original data sets by using same model predictions. The final result of those three experiments were the MEC prediction model showed the

## Chapter 2 Literature Review

best overall performance for predicting the visual target distinctness with lower error probability when comparing with other prediction models. This result showed that it can satisfy the seven postulate proposed before, therefore it is valid that by using information gain the distinctness for visual target can be predicted or determined.

From this study it clearly demonstrated how the information gain works towards the images which are much more complex and have more noise that could make the object determination harder for the system. Through many algorithms used for the information gain, this study used many complicated images to show how the information gain can determine the image distinctness that might be very difficult to do by observing with naked eyes. With those experiment results they showed that the distinctness among different images were accurately determined. This can be useful in finding the main objects in an image for placing the logo in it by using information gain approach, even though it does not comparing many images at the same time. The only problem in this study is the images for the purpose of testing might not be sufficient to show how accurate the results are in each experiment conducted. It is suggested that the images used can be enhanced in terms of its quantity and variety. The variety means that the complexity from many images can be increased so that the experiment results will be more reliable and accurate with various information gain algorithms.

## 2.2 Literature Review on Skin Colour Tone Model

Besides the information gain, the studies about the skin colour tone models (Singh, Chauhan, Vatsa & Singh 2003; Terrillon, David & Akamatsu 1998; Zangana & Al-Shaikhli 2013; Lakshmi & PatilKulakarni 2010) are also reviewed as it is useful for detecting the main objects in images especially for the humans. Skin colour tone model is based on the colour space model. By using skin colour tone model, the system can easily detect humans who are the main objects in images by comparing the colour with algorithms provided. In the following studies there are many approaches or algorithms proposed for detecting the human skin in image through various types of colour space models used.

There is one study conducted by Singh, Chauhan, Vatsa and Singh (2003), proposed a new approach for detecting human face and skin in images. It was proposed based on three colour space models that are Red, Green and Blue (RGB) colour space, YCbCr colour space and Hue, Saturation and Intensity (HSI) colour space. In this study the authors explained how those three colour space models use their own skin colour classification for localising and detecting the human face and skin in images. Their algorithms were also explained and analysed so that their strengths and weaknesses can be identified. Based on the analysis done, the study proposed a new algorithm that will combine skin colour classifications of those three colour space models.

In the new algorithm proposed, the results generated from those three colour space models will be combined for finding the human skin and face. Later it detects the skin regions in image based on the combined results and then the face in that region will be extracted. There was one assumption made for the algorithm proposed, that was if one or more colour space algorithms detect the skin regions in an image while other than them give the false result for the same image used, then the algorithm proposed will still extract the face from skin regions through combined colour space algorithms (Singh, Chauhan, Vatsa & Singh 2003). After performing skin colour statistics on the image retrieved, the image will be transformed to become the binary image which is aimed to remove saturation and hue as well as the noise of image. At this point the algorithm now can extract the eyes, ears and mouth from that processed binary image, and the components extracted will be formed a triangle shape. The triangle shape formed can help the algorithm proposed to identify potential facial region in the image

by estimating the coordinates of four corner points that can also form the shape of rectangle. This is how the algorithm proposed can estimate and identify the location of human face and skin in an image. The new algorithm proposed for detecting face in the images provides an effective and yet high accurate results that may minimize the weaknesses from those three colour space algorithms.

The detailed experimental comparison for the new algorithm proposed and three colour space algorithm was performed on the images obtained from the internet and Indian Institute of Technology, Kanpur (IITK) (Singh, Chauhan, Vatsa & Singh 2003). The false detection rate and false dismissal rate were used as a comparative measure for each algorithm in experiment. The experiment result showed that RGB colour space is not performing well in face detection based on skin colour classification, means that it cannot detect the face and skin in images accurately. It has highest false detection rate and false dismissal rate among the four algorithm used. While the YCbCr colour space has better result than RGB colour space. Its false detection rate and false dismissal rate is lower than RGB colour space. The result for HSI colour space is similar to YCbCr colour space, but its false detection rate is slightly higher than YCbCr's. Finally the new algorithm proposed has the best result and performance among four algorithms in the experiment. It has very low false detection rate and false dismissal rate. This showed that the new algorithm proposed is far more effective and accurate for detecting the face in an image than using one of the three colour space algorithms alone. Therefore this study proved that by combining many colour spaces together it can perform the detection of human skin and face more effective in images.

This study can be a strong proof for showing that instead of using one colour space for human skin and face detection in the images, combining many colour spaces together is one of the good approaches used in order to perform human skin and face detection more effectively. It also helps to reduce the probability of false detection and dismissal happened which can be redundant and ineffective. Based on the experiment result although the performance of new algorithm proposed is better than others, the duration for detecting face and skin in images is much longer than others, its duration is twice as others. This is because it needs more time to combine all results from three colour space algorithms and then identify the skin and face in images based on those results. Therefore its performance in terms of time may not be effective although it has better

detection accuracy when comparing with others. The study can be further extended for enhancing and shortening the algorithm performing duration.

There is another study (Zangana & Al-Shaikhli 2013) that is quite similar to the previous one. This study also proposed a new approach of skin colour space algorithm in order to detect or identify the human face and skin in the images. The study also stated that there are some problems when using colour as a tool to identify the human face in the images (Zangana & Al-Shaikhli 2013). One of them is the colour denotation for the images captured, it is affected by many factors like the ambient light and object movement in the images. Another problem is some different cameras can produce major different colour values even though the cameras capture the images for the same person under the same lighting conditions. And also the skin colour variety among the people is also one of the problems stated. There are many colour spaces that can be used to solve these problems. Therefore in this study there was a comparative experiment for three skin colour face detection algorithms that were RGB colour space, YCbCr colour space and HSI colour space, and also a new proposed algorithm that was based on the YCbCr colour space model for its skin colour classification (Zangana & Al-Shaikhli 2013).

In this new proposed algorithm, the image that is decomposed into RGB colour space will be converted to YCbCr colour space. Before the colour space conversion the image will be resized into the size of  $256 * 256$  pixels. Then in order to get the new chrominance components from the image, the current colour space YCbCr was transformed into  $YC'bC'r$  colour space. This means that chrominance in the image will be separated with the luminance as well as compactness of the skin cluster in image. This can be done properly since the chrominance components in skin colour tone are independent from luminance components, even though in practical the skin colour tone is actually indirectly depending on luminance (Zangana & Al-Shaikhli 2013). After  $YC'bC'r$  colour space image is obtained, it will detect skin colour tone for image and then determine the face area in the image based on the  $C'r$  component in algorithm. By using the MATLAB system the detailed comparative experiment for testing the algorithm accuracy was conducted with three colour space algorithms and a new proposed algorithm. Many sample images obtained from internet will be used in the experiment. As similar to the previous study the rate of false detection and false dismissal were also be used as a measure for the experiment result. The process of



conducting the experiment was similar to the previous study, but their new proposed algorithms may be different with each other.

After conducting the experiment for testing the accuracy of those four algorithms, the result was generated and shown. The result showed that RGB colour space had the lowest accuracy rate, this means that it is not capable for detecting face in image with skin colour classification. In the experiment it also retrieved non-skin region as the parts of skin region, hence the accuracy is lower based on the experiment result. While the colour space YCbCr had much better result when comparing with RGB colour space. It indicated that this colour space is able to extract the skin region from an image more effective than RGB colour space. At the same time HSI also showed a good accuracy rate which was nearly equivalent to YCbCr colour space. Both YCbCr and RGB colour space had lower false dismissal rate but had high false detection rate. Other than those three colour space, the new proposed algorithm showed the best result among all the algorithms involved in experiment. It has very low false dismissal and false detection rate, hence it can effectively estimating the face region in the images.

When comparing with the previous study, the new algorithm proposed in this study is different with the previous one as it is just based on one colour space that is YCbCr while the new algorithm proposed from previous study is based on three colour space models. This can be a good example to show that the face detection in images can be improved further with the enhanced algorithm based on the existing colour space. However the new algorithm in this study has some weaknesses or limitations, one of them is it may not be able to detect multiple human faces in an image. So it might not be practical for face detection when the image have many human faces. This study needs to conduct some further experiment for testing whether the new proposed algorithm can identify many faces in an image. Another problem existed is this study did not show how much the time for the new proposed algorithm to perform the face detection in images. This may provide the information less clearly to the people about how well the new proposed algorithm is when comparing with others.

The study conducted by Lakshmi and PatilKulakarni (2010) had proposed different approach of algorithm for detecting faces in the images compared with the previous two studies that were also related to skin colour tone in image processing. In this study it is mainly focusing on how the segmentation algorithm can detect or identify multiple

faces in a colour image using many approaches together that are colour spaces and edge detection techniques. The new proposed algorithm will be based on colour space and the edge detection which will be explained later.

There is a new term that was used in this study. That new term is the edge detection. Edge detection is a process of determining and locating the sharp discontinuities (known as edges) in an image. The shape discontinuities here means the unexpected changes in pixel luminance that can form the boundaries for the objects in an image (Lakshmi & PatilKulakarni 2010). The study stated that it can be an important tool for image segmentation in image processing. There were three edge detections will be used in this study. One of them is Robert's Cross Edge Detection. It is an operator that can execute a simple, fast to compute, and two dimension gradient measurement on an image. The operator contains a pair of 2 x 2 leeway kernels or matrixes, one kernel is the other one rotated by 90 degree. The kernels also contain the pixel values that represent the estimated absolute magnitude for the spatial gradient in an image (Lakshmi & PatilKulakarni 2010). Canny Edge Detection was also included in this study. It is a method used to find the edges in which local maxima value for the gradient of image will be obtained. It mainly detects the weak edges in an image instead of finding the noise surrounding. Another term that is related to edge detection mentioned in this study is Prewitt Edge Detection. Its main function is to identify the horizontal and vertical edges available in an image. It will return the edges value at those points in which the image gradient is maximum (Lakshmi & PatilKulakarni 2010). This can be explained that this study will combine Canny Edge Detection and Prewitt Edge in order to remove unnecessary weak edges obtained and remain the strong boundary edges.

The new proposed algorithm was based on HSI and YCbCr colour space models, and using Canny and Prewitt Edge Detection in order to segment the skin regions or identify many human faces in an image. There were many steps for executing this algorithms, they are (Lakshmi & PatilKulakarni 2010):

1. Based on HSI colour space the input image is skin segmented. Then various structural operations are carried out on that skin segmented regions.
2. YCbCr colour space will be used for skin segmenting on the input image. Then various structural operations are also carried out on that skin segmented regions.

3. Single segmented image is formed by combining the skin segmented regions retrieved from Step 1 and Step 2. In order to obtain Combined-Segmented image, the algorithm will carry out connected component analysis on the single segmented image.
4. The input image is converted into grey scale image. By using Canny and Prewitt edge detection algorithms the edge-image from input image will be retrieved separately. Then both of edge-image values are combined and complemented to estimate the region boundaries in input image.
5. Region boundaries retrieved from Step 4 is multiplied with the Combined-Segmented image from Step 3. The final image will be generated after multiplication.

The segmented regions in final segmented image may contain holes which are because of the existence of eyes and mouth in the face. In the algorithm it will be assumed to be segmented face regions and other regions are eliminated. In the experimental test there were two algorithms used for testing the face detection accuracy. One of them is the new proposed algorithm and another one is Robert Cross Edge Detection algorithm combined with YCbCr colour space. They were also tested for generating image boundaries with same structuring element. There were also many images captured with many persons and they will be used as the samples for the testing. The result of testing showed that the new proposed algorithm can accurately and effectively segment many faces in an image. While the combination of Robert Cross Edge Detection algorithm and YCbCr colour space does not generate good result for segmenting the image, especially when the persons in image wearing the spectacles. That combined algorithm will segment the faces of persons wearing spectacles into different face regions instead of only one face region for each person. This showed that the new proposed algorithm can detect human skins and faces in images with minimum errors while the other one may not identify the skins and faces accurately even though it can segment the face region in some images.

Although this study is different with the previous two, it has still proven that it is encouraged for the system to use many algorithms and colour spaces together in order to detect human skins and faces in the images more effectively and accurately. The new proposed algorithm had been showing that multiple faces can also be detected by using appropriate colour spaces and algorithms. The previous studies are not focusing about

## Chapter 2 Literature Review

this aspect, instead they were just testing for detecting a single face in the image. There is a limitation for the algorithm proposed by this study. It is the algorithm may detect the objects mistakenly that have similar colours with the human skins. So when there are less objects which have similar colours with human skins in the image, the proposed algorithm can accurately detect the human skins and faces. In the other way it may also mistakenly detect those objects as the human skins and faces when they have occupied some certain significant amount of spaces in an image.

### **2.3 Literature Review on Feature Point Detection**

There are many studies are also conducted that relates to the feature point detection (Nain, Laxmi & Bhadviya 2008; Kautsky, Zitova, Flusser & Peters 1998). Feature point detection is a method that will help to make local decisions at each image point whether it is a feature for the image of a given type at that point, this can be done through abstractions of image information by computer. The features in detection result are the subsets of the image domain, and they are usually in the form of isolated points, continuous curves or connected regions (Feature Detection (computer vision) 2014). In the Auto-Logo-Tagging System, feature point detection can be used to determine the significant areas so that the logo can or cannot be tagged in these areas. In this section there are few articles will be reviewed in order to understand and know how the feature point detection works in the image processing.

The study conducted by Nain, Laxmi and Bhadviya (2008), stated the significant of feature point detection in the image processing. In this study, the interest point or feature point is defined that it is the image location where the signal changes two-dimensionally. Corners, T-junctions and locations in an image where the texture changes significantly are the examples of interest point. This study also indicated that corner is the intuitive type of feature point. This is because it is constant for the rotation and small changes can be inspected under different lighting. Another reason is corner can also minimize the number of data that is needed to be processed without losing the most significant information in the grey level image. That is why corner detection is mainly used as the first step in many vision tasks such as tracking, simultaneous localization and mapping, and recognition (Nain, Laxmi & Bhadviya 2008).

In this study, the feature point detector was proposed which is based on the situation that the feature points are nothing else but there are some unexpected changes in two dimensions in them. The main intention of the proposed algorithm is to make it possible for detection of the corners and all other points of interest in an image, as their information content is larger than the desired threshold (Nain, Laxmi & Bhadviya 2008). In order to achieve the main aim of proposed feature point detector, there are five short and effective steps of algorithm to be used for implementing the proposed feature point detector. Those five steps are (Nain, Laxmi & Bhadviya 2008):

1. Apply the Difference Mask with threshold parameter P1.

A very simple 2 x 2 mask will be used in this step instead of using the usual convolution masks differing from 3 x 3 to 7 x 7. It will replace the convolution with simple differences between intensity values. Then these differences will be compared with the threshold P1. The absolute sum of its values will represent the cornerity strength for that pixel.

2. Apply the partial averaged Gaussian Mask to the points satisfying Step 1.

The application of a pseudo-gaussian kernel which is partial averaged Gaussian Mask is used and applied with Difference Mask in this step. This can overcome the problem of responding to every noisy pixel by Difference Mask which is very redundant in feature point detector. This is caused by the image that consists of many types of noises and the blurred feature points. The use of pseudo-gaussian kernel can also increase the noise resistance of the algorithm proposed.

3. Calculate the Difference Mask with threshold parameter P2.

The Difference Mask is applied again with different threshold P2 when the new gaussian averaged values of four pixels under consideration is calculated.

4. Eliminate false positives

The candidate pixels which are only two-pixel connected in the diagonal direction and the part of diagonals are eliminated. This can avoid the situation in which all diagonal edges are responded strongly by the orthogonal Difference Masks. Once the candidate pixel is eliminated, its cornerity strength is reduced by half. It is a very important process as it can determine the locality of the still existing candidate points.

5. Determine localization of the desired feature points.

Cornerity strength for each of the four pixels in 2 x 2 pixel patch will be compared with each other. The pixel that has highest cornerity value will be chosen as the true position of feature point.

In order to test the proposed algorithm in terms of consistency and accuracy there were many transformations will be applied to the experimental cases. The experiment will use an image under many transformations for testing the proposed algorithm. The transformations used in the experiment are rotation variations, scaling variations, blurring, 3-D projection and artificial noise. The proposed algorithm will be compared

with other existing popular feature point detectors. The experiment result showed that the proposed algorithm has very good accuracy and consistency compared with other feature point detectors. This indicated that the simple yet effective algorithm is successfully evaluated for its performance in many variables like noise, accuracy and deformation speed.

Feature point can play the important role in the image processing as it helps to determine which points in an image are significant. By proposing an effective feature point detector it can minimize the errors occurred in the process and can help to reduce the resources needed for running the detector. The proposed algorithm by Nain, Laxmi and Bhadviya (2008) provides a simpler way to find the feature points in an image since the algorithm does not use much operations for finding the feature points. It can reduce the complexity for the system for running the feature point detector. There is one limitation identified for the proposed algorithm. The limitation is it might take longer time to perform feature point detection as it needs to compare and compute the threshold twice in the process, therefore it may cause larger delay for the system that implements the proposed algorithm.

There is another study done by Kautsky, Zitova, Flusser and Peters (1998) that is related to feature point detection. In this study it may differ from previous one as it mainly focused on detecting the feature points with multiframe instead of single frame of images. This means that it will detect the feature points in two or more images of the same scene which are supposed to be noisy, blurred, rotated and shifted with respect to each other (Kautsky, Zitova, Flusser & Peters 1998). There is one condition that needs to be fulfilled by the mutiframe feature point detection method, it is the condition of repeatability. This means that the detection results should not be influenced by the geometry of image, radiometric conditions and additive noise. The condition also sets that the group of points in all frames needs to be overlap at maximum level. The main intention for the proposed feature point detection is for the area of remote sensing, in which the scanning of the earth is done by satellite or high-flying aircraft in order to retrieve information about it. In this study the individual frames are assumed to have different contrast, be rotated and shifted with respect one another, be degraded by a linear shift-invariant blur, and be corrupted by the additive random noise (Kautsky, Zitova, Flusser & Peters 1998). Therefore the fundamental requirement is to have the

feature point detection that can work on different distorted frames, and this will lead to high repetition rate for the detection itself.

In order to manage different distorted images, this study proposed a new method or algorithm for feature point detection through the use of a parameter approach. In this study, the feature point can be defined as a point. A point is belonging to the two edges with an angle from the interval in between regardless of its orientation (Kautsky, Zitova, Flusser & Peters 1998). The following shows the steps for the proposed algorithm of feature point detection (Kautsky, Zitova, Flusser & Peters 1998):

1. The following inputs will be used for finding the feature points:
  - The image in a particular size.
  - The desired number of feature points.
  - The radius of neighbourhood for the mean value computation.
  - The interval where the angle between feature point candidate's edges has to be from.
  - The minimum allowed distance between feature points and a straight line.
  - The maximum allowed distance curvature divergence for straight line candidates.
  - The minimum allowed distance between two feature points.
2. Initialize the zero matrix that has the same size with the input image.
3. Calculate an image function of local mean values of input image, by using circular neighbourhood in a certain point with the input radius of neighbourhood.
4. Calculate the weight function of local variables.
5. Detect the feature point candidates. In this step it obtains the angle between edges and the number of them passing through each pixel. This can be obtained from the number and distribution of local sign changes in the difference of functions in Step 2.
6. Eliminate the false feature point candidates. At this point after the Step 5 there are some impermissible points which are treated as false feature point candidates. Those points are not the corners but close to a straight line, or are the true corners but having a small variation in the grey levels. After eliminating the false feature point candidates, the best feature points are chosen after maximizing the function. The function can quantify the significance for each



point. The algorithm will include the requirement that will not lead to two feature points closer to each other than the distance defined by user.

7. Select the feature points. In this step the algorithm will choose the feature points among a group of best feature points made in Step 6 that are satisfying the criteria set and have maximized weight function.

In order to verify the capabilities of the proposed feature point detection method or algorithm in this study, the experiment was carried out and demonstrated. The proposed method was compared with other classical methods from Kitchen and Rosenfeld, and Harris (Kautsky, Zitova, Flusser & Peters 1998). The experiment used satellite images for the testing since the intention of application for proposed method is in the area of remote sensing. In this experiment testing, those three feature point detection methods will be tested for their success rate to detect the number of identical feature points in both original and degraded or rotated frames. The experiment result showed that the proposed method was outperforming the other feature point detection methods, however it can only do it so when the images are heavily blurred and slightly rotated. But the Harris's feature point detection was performing better than the proposed method when the images are not significantly blurred and has more rotation angle (Kautsky, Zitova, Flusser & Peters 1998). The proposed method was comparable with Harris's feature point detection in all other cases. But the Kitchen and Rosenfeld's feature point detection had the worse success rate of detection in all cases. The result also showed that the proposed method was performing faster than others and its computational cost was also lower.

This study evinces that there are many ways to determine the feature points in an image. The proposed method in this study uses more user-defined parameters to identify the feature points. This can be a strength for it as the user can customize the result of feature point detection by changing the input parameters. The proposed method is also able to run its process with the lower usage of resources or computational cost. This is one of the important aspects because it can arouse the performance issue when using the detection in the system. Therefore it is desirable to be used to identify the feature points in an image. However it may not perform well when the image or frame is less blurred and has bigger rotation angle, it might generate less accurate results in this situation. Overall it is performing well in terms of speed and accuracy as it is well tested and has efficient algorithm implemented inside it.

## 2.4 Literature Review on Wavelet Transform

Wavelet transform is one of the mathematical terms that is used for carrying out the process of signal analysis when the signal frequency varies over time (MathWorks, n.d.). In mathematics, a series of wavelet is a representation of square-integrable function (real- or complex-valued measurable function). It is done by a particular orthonormal series (a type of vector) generated by the wavelet (Wavelet transform 2014). Wavelet transform is widely used in many multimedia fields, such as speech and audio processing, image and video processing, biomedical imaging, and 1-D and 2-D applications in communication and geophysics (MathWorks, n.d.). In image processing, wavelet transform can be used to compress and denoise the image. It is very useful in image processing as it helps to solve the problem of image distorting after compression. There are some studies (Gracemann & Miikkulainen 2005; Ruchika, Singh & Singh 2012) that discussed and proposed their own methods for applying the wavelet transform in the image processing.

The study done by Gracemann and Miikkulainen (2005) had the interesting approach for wavelet transforms to compress the image effectively. The study stated that the wavelet transform plays the important role in image compression. The wavelet-based image coders stated in the study are easily available, popular and can outperform the traditional coders based on discrete cosine transform (DCT) (Gracemann & Miikkulainen 2005). Its performance is relying on large extent of wavelet choice. Therefore the standard wavelets which are commonly used and performing well in compression of photographic images are applied in wavelet-based image coders. However there are some types of images that do not have same statistical characteristic with photographic images, for example the fingerprint images, medical images, satellite images and scanned documents (Gracemann & Miikkulainen 2005). This can cause the performance and accuracy of result for wavelet transform may be decreased due to the standard wavelets do not match those images and those images are frequently stored in large databases that have similar images.

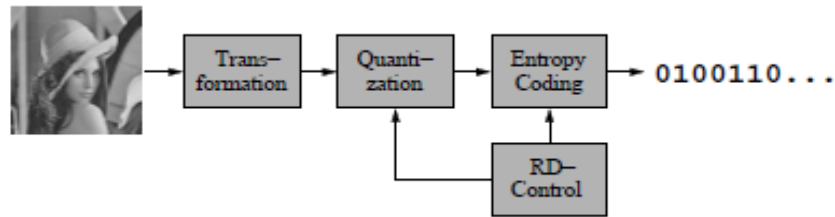


Figure 2-1: The structure of transform or image coder (Gracemann & Miikkulainen 2005).

In order to solve the problem mentioned before, a new algorithm was proposed in this study. The algorithm proposed was the coevolutionary genetic algorithm that can be used in wavelet transform. It is based on Enforced Sub-Populations (ESP) (Gracemann & Miikkulainen 2005) and Lifting (mathematical technique) that is used to find the wavelets which adapt to a certain class of images specifically. Lifting is an approach that has the effective way to create the complementary filter pairs. In order to construct a new filter pair from existing pair a lifting step which is a finite filter is used in lifting. There are two important characteristics of lifting that need to be concerned. One of them is the lifting will retain the biorthogonality of it, this means that the new filter pair will be definitely complementary which is same with the original pair no matter what lifting step is used. Another characteristic is that any wavelet with finite filters can be represented as a sequence of lifting steps (Gracemann & Miikkulainen 2005). These characteristics can make the lifting to become a very powerful tool to be used for creating the new wavelets.

While coevolutionary genetic algorithm used in wavelet transform is closely related to ESP neuroevolution algorithm. In ESP the number of populations of single neurons in parallel will be evolved by it. Its concept is that one neuron from each sub-population will be chosen repeatedly by ESP in order to form the candidate networks. This concept will be applied in the coevolutionary genetic algorithm. This means that the algorithm will evolve several populations of lifting steps in parallel, then randomly combine them to create the new wavelets (Gracemann & Miikkulainen 2005). The following figure shows how the algorithm works based on wavelet ESP.

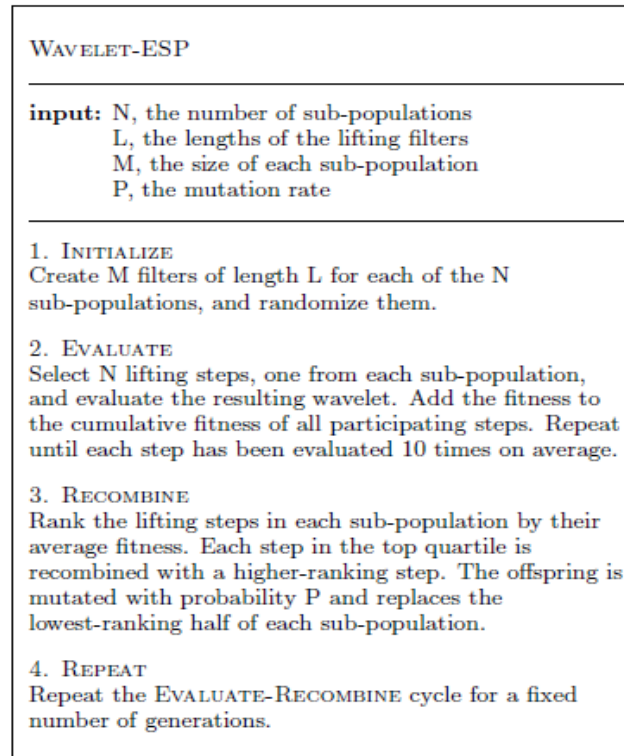


Figure 2-2: The ESP algorithm which is applied to the wavelets (Gracemann & Miikkulainen 2005).

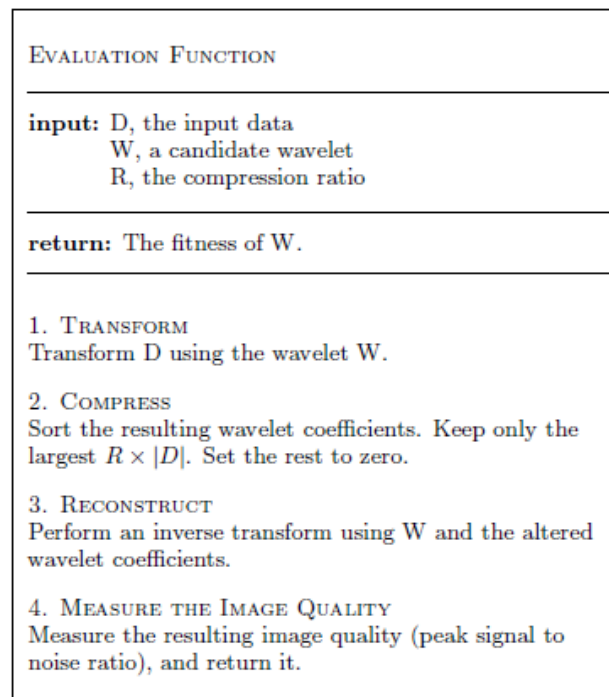


Figure 2-3: The evaluation function in the ESP algorithm (Gracemann & Miikkulainen 2005).

## Chapter 2 Literature Review

The evaluation function in the above figure is an idealized version of the image or transform coder. It uses only partial amount of wavelet coefficients for reconstruction and set the rest to zero, instead of entropy-encoding and quantizing the coefficients. This method is less costly and can help to select the compression ratio accurately, this can lead to the distortion outcome to be used directly as the fitness measurement for wavelet transform.

This study shows a good example of wavelet transform even though its main intention is focusing on non-photographic images like fingerprints and scanned documents. The proposed algorithm in the study provides an effective way for wavelet transform that can lead to producing better quality of images after compression, in other words the lossless compressed images. Lower cost of operation and higher accuracy can be its strengths when it is compared with other algorithms. Besides that, with the effective algorithm proposed in this study, the image or transform coder which uses the algorithm can compress the images with lesser time since it only uses certain amount of wavelet coefficient for wavelet reconstruction. The algorithm proposed can be effectively used on non-photographic images, but not on photographic images. It might produce inaccurate results when it is used on photographic images as both of them have different statistical properties. Therefore this limits the uses of the algorithm especially on the photographic images. Its concept and method described in this study can be taken as great reference for wavelet transform in image processing, specifically in image compression.

Other than the new coevolutionary genetic algorithm proposed for wavelet transform stated in previous study, there is another study that focuses on compressing the medical images using wavelet transform. The study conducted by Ruchika, Singh and Singh (2012) explained the importance of medical image in medical industry. Due to this reason the medical image compression plays the vital role for database storage and medical data transfer for the diagnosis purpose. The statistical redundancy which is presented in real world images has been utilized by various traditional image compression methods (Ruchika, Singh & Singh 2012). Reducing the redundancy can only produce a very small amount of compression after using the traditional image compression methods. And some important and non-redundant information or data must also be removed so that the higher ratios of compression can be achieved.

This study also stated that the medical image compression is a challenging technique in image processing because there are some high frequency elements in the image may contain relevant and significant information that is used for the purpose of diagnosis. In the medical image compression applications, if the compression approaches are able to retain all important and relevant image information needed, then the diagnosis done is considered as effective (Ruchika, Singh & Singh 2012). Due to this reason its applications will be suffering and ineffectively used because the compression approaches might not be able to retain all information of image needed, for example the telemedicine, and the fast searching and browsing of medical volumetric data. Therefore in order to encounter the problem stated, wavelet transform is used for the medical image compression. There are some characteristics of wavelet transform that will make the discrete wavelet transform (DWT) to become one of the most significant techniques for image compression. Those characteristics are multiresolution representation, energy compaction, blocking artefacts and decorrelation (Ruchika, Singh & Singh 2012). Because of this the wavelet transform is used in the proposed medical image compression methods.



Figure 2-4: Basic model of compression system (Ruchika, Singh & Singh 2012).

The new algorithm for medical image compression was proposed in this study. It is based on the basic model of compression system which uses the wavelet transform, as shown in Figure 2-4. It also uses Huffman encoding and threshold to reduce the redundancy of medical image information and DWT coefficients. In data redundancy reduction inside the basic model of compression system, it will remove highly correlated data which has low frequency details in the image (Ruchika, Singh & Singh 2012). In order to reduce the redundancy more effectively, DWT has been used in this compression system. It can remove non-significant information inside the image

effectively since it has energy compaction efficiency and high decorrelation process. While Huffman encoding can also reduce certain amount of redundancy of the data in image. One of the reasons is that the Huffman encoding is belonging to a category of code which has the variable in codeword length. This means that the individual symbols will have their own different length, as it leads the message to be encoded or represented with bit sequences (Ruchika, Singh & Singh 2012). Huffman encoding can reduce the redundancy through the different possibilities of incidents for different symbols. The shorter codewords will be encoded with the symbols which have higher probabilities of incidents, and vice versa. The following figure shows how the new proposed algorithm works on compressing the medical image through the use of DWT and Huffman encoding.

- For Compressing the Medical Image:**

  1. The DWT of the medical image is generated by obtaining wavelet decomposition coefficients for the desired levels. The numbers of levels are decided by the entropy of the image.
  2. A threshold for the decomposed image coefficients is selected, below which all the coefficients are made zero. This reduces the band space of the image signal, as large number of coefficients are made zero.
  3. Huffman encoding on the thresholded coefficients is applied to reduce the redundancy in the coefficient data.
  4. The thresholded and Huffman encoded coefficients are saved instead of the image.

**For Uncompressing the Medical Image:**

  1. When the image is to be uncompressed, the Huffman decoding is done on the coefficients and the threshold coefficients are obtained.
  2. Image is regenerated from these threshold coefficients by taking inverse discrete wavelet transform (IDWT).

Figure 2-5: Thee compression algorithm proposed in the study (Ruchika, Singh & Singh 2012).

Although this study is more focusing on compressing the medical image, its wavelet transform application on the new proposed compression algorithm can be referred and implemented to some extent. This compression algorithm using the wavelet transform can provide more accurate results as it removes the redundant information in image

## Chapter 2 Literature Review

which may contribute to producing less exact outcome of compressed image. Besides that with the use of Huffman encoding the proposed algorithm is able to compress the medical images effectively without losing the significant data. This means that the lossless compressed images are produced in the end of compression process. In the image compression lossless is one of the very important properties of good compressed image, therefore the algorithm proposed in this study can be useful for compressing the images. However there are some wavelets from same domain which is medical images that might not be suitable to be used for the algorithm as the information properties among them are varies and different from each other. Thus the results generated might be not accurate based on the wavelets used from the images.



## 2.5 Summary of Literature Review on Information Gain

Table 2-1: The summary of literature review on information gain.

<b>Algorithm Proposed / Algorithm Studied</b>	<b>Strengths</b>	<b>Limitations</b>
Performance of the Kullback-Leibler information gain for predicting image fidelity by Garcia, Fdez-Valdivia, Rodriguez-S´anchez and Fdez-Vidal (2002)		
–Kullback-Leibler information gain	<ul style="list-style-type: none"> <li>–It can be used with minimum number of properties.</li> <li>–It can detect differences between two images with minimum level of errors.</li> <li>–Its accuracy for detection is consistent among the images.</li> </ul>	<ul style="list-style-type: none"> <li>–Its execution time is twice longer than other algorithms.</li> <li>–There is some difficulties to implement algorithm.</li> </ul>
Minimum error gain for predicting visual target distinctness by Garcia, Fdez-Valdivia, Rodriguez-S´anchez, Fdez-Vidal and Fuertes (2001)		
<ul style="list-style-type: none"> <li>–Root Mean Square Error (RMSE)</li> <li>–Selective gain</li> <li>–Compound gain</li> <li>–Minimum Error Information Gain (MEG)</li> <li>–Minimum Compound Gain (MEC)</li> </ul>	<ul style="list-style-type: none"> <li>–It can detect the objects in an image easily even though there are some significant noises presented.</li> <li>–It has lower probability of error occurred for detecting the objects in an image.</li> </ul>	<ul style="list-style-type: none"> <li>–Its execution time might be relatively slower when there are more and more noises presented in an image.</li> </ul>

## 2.6 Summary of Literature Review on Skin Colour Tone Model

Table 2-2: The summary of literature review on skin colour tone model.

Algorithm Proposed / Algorithm Studied	Strengths	Limitations
<b>A Robust Skin Colour Based Face Detection Algorithm by Singh, Chauhan, Vatsa and Singh (2003)</b>		
<ul style="list-style-type: none"> <li>–Combination of three colour space, they are Red, Green and Blue (RGB) colour space, YCbCr colour space and Hue, Saturation and Intensity (HSI) colour space</li> </ul>	<ul style="list-style-type: none"> <li>–It is able to detect various skin colours of humans from different races accurately.</li> <li>–It has lower false detection and false dismissal rate for detecting faces in the image.</li> </ul>	<ul style="list-style-type: none"> <li>–It may not effectively detect the human faces in various lighting conditions.</li> <li>–Its false detection and false dismissal rate may be increased when there are more noises presented in the image.</li> </ul>
<b>A New Algorithm for Human Face Detection Using Skin Colour Tone by Zangana and Al-Shaikhli (2013)</b>		
<ul style="list-style-type: none"> <li>–Combination of two colour space, they are Red, Green and Blue (RGB) colour space and YcbCr colour space, and YC'bC'r colour space will be obtained from YcbCr colour space</li> </ul>	<ul style="list-style-type: none"> <li>–It can detect the human faces in the image in various lighting conditions.</li> <li>–It is able to detect various skin colours of humans from different races accurately.</li> <li>–The process will be faster as the image is resized before it is being processed.</li> </ul>	<ul style="list-style-type: none"> <li>–The face detection may not be accurate because the image is resized that may lead to distortion of the image sharpness.</li> <li>–It may need more resources in computation as the process of algorithm is complex.</li> </ul>
<b>Segmentation Algorithm for Multiple Face Detection in Colour Images with Skin Tone Regions using Colour Spaces and Edge Detection Techniques by Lakshmi and PatilKulakarni (2010)</b>		
<ul style="list-style-type: none"> <li>–Combination of HSI and YcbCr colour space models, using Canny and Prewitt Edge Detection</li> <li>–Robert Cross Edge Detection</li> </ul>	<ul style="list-style-type: none"> <li>–It can detect multiple faces in the image at the same time.</li> <li>–It is able to detect the human skins with various lighting conditions in the image.</li> <li>–It can detect the human skins which have different skin colours.</li> </ul>	<ul style="list-style-type: none"> <li>–It cannot effectively detect human faces when there are some objects that have similar skin colours with human skins.</li> </ul>

## 2.7 Summary of Literature Review on Feature Point Detection

Table 2-3: The summary of literature review on feature point detection.

<b>Algorithm Proposed / Algorithm Studied</b>	<b>Strengths</b>	<b>Limitations</b>
<b>Feature Point Detection for Real Time Applications by Nain, Laxmi and Bhadviya (2008)</b>		
<ul style="list-style-type: none"> <li>– Short algorithm for feature point detection that uses smaller mask</li> </ul>	<ul style="list-style-type: none"> <li>– Less resources are needed for performing the process.</li> <li>– Most of the errors can be minimized during the process.</li> <li>– It is easy to be implemented.</li> </ul>	<ul style="list-style-type: none"> <li>– Time consuming as the process in algorithm compares and computes threshold twice.</li> <li>– Results generated may be less accurate as simple mask is used.</li> </ul>
<b>Feature point detection in blurred images by Kautsky, Zitova, Flusser and Peters (1998)</b>		
<ul style="list-style-type: none"> <li>– Algorithm for feature point detection that uses a parameter approach</li> </ul>	<ul style="list-style-type: none"> <li>– It uses more user-defined parameters that will lead to more variety results generated.</li> <li>– It can run the operation with lower computational cost.</li> <li>– Its results generated will be more accurate as the false feature points candidates are eliminated in the process.</li> </ul>	<ul style="list-style-type: none"> <li>– Might not perform well in various condition, for example the image is less blurred and has bigger rotation angle.</li> </ul>

## 2.8 Summary of Literature Review on Wavelet Transform

Table 2-4: The summary of literature review on wavelet transform.

Algorithm Proposed / Algorithm Studied	Strengths	Limitations
Effective Image Compression using Evolved Wavelets by Gracemann and Miikkulainen (2005)		
–Coevolutionary genetic algorithm used in wavelet transform	<ul style="list-style-type: none"> <li>–It requires less resources to run the process and yet has high accuracy for compressing images.</li> <li>–Due to the use of certain amount of wavelet coefficient lesser time is required to compress the images, even though it involves several evaluation in the process.</li> </ul>	–Its usage is limited to non-photographic images as both photographic and non-photographic images have different statistical properties.
Compression of Medical Images Using Wavelet Transforms by Ruchika, Singh and Singh (2012)		
–Algorithm based on the basic model of compression system and uses Huffman encoding, DWT and threshold	<ul style="list-style-type: none"> <li>–It can compress the images without losing significant data in the images.</li> <li>–Removing redundancy in the algorithm can lead to producing more accurate results after compressing the images.</li> <li>–The images compressed can be easily uncompressed as the algorithm has uncompressing function.</li> </ul>	<ul style="list-style-type: none"> <li>–It might be more difficult to be implemented compared to others as it involves a lot of processes and levels.</li> <li>–The wavelets from the same domain might not be suitable to be used in algorithm since their information properties are different from each other.</li> </ul>

## 2.9 Review of Existing Software

There are many existing image editing software available in the market that can help the photographers to edit their captured images or photos. In this section there is a software that will be reviewed in order to express the problem stated in Section 1.1 of Chapter 1. That software is the Adobe Photoshop Creative Suite 6 (CS6) which is developed and published by Adobe Systems. It is one of the most popular raster graphic editor software that provides industry standard services to individuals as well as multimedia organizations in the competitive market.

Most of the photographers nowadays will use Adobe Photoshop CS6 to embellish their images captured. They will also use it to place the logo in a certain position of the images. The following will show how photographers as the users place or tag a logo in an image by using Adobe Photoshop CS6:

1. Open an image and a logo in Photoshop, the logo should be in PSD (Photoshop Document, a layered image and default format for saving the data used in Photoshop) or PNG format. This can allow the image and logo to be in semi-transparent background.

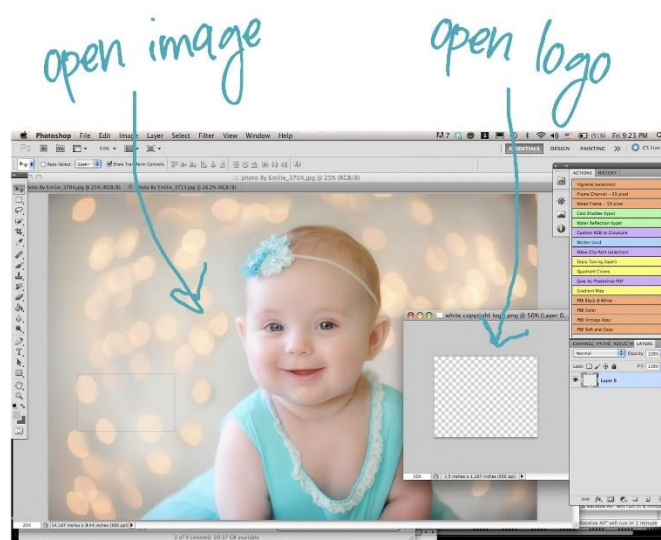


Figure 2-6: Illustration for Step 1 (Smith 2012).

## Chapter 2 Literature Review

2. Use the Move tool in Photoshop (or using shortcut “V”) to click and drag the logo onto the image.

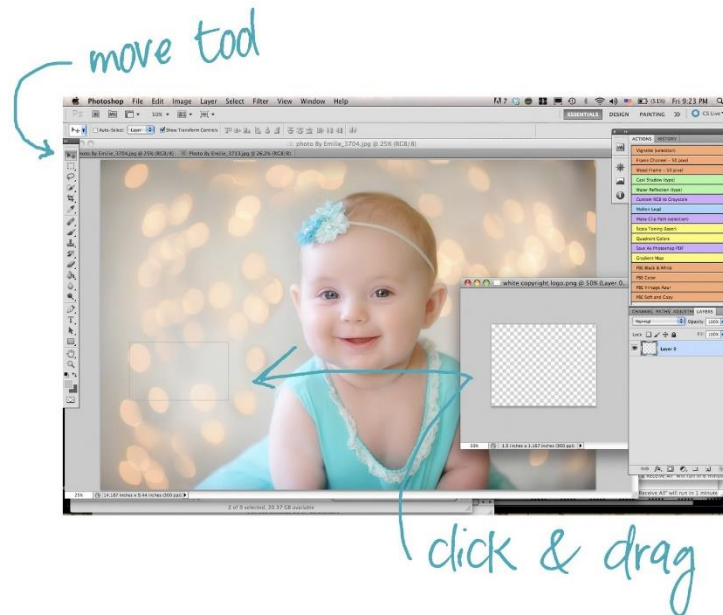


Figure 2-7: Illustration for Step 2 (Smith 2012).

3. Check in the checkbox that indicates “Show Transform Controls”. It will allow the logo to be changed in size freely.

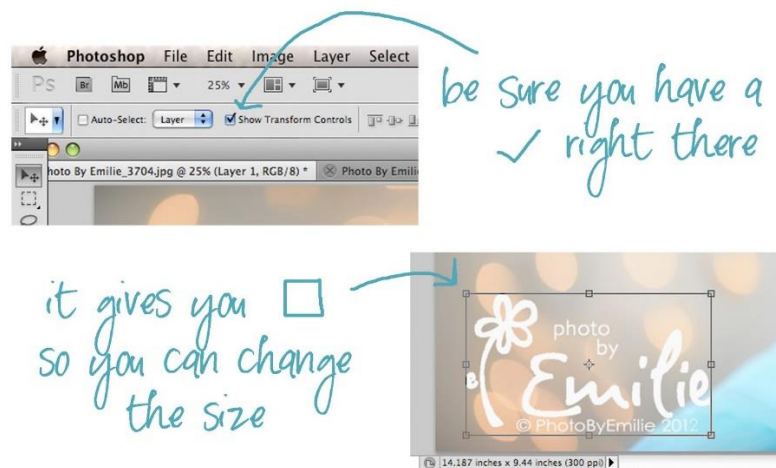


Figure 2-8: Illustration for Step 3 (Smith 2012).

## Chapter 2 Literature Review

4. Use the Move tool in Photoshop (or using shortcut “V”) to change the size of the logo and place it to the position desired in the image.



Figure 2-9: Illustration for Step 4 (Smith 2012).

The above steps show that the photographers can freely resize and place the logo anywhere that they desire in an image. The following will show how to automatically place a logo in many images at the same time, at the same fixed position in many images:

## Chapter 2 Literature Review

1. Insert an image and a logo into Photoshop CS6. Then select Window > Action, and the Action window will be popped out.

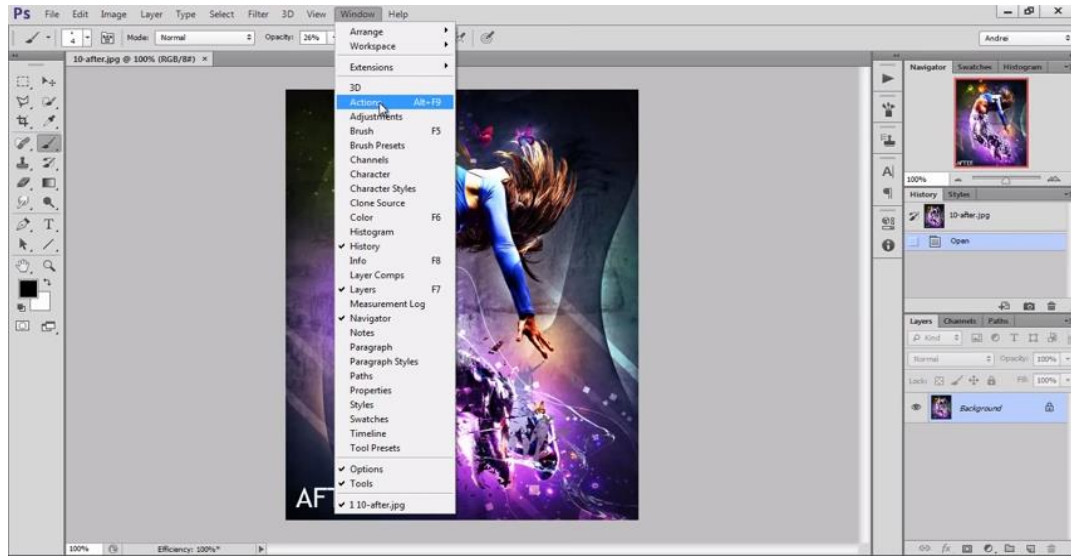


Figure 2-10: Illustration for Step 1 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

2. Press a file icon at the window and it will create a folder to hold the action. Then click the “Create New Action” button which is beside the file icon pressed just now. Another window will also be popped out.

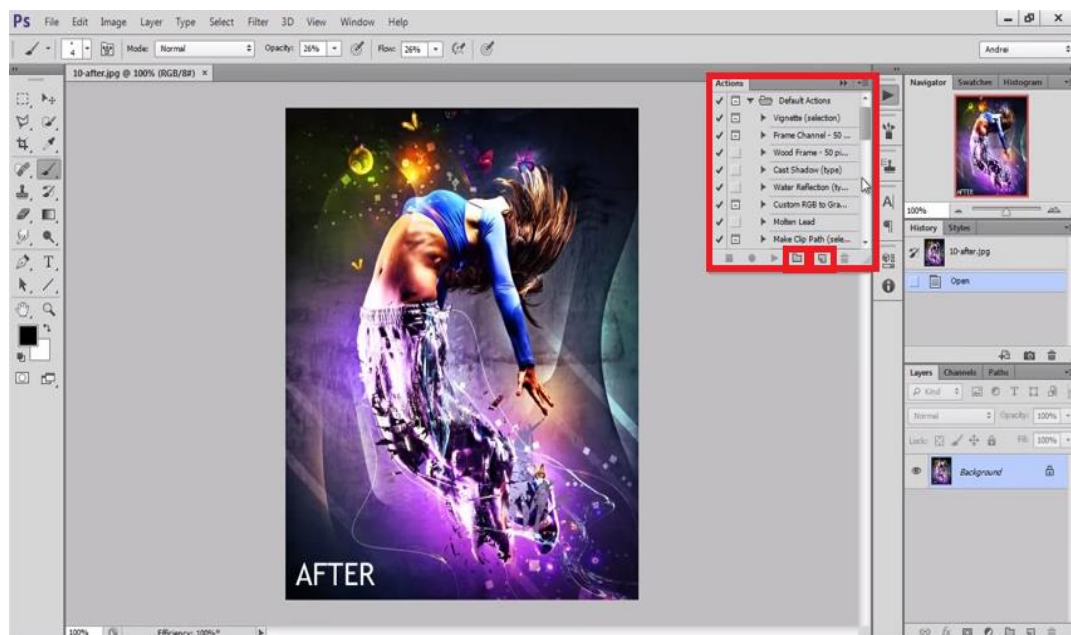


Figure 2-11: Illustration for Step 2 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).



## Chapter 2 Literature Review

3. At the window popped out, just set the name and a set of function key (used to start the action) for the action. And then press record button.

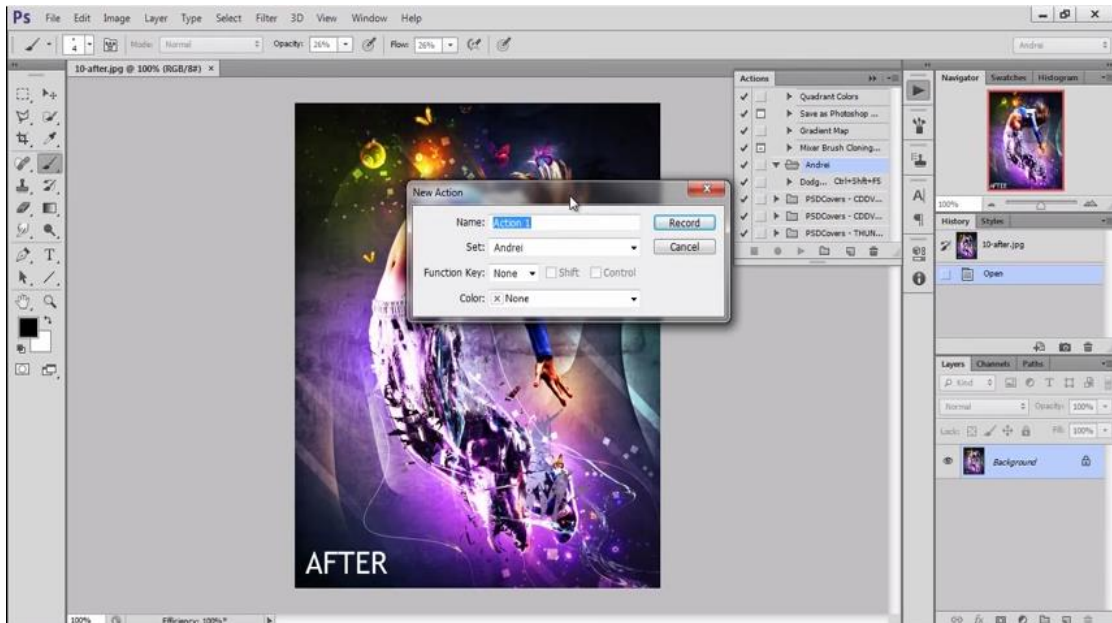


Figure 2-12: Illustration for Step 3 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

4. Once the action is finished setting up. Press the square button (Record Action button) which is left side of the file icon to start recording the action.

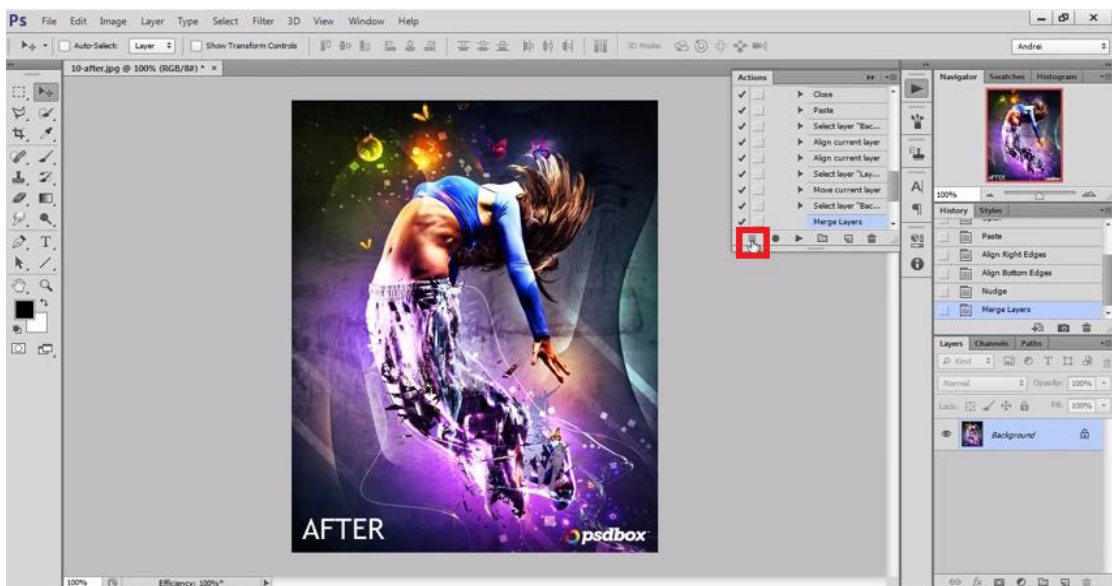


Figure 2-13: Illustration for Step 4 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

## Chapter 2 Literature Review

5. Place the logo in the image accordingly and these actions will be recorded. After placing logo properly in the image, press the red circle button (Stop Recording Action button) beside square button to stop recording the action.

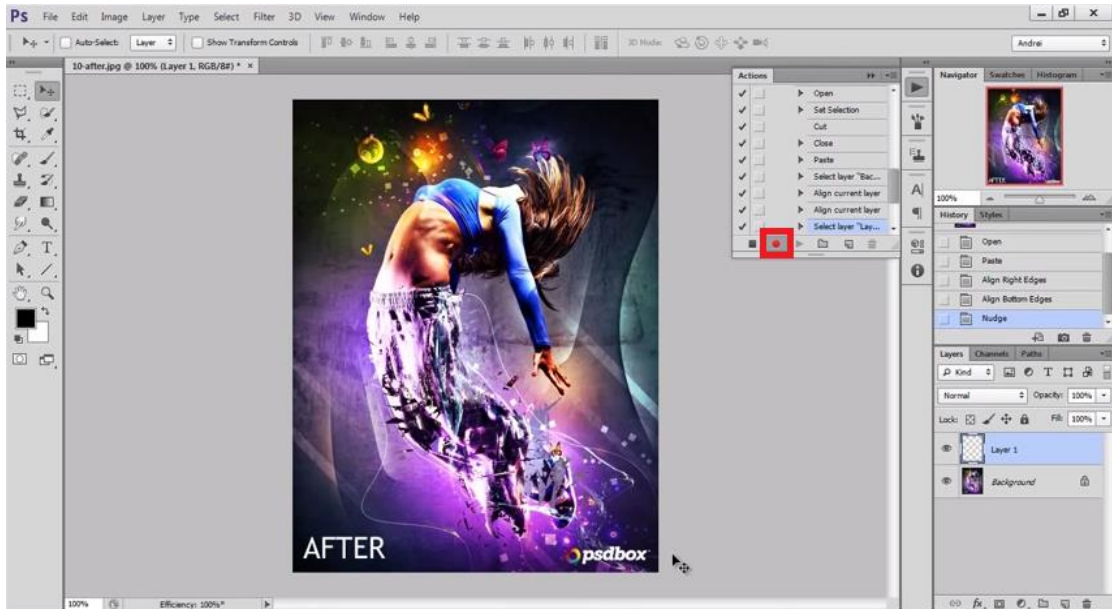


Figure 2-14: Illustration for Step 5 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

6. After stop recording the action, select File > Scripts > Image Processor to configure the recorded action for other images.

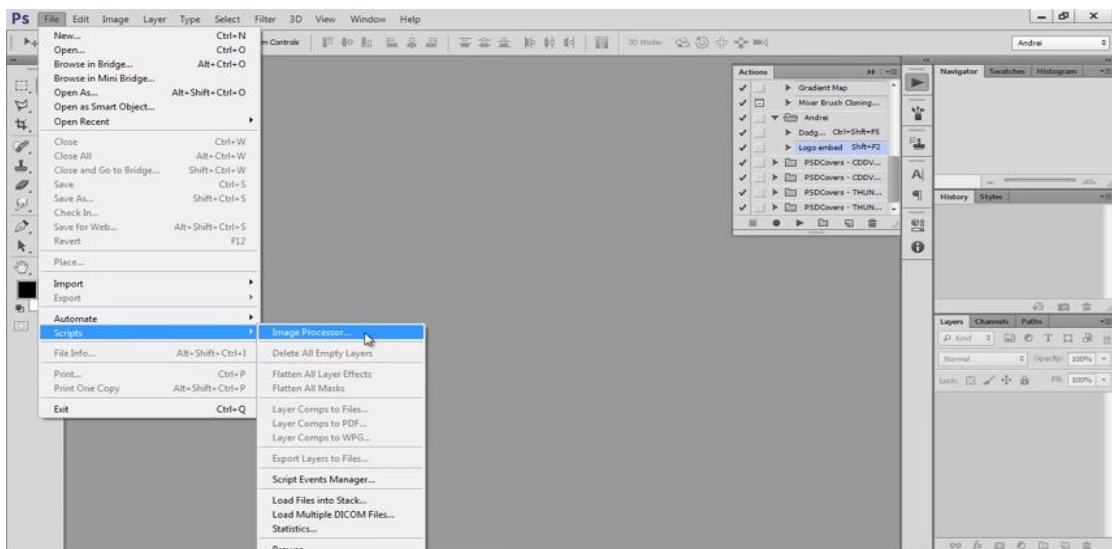


Figure 2-15: Illustration for Step 6 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

## Chapter 2 Literature Review

7. In the Image Processor window, select the file that contains the images which are needed to be processed, and also select the destination file to save the processed images. Then choose the action recorded that will be used to process the images. After that click Run button to start the action.

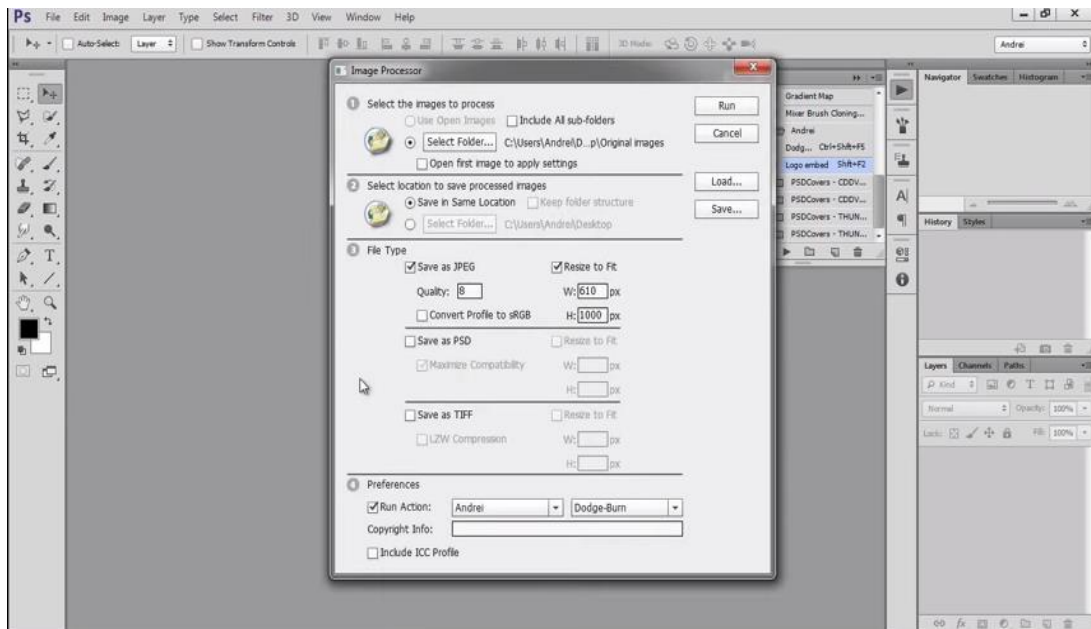


Figure 2-16: Illustration for Step 7 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

8. Photoshop will run the action automatically for all images selected.

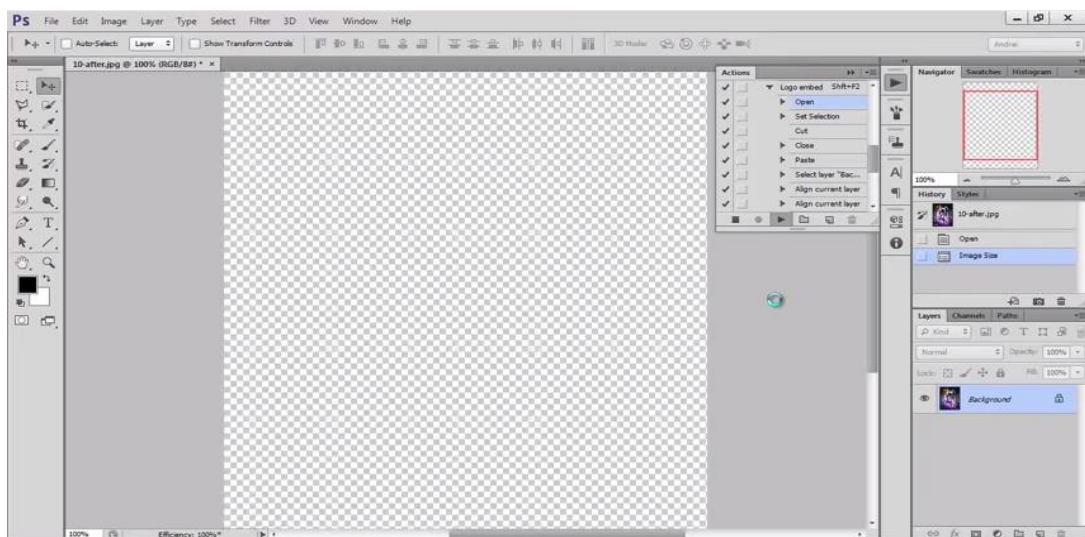


Figure 2-17: Illustration for Step 8 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

9. After completing the action for each image, the images processed will be saved to the destination folder specified before. The logo will be placed at the same position in each image selected.

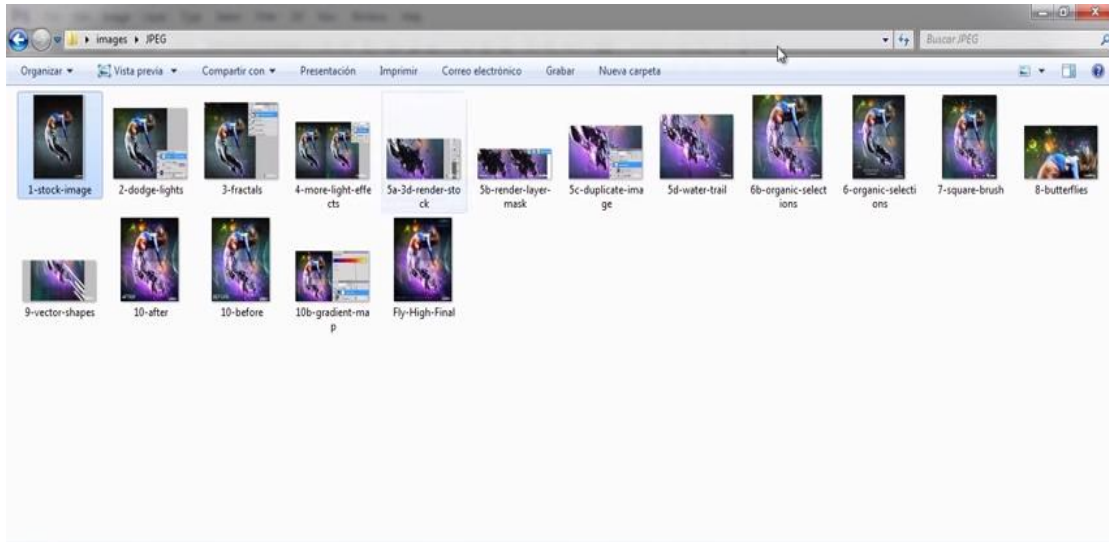


Figure 2-18: Illustration for Step 9 (Resize & Watermark Multiple Images Automatically in Photoshop CS6 2013).

From the steps shown above, Photoshop CS6 can efficiently place the logo in each image automatically. But there are some cases when the logo in some images cannot be easily found as some certain positions in the images will cover up the logo. Therefore the photographers either need to manually place the logo in images one by one or using automatic way to do those tasks. Besides that, Photoshop CS6 is also able to create the frame for each image and place the logo at the frame (as shown in Figure 2-14). This can be done by using similar process that places the logo in each image as shown just now. The following will show the strengths and weaknesses for using Photoshop CS6 to tag or place the logo in images:



Table 2-5: The strengths and limitation for using Photoshop CS6 to tag the logo in images.

Strengths	Limitations
<ul style="list-style-type: none"> <li>-It can handle many images to be processed in a certain period of time.</li> <li>-It provides greater flexibility on how the logo can be shaped and placed in an image.</li> </ul>	<ul style="list-style-type: none"> <li>-It cannot detect the best position for the logo to be placed in an image.</li> <li>-It might not be user friendly for some certain users as it can be confusing for them.</li> <li>-It may consume more resources for processing the images when there are some complex actions need to be done and more images need to be processed by it.</li> </ul>

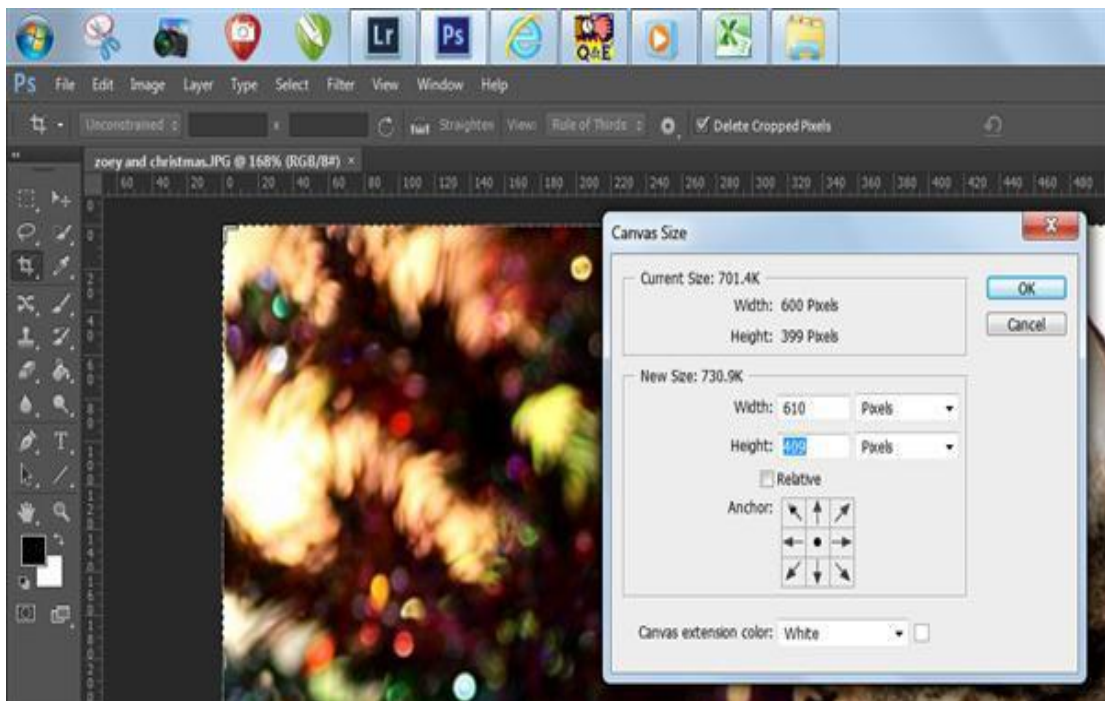


Figure 2-19: One of the ways to add the frame for an image (Ray 2012).

## Chapter 3 Methodology

### 3.1 Block Diagram

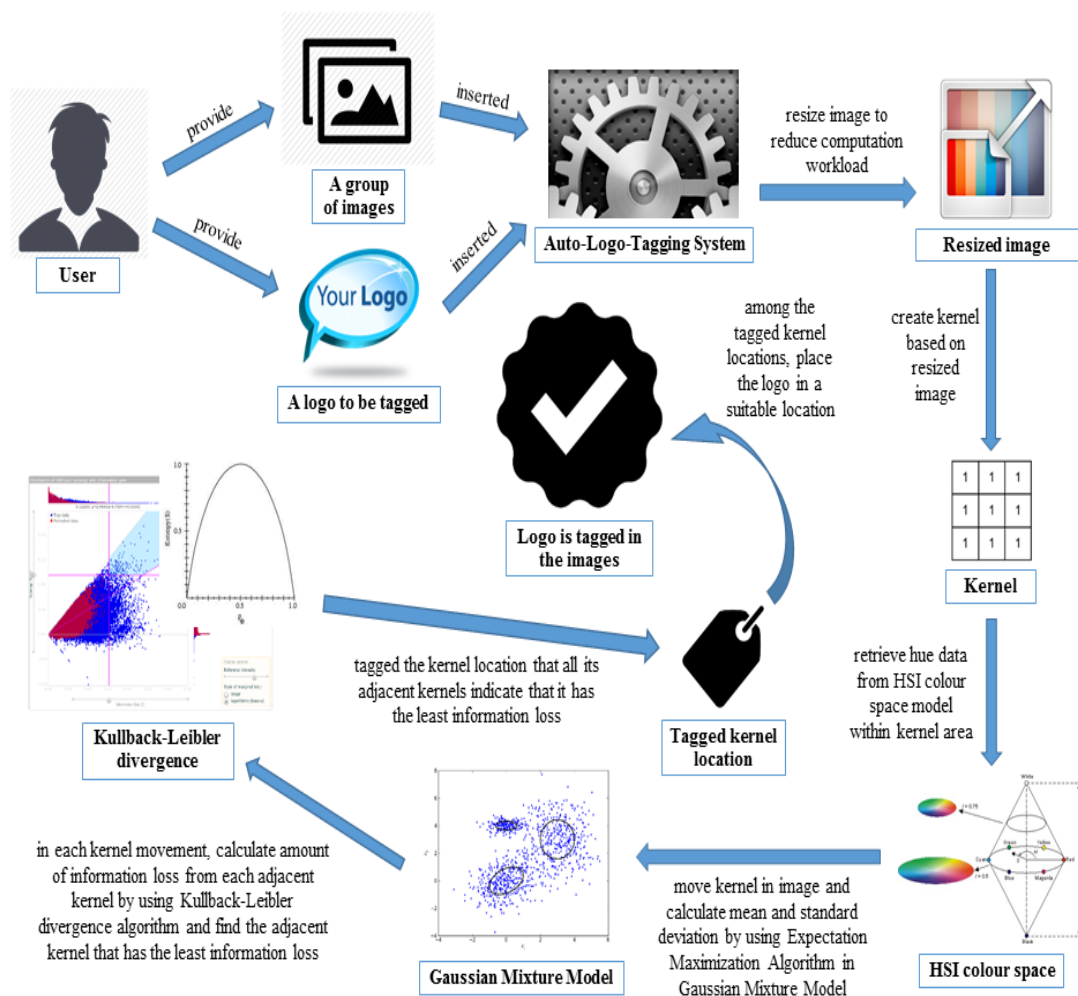


Figure 3-1: Block Diagram for illustrating overall process for Auto-Logo-Tagging System.

In order to solve the problem of tagging logo, there are three sections in this Auto-Logo-Tagging System that are divided so that it will be easy to be developed. Those modules are Calculate Mean and Standard Deviation in Moving Kernel, Compute Information Loss from Adjacent Kernels and Find Suitable Location for Logo Placement.

### 3.2 System Methodology

#### 3.2.1 Calculate Mean and Standard Deviation in Moving Kernel

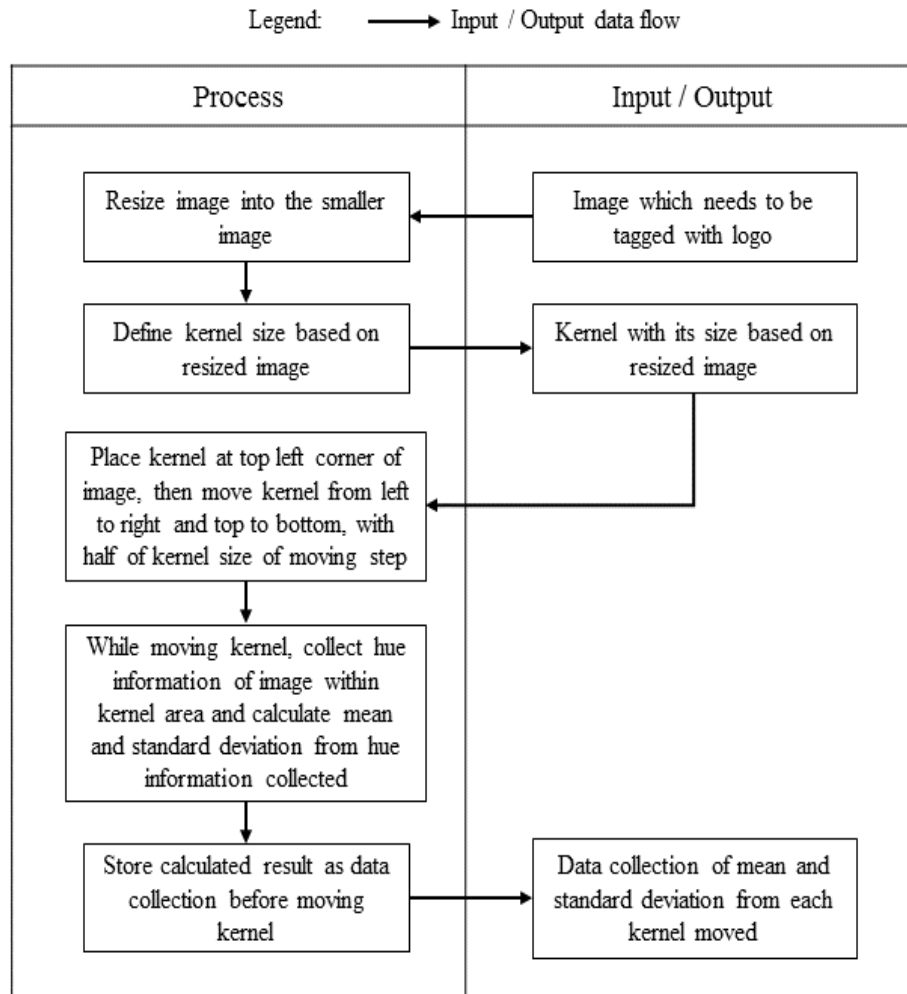


Figure 3-2: The input-process-output flow chart for Calculate Mean and Standard Deviation in Moving Kernel Module.

At first, when the image is inserted into the system it will be resized to a smaller image in order to decrease the workload for computation. Then the system will create a kernel and its size is based on the resized image, this means that its width and length are certain percentage of the width and length of resized image. After the kernel is created it will be placed at top left corner of resized image. The kernel is moved from left to right and top to bottom, its moving step is half of the kernel size. Before it is moved, the system collects hue information from HSI colour space model within kernel area. The hue

information collected will be used to calculate mean and standard deviation that represent hue information.

The mean and standard deviation are calculated by using Expectation Maximization algorithm in Gaussian Mixture Model. Gaussian Mixture Model is used to assume that all the data points in probabilistic model are generated from a mixture of a finite number of Gaussian distributions with unknown parameters (scikit-learn n.d.). This model can be seen as generalizing k-means clustering which can incorporate information about the structure of the data and the centres of latent Gaussians (scikit-learn n.d.). While Expectation Maximization algorithm will help Gaussian Mixture Model to estimate parameters in probabilistic models with incomplete data (Chuong & Batzoglu 2008). It is used for finding parameters of statistical models in maximum likelihood estimates where unobserved latent variables are depended by the model (Expectation-maximization algorithm 2015). The algorithm helps to determine the approximate mean and standard deviation in a group of hue data which is collected earlier.

After mean and standard deviation in current kernel are calculated, the kernel will be moved to another position to calculate another mean and standard deviation from hue information in moved kernel area. This process is repeated until the kernel is moved to bottom right corner of resized image. Those calculated mean and standard deviation will be stored as data collection for further calculation.



### 3.2.2 Compute Information Loss from Adjacent Kernels

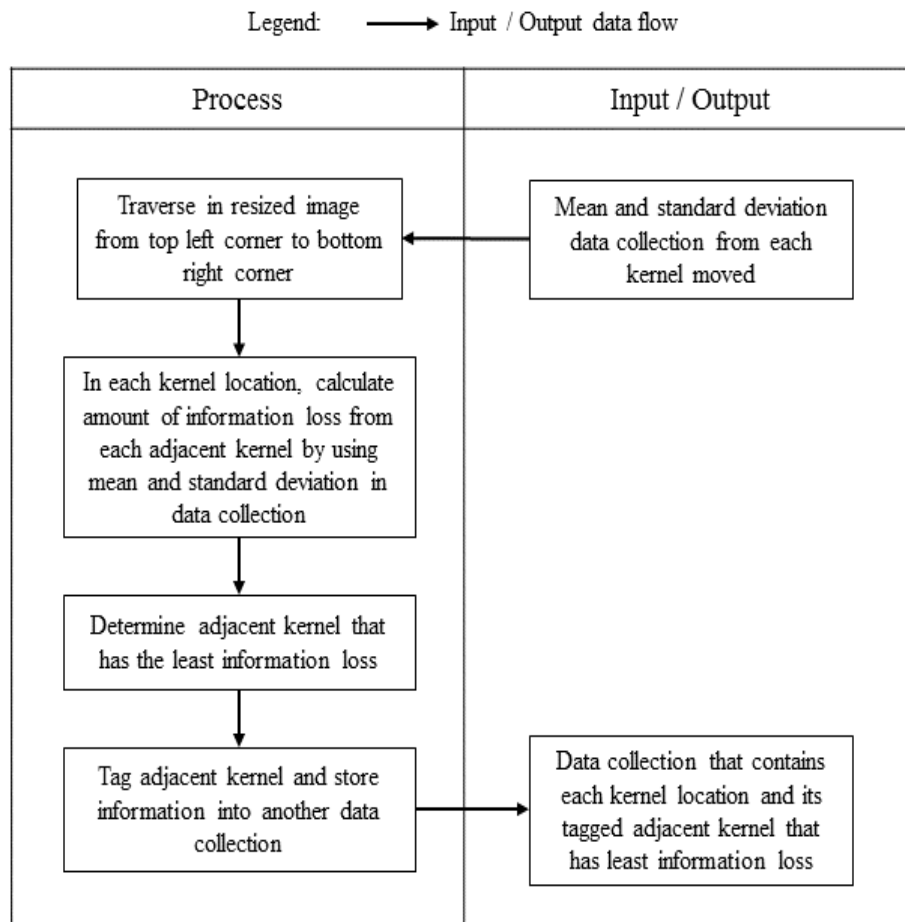


Figure 3-3: The input-process-output flow chart for Compute Information Loss from Adjacent Kernels Module.

Once mean and standard deviation from each kernel movement are calculated and collected, the system will traverse the resized image again by using the kernel created earlier. Similar to the previous steps, the system starts traversing from top left corner to bottom right corner in resized image. When the system traverses the resized image, in each current kernel location it will calculate the information loss from each adjacent kernel. Those adjacent kernels are from left, bottom, right and top of the current kernel location. The information loss is calculated by using the Kullback-Leibler divergence algorithm. It can indicate whether there is any changes from current kernel location to another kernel location when moving based on the amount of information loss. The

### Chapter 3 Methodology

mean and standard deviation in data collection which are calculated in previous steps are needed to compute the information loss.

In current kernel location, after finishing calculating the information loss from adjacent kernels, the system will identify which adjacent kernel has the least amount of information loss. The system will tag that adjacent kernel and store the related information into another data collection. After that it will move to next kernel location and repeat the same processes.

### 3.2.3 Find Suitable Location for Logo Placement

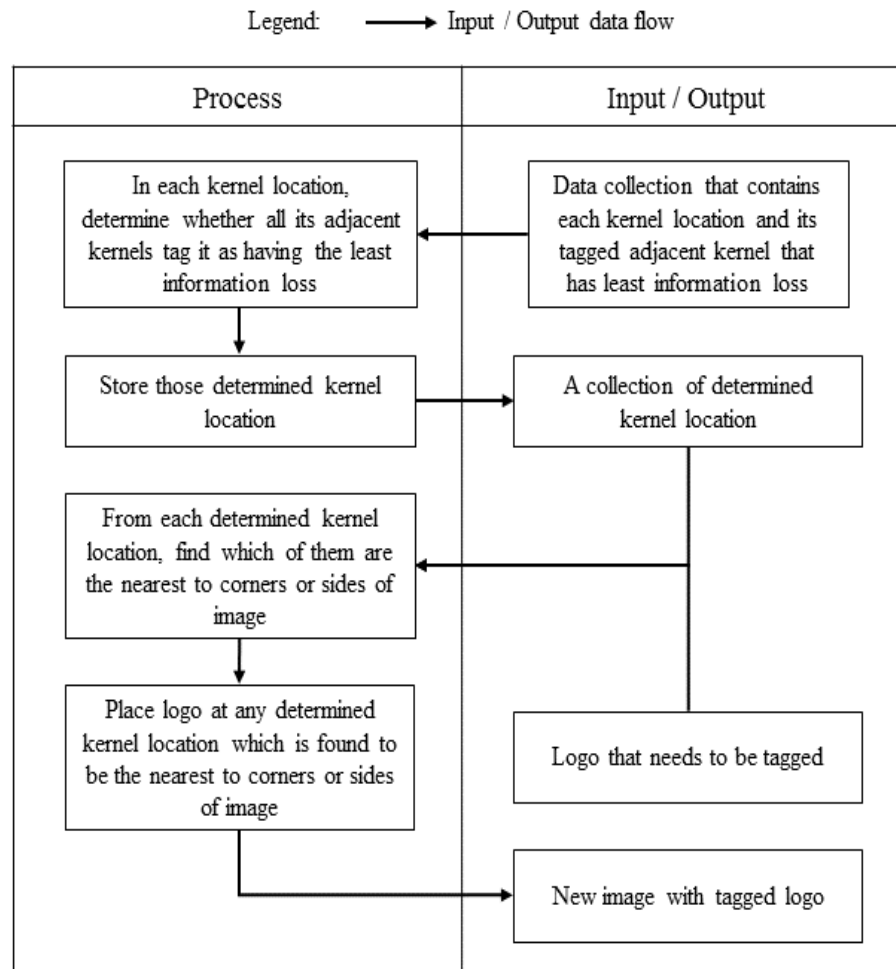


Figure 3-4: The input-process-output flow chart for Find Suitable Location for Logo Placement Module.

After calculating the information loss and tagging the adjacent kernel which has the least information loss in each kernel location, the system will find a suitable location to place the logo so that the logo can be shown clearly in the image. In each kernel location, the system will determine whether all its adjacent kernels are tagging it as having the least information loss. It uses the data collection that contains each kernel location and its tagged adjacent kernel which has been created earlier in the process of finding. Those determined kernel location will be stored into another data collection for further use.

After that the system will use this data collection of determined kernel location to find a proper location for logo to be placed in the image. In order to do this, the system will

### Chapter 3 Methodology

check the determined kernel locations in data collection one by one to find out whether each of them is the nearest to the sides or corners of the image. By placing the logo near to the sides or corners of image, the people can easily recognize or detect the logo placed in image. The determined kernel location is ignored if it is in the skin region. The logo will not be placed in skin region so that it would not cover this region in image, instead the system will place it in background of the image. Then the system places the logo at the kernel location which is found to be the nearest to corners or sides of the image. In the end an image with the tagged logo will be produced as an output of the system.

### 3.3 Experiment Setting

In the beginning of this project, there are about two experiments will be set in the project prototype which helps to build or establish the fundamental functions of the Auto-Logo-Tagging System. Those experiments are the Identification of Background and Foreground Using Kullback-Leibler Information Gain and the Identification of Human Skin Using HSI Colour Space Model.

#### 3.3.1 Identification of Background and Foreground Using Kullback-Leibler Information Gain

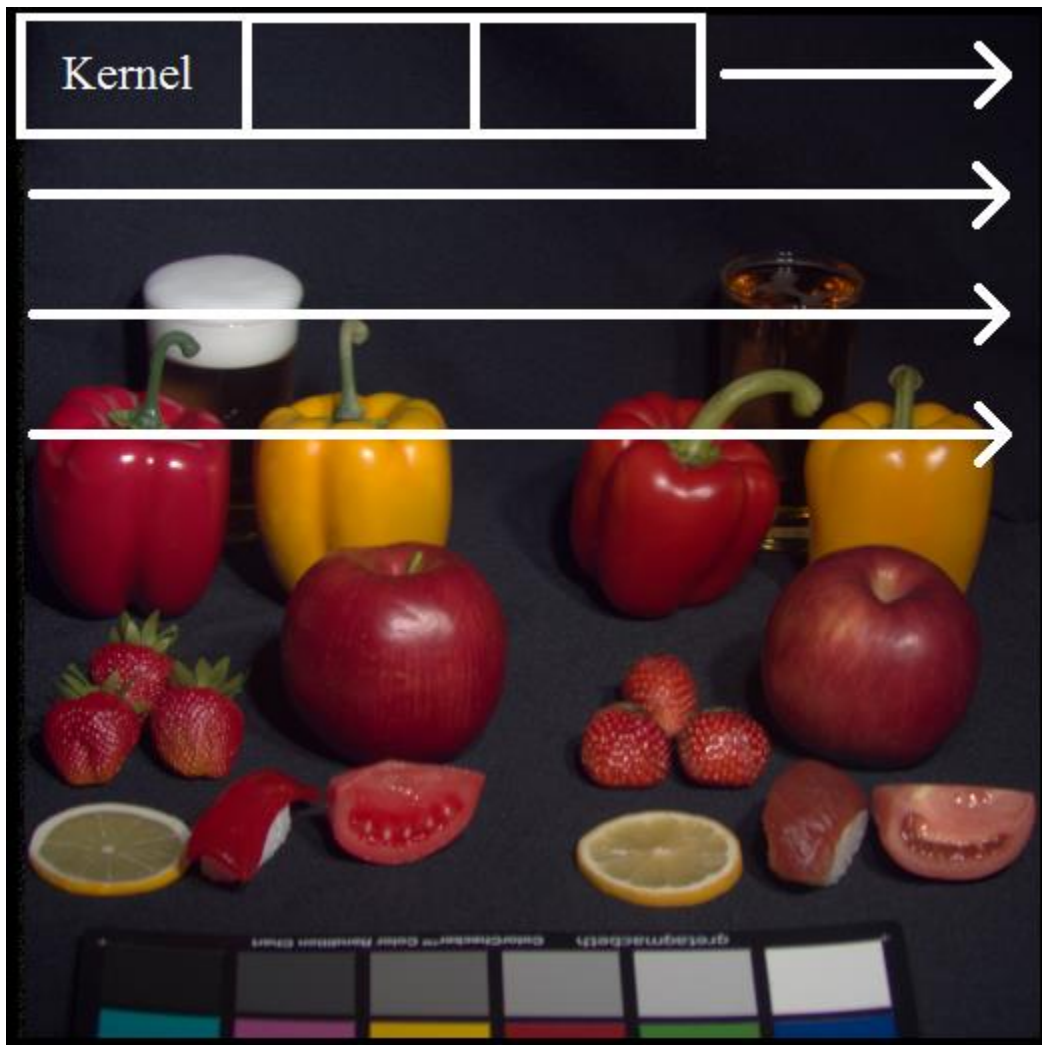


Figure 3-5: The traversing of kernel in the image (Columbia University, n.d.).

```
public void getCustomKernel(int[,] imageK, Byte[,] imageData, int oRow, int oCol, int lRow, int lCol)
{
    int r = lRow / 2;
    int c = lCol / 2;

    for (int row = r; row < oRow - r; row++)
    {
        for (int col = c; col < oCol - c; col++)
        {
            imageK[row, col] = new int[lRow * lCol];

            for (int kr = 0; kr < lRow; kr++)
            {
                for (int kc = 0; kc < lCol; kc++)
                {
                    imageK[row, col][kr * lRow + kc] = imageData[row - r + kr, col - c + kc, 0];
                }
            }
        }
    }
}
```

Figure 3-6: The code for creating custom kernel based on size of the logo.

Figure 3-5 shows how the kernel is traversing throughout the image. The kernel size is based on the size of logo which is needed to be tagged in image. The experimental system will use the kernel to traverse the image iteratively, and in each iteration it will calculate the value differences from image. Once the traversing and calculation are done, the experimental system will gather all values calculated and form the probability distribution for the image. This process is similar with the procedure in Chapter 3.2.2 Gather Probability Distribution from Image. The Figure 3-6 shows the C# programming code for creating the custom kernel based on size of the logo. It is useful as there are various sizes of logo that will be used to be tagged in image, so it would be easier for the experimental system to create the kernel dynamically.

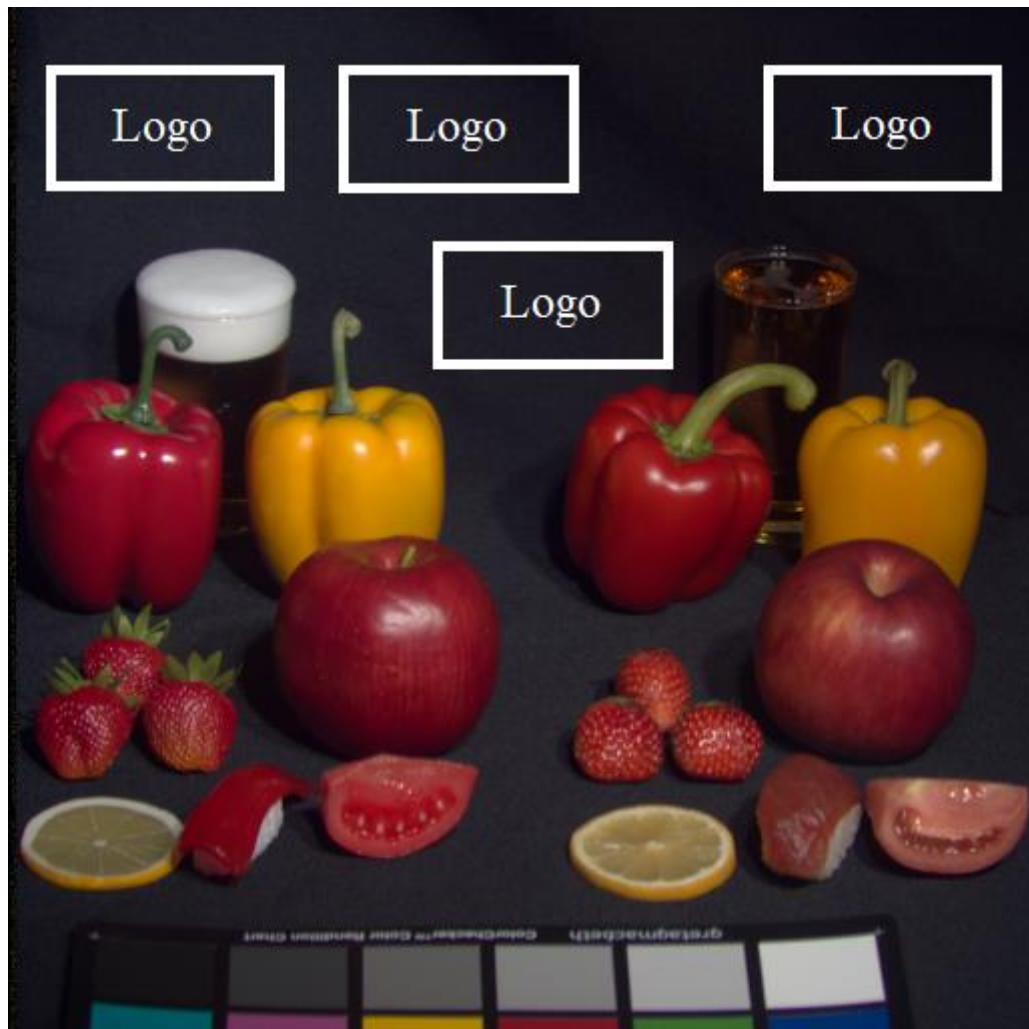


Figure 3-7: Expected result in the experiment (Columbia University, n.d.).

Figure 3-7 shows the expected result which is critical in this experiment. This is because with the use of probability distribution formed, the foreground and background in the image should be identified. It can be done by finding the lower information gain in probability distribution. The finding can be executed by using Kullback-Leibler information gain algorithm. Once the foreground and background are identified, the placement of logo can be easily done as the logo will be placed at background in the image. So the experimental system should be expected to identify the logo placement locations as shown in Figure 3-7. After that it will compute the score in each logo placement location and the highest score will be selected to place the logo in image.

### 3.3.2 Identification of Human Skin Using HSI Colour Space Model

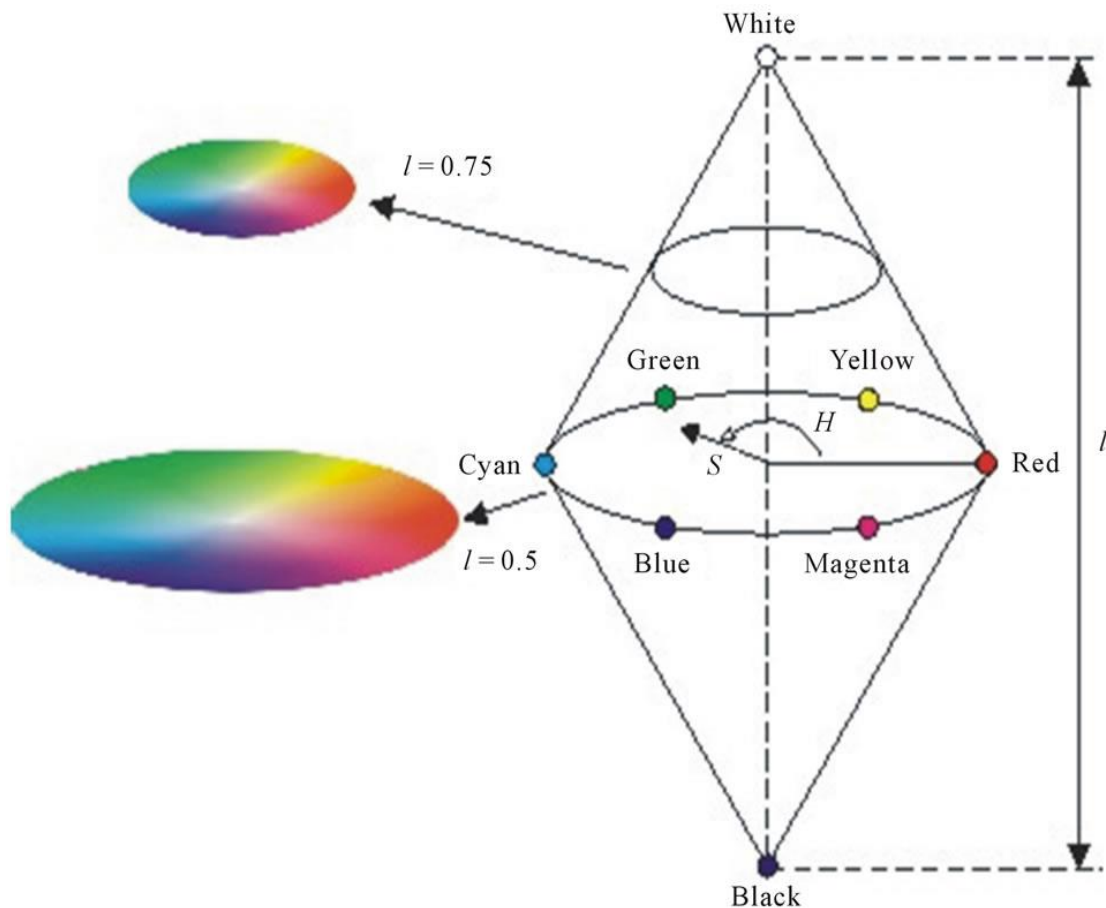


Figure 3-8: The HSI colour space model (Maraqa, Al-Zboun, Dhyabat & Abu Zitar 2012).

In this experiment, the HSI colour space model is used in order to identify the skin regions in image and find out the location which is suitable for logo to be placed without interfering the main objects or skin regions in the image. At first the process is similar to the previous experiment, the kernel is traversing throughout the image. However in the process of traversing, instead of finding the value differences and generate the probability distribution, the experimental system will compute how much the skin pixels occupy within the kernel in percentage. The skin pixel is based on the value retrieved from Hue value in the HSI colour space model, which values are extracted from the image.

After traversing the image, the system will find out which kernel region has less portion of skin pixels. In this experiment, the system sets the limit that there is less than 10%



of pixels in kernel is occupied by skin pixels. This means that if the skin pixels occupy the kernel more than 10% of pixels in kernel, that kernel region will not be selected as the location for placing the logo in image. There will be many possible locations for the logo to be tagged in image after the traversing and calculation. Therefore the system will evaluate each possible location and then choose the best location for tagging the image. The expected results for the sample image is shown in Figure 3-10.

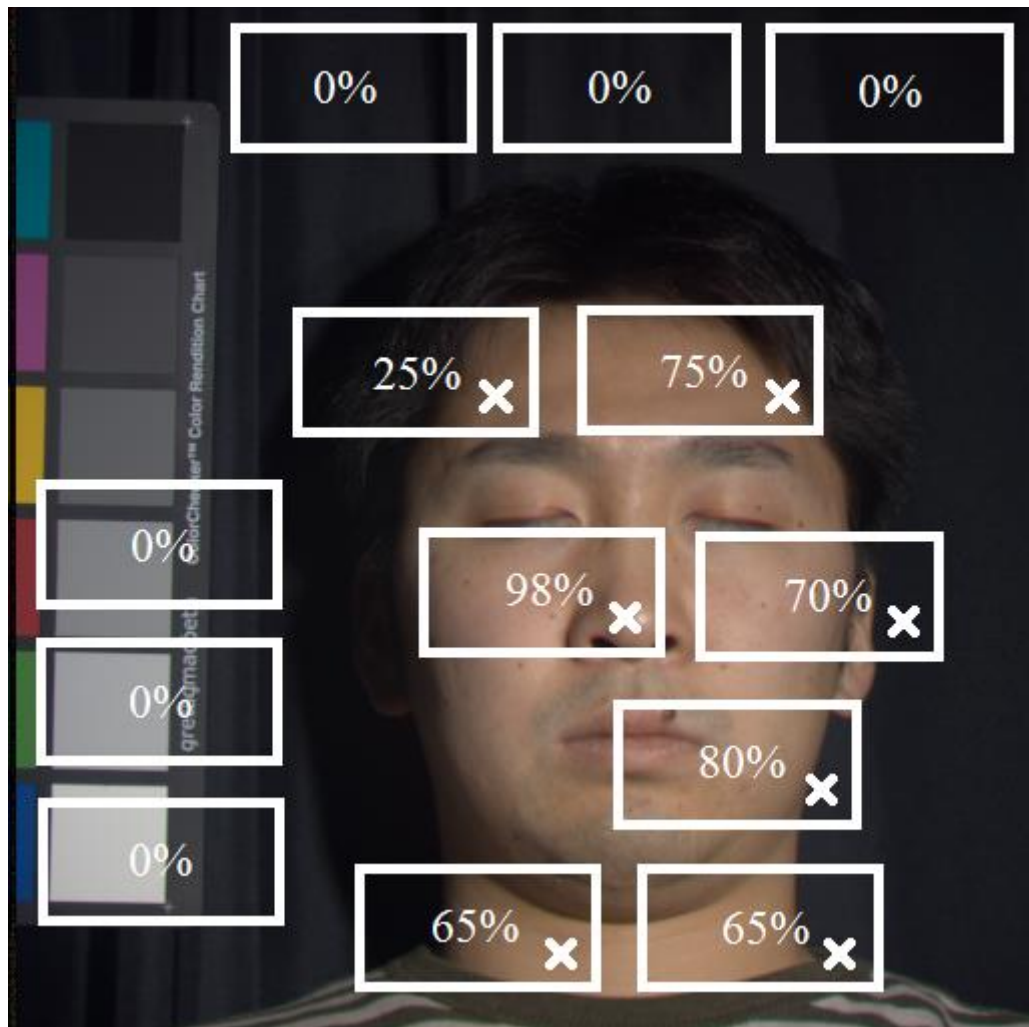


Figure 3-9: The percentage of pixels in kernel that is occupied by skin pixels  
(Columbia University, n.d.).

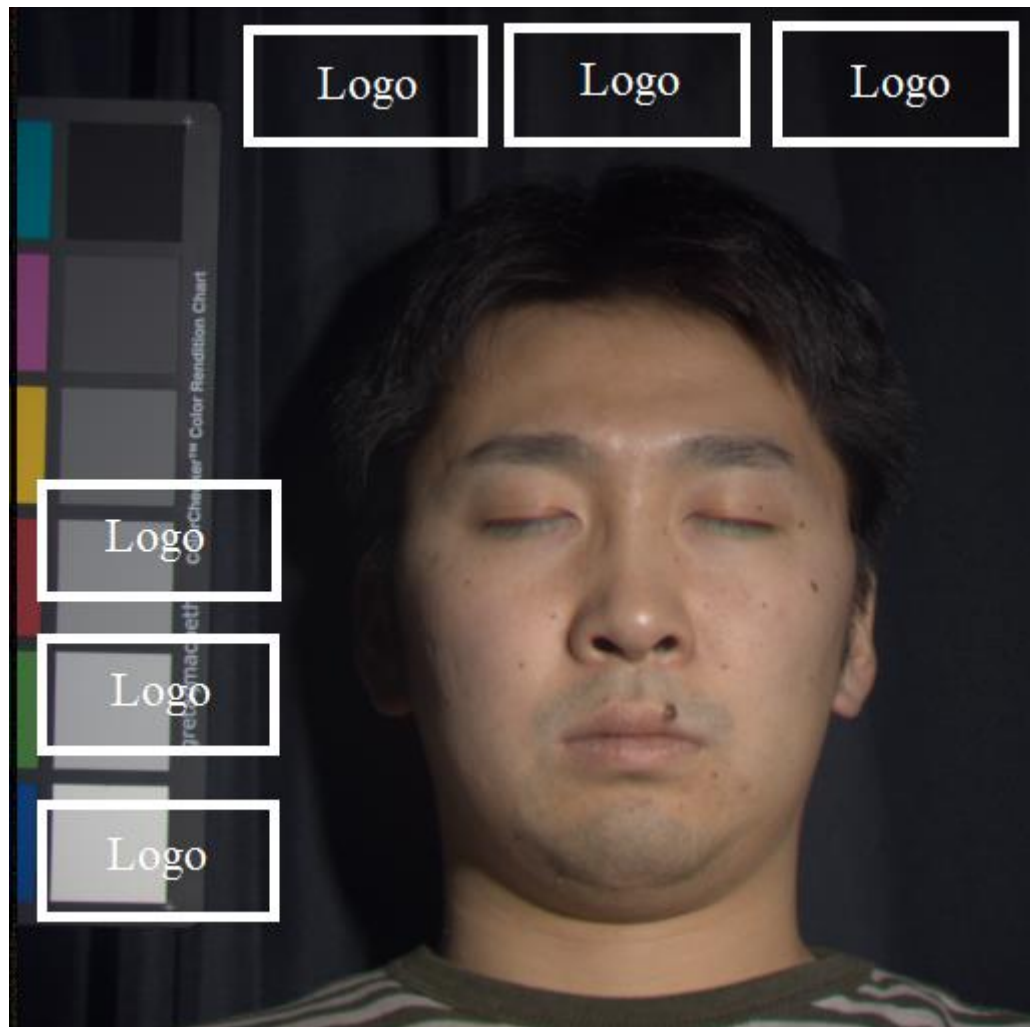


Figure 3-10: Expected result that shows the possible location for tagging the logo in image (Columbia University, n.d.).

### 3.4 Actual System and Algorithms Walkthroughs

In this section there are some explanations on how the actual system and algorithms works.

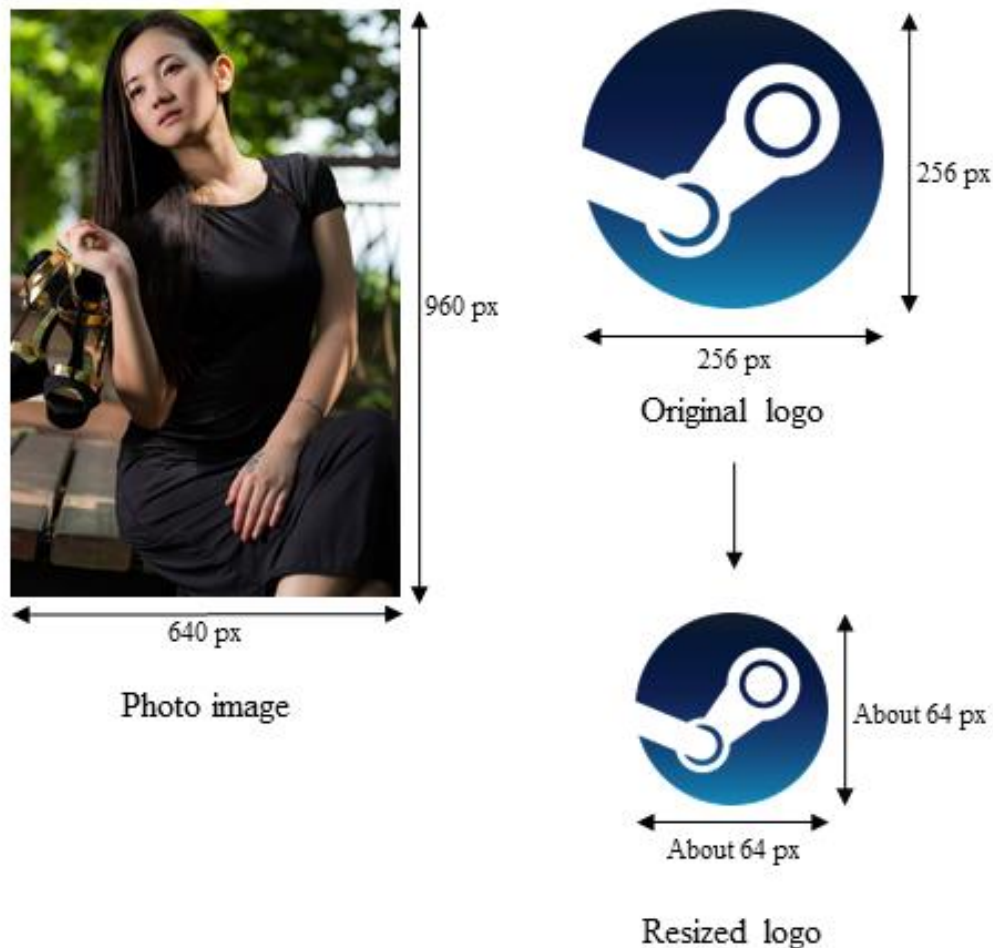


Figure 3-11: Logo is resized based on original photo.

In the beginning, the system will retrieve the photo image and logo from certain file directory and start to determine the width and height of both images. The width and height of logo will be resized based on the width and height of photo image. The maximum width and height for resized logo is about 10% of the width and height of photo image. If the photo image has the width of 640 px and the height of 960 px, then the maximum width and height that the logo can be resized is about 64 px and 96 px respectively. The reason that the logo needs to be resized is to prevent any undesired

result produced when the logo size is larger and it can distort the photo image when logo is placed or tagged on it. Before performing any computation, the system will resize both photo and logo image into smaller size to reduce the computational workload for the system.

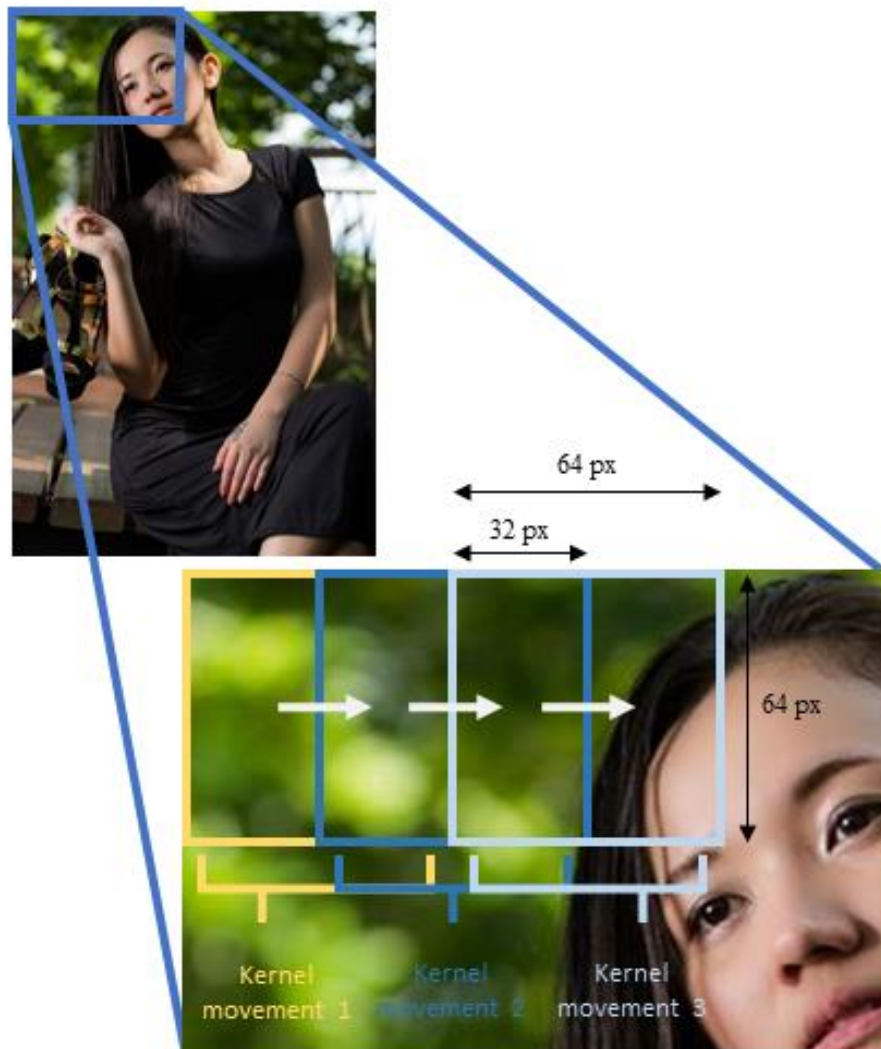


Figure 3-12: Kernel movement in photo image.

After the logo is resized, a kernel is created based on the width and height of resized logo. The use of kernel is to traverse the entire image and retrieve pixel value within the kernel area for further calculation. At first the kernel will be placed at the top left corner, and will be moved from left to right and top to bottom. It is moved with half of kernel width as its moving step, this means that after doing certain computation the

kernel will be moved right by half of its kernel width as shown in Figure 3-12. When it is moved to the right end of photo image, the kernel will be placed at left end of photo image again but with half of its length lower than its original position as shown in Figure 3-13. After it is placed it will move to the right end again, and this process is repeated until the kernel cannot move to anywhere at the bottom right corner of photo image.

Before it is moved to next position, the system will collect each hue value from each pixel within the kernel area and all hue values will be normalized for producing accurate result. The normalized hue values are treated as the independent observations. These observations will be used to calculate the mean and standard deviation for the observations. They are calculated by using the Expectation-Maximization Algorithm in Gaussian Mixture Model (refer to the equations in Chapter 6.1 Gaussian Mixture Model and Expectation-Maximization Algorithm). After the computation is completed in the current position, the mean and standard deviation values for current kernel position are stored and the kernel will be moved to next position and repeats the same process.



Figure 3-13: Kernel will be moved to lower position after reaching right end of photo image.



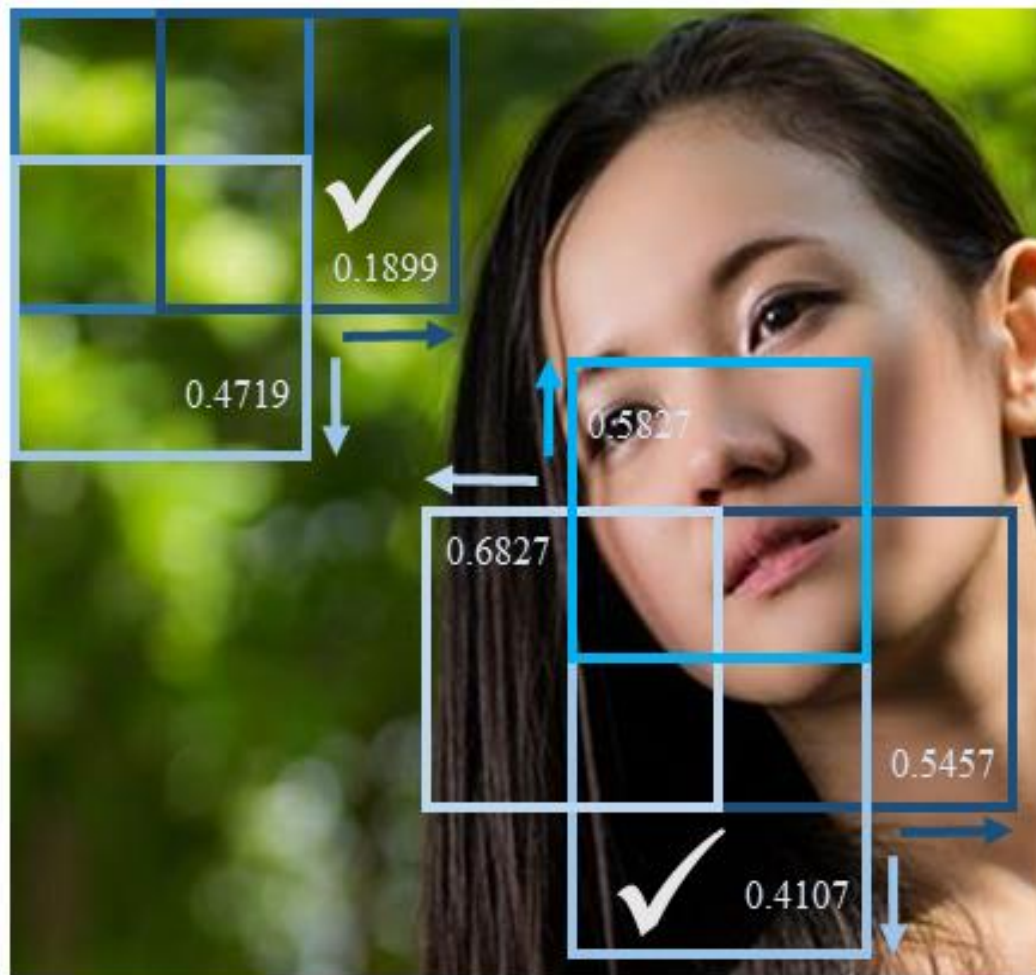


Figure 3-14: System will tag kernel position that has the lowest information loss from current kernel position.

When the computations of mean and standard deviation for each kernel position are completed, the system will traverse the whole photo image again by using same kernel. The kernel positions traversed earlier will be passed through again. This time the system will find out how much the information loss to surroundings from the kernel itself. In each kernel position, the system will calculate the information loss from the current kernel position to kernel's right, bottom, left and top positions as shown in Figure 3-14. The amount of information loss from one position to another position can be calculated by using Kullback-Leibler Divergence Algorithm (refer to the equation in Chapter 6.2 Kullback-Leibler Divergence Algorithm).

Once the information loss is computed to kernel's surroundings, the system will tag the kernel position that has the least information loss from current kernel position. Then

the system will continue calculating information loss and identifying kernel position which has the least information loss in other kernel positions until the kernel reaches to the bottom right corner of photo image. In each kernel position the information of tagged position will be stored by the system for further usage.

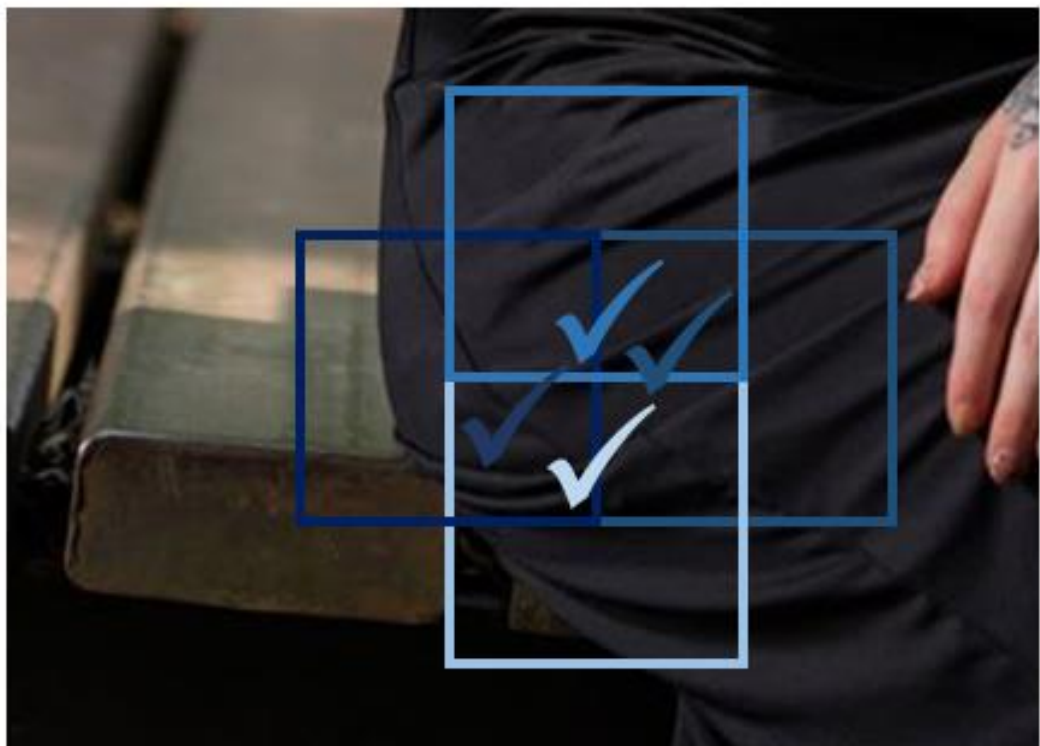


Figure 3-15: Kernel position which has been tagged as having the least information loss by surrounding kernel positions.

## Chapter 3 Methodology

After computing the information loss and identifying the kernel position which has the least information loss in each kernel position, the system will find the suitable locations for placing the logo. The main condition for finding the suitable locations is that in current kernel position, each of adjacent kernel positions are tagging it as having the least information loss. The system will pass through each kernel position again to perform this process. As illustrated in Figure 3-15, there are many kernel positions are tagged by each of their adjacent kernel positions. These tagged positions will be identified and stored by the system as the potential positions for placing the logo as shown in Figure 3-16.

Among the potential positions for placing the logo, the system will inspect each of them to determine whether it is the nearest to the side or corner in photo image. By placing the logo nearer to the side or corner in image it helps the people to easily identify the logo in photo image. The potential position will be ignored by the system from inspecting when it is in the skin region. The skin region can be detected by using three colour spaces, there are RGB colour space, YCbCr colour space and HSV colour space (refer to Chapter 6.3 Determination of Skin Pixel in the image). After inspecting the system will place the logo at the potential position that is the nearest to side or corner in photo image.



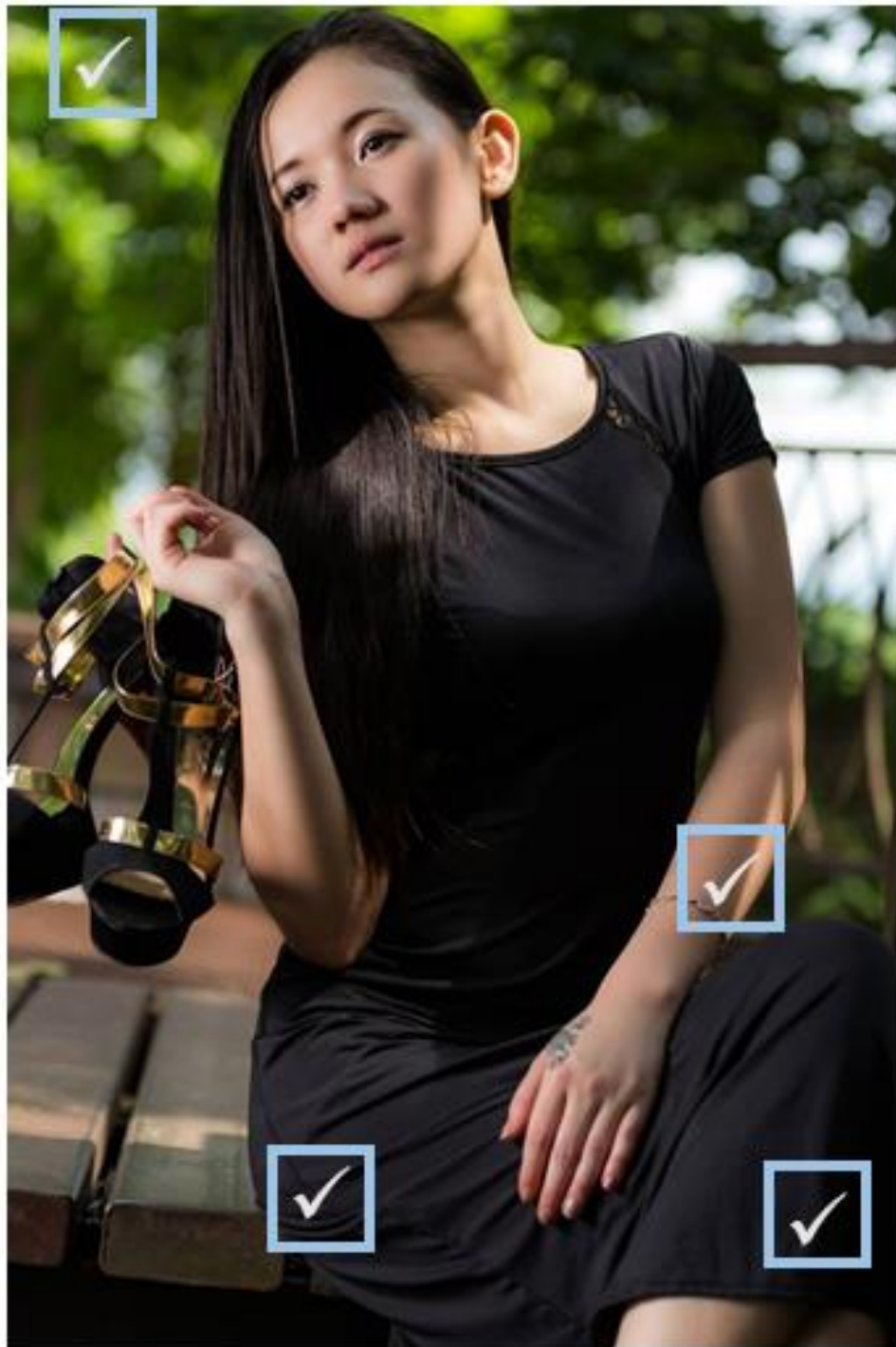


Figure 3-16: Potential positions for logo placement are identified by the system.



Figure 3-17: The final result when the logo is placed at suitable location.



Figure 3-18: Another logo placement in other photo image by the system.



Figure 3-19: Another logo placement in other photo image by the system.

## **Chapter 4 Conclusion**

In a nutshell, Auto-Logo-Tagging System is designed and developed to minimize the problem of tagging logo in images faced by photographers. There are many similar systems exist in the market but they have some weaknesses that cannot help to solve the problem. Therefore this system can serve the users well by automatically tagging the logo in images accurately. In the process of developing the system, there is one biggest concern which can impact greatly to the system. That is whether the Kullback-Leibler divergence can be appropriately used for separating the background and foreground in the image accurately.

Kullback-Leibler divergence is an approach to find how much the information lost from a collection of data (which can be also treated as a set of probability distribution) to another. This can help the system to know which parts in the image have lesser information loss. The parts in image which have lesser information loss is a suitable place for placing the logo as there are no other object can cover up the logo when it is placed. Through this method many major objects can be easily identified by the system. Therefore Kullback-Leibler divergence in which the Auto-Logo-Tagging System uses its information gain algorithm provides the significant amount of contribution to the system developed.

Throughout this project the prototype will be first developed and it includes many experiments mentioned in Chapter 3.3 Experiment Setting. Those experiments are conducted to show that how the Kullback-Leibler information gain algorithm and HSI colour space model work in this system. They are quite significant as they are primarily used to identify the objects in image which is the fundamental function in this system. Therefore the experiments need to be carefully carried out so that the accurate results can be produced.

Other than Kullback-Leibler divergence algorithm, the Expectation-Maximization algorithm in Gaussian Mixture Model is also used in this system. It is mainly used for finding and estimating many sets of mean and standard deviation in a collection of data or observations. It is very useful as it can estimate the unknown parameters that represents the mixing value for the Gaussians, mean and covariance in a set of data (Expectation-maximization algorithm 2015). As shown in Chapter 3.4 Actual System and Algorithms Walkthroughs, the system combines the Expectation-Maximization

## Chapter 4 Conclusion

algorithm and Kullback-Leibler divergence algorithm in order to find out many locations in image that has lesser information loss. By combining both algorithms, the logo can be placed in a location of photo image that can be seen clearly by the people.

### Chapter 5 Reference

- Abdul Rahman, NA, Kit, CW & See, J n.d., 'RGB-H-CbCr Skin Colour Model for Human Face Detection', *Faculty of Information Technology, Multimedia University*. Available from: <<http://academic.aua.am/Skhachat/Public/Papers%20on%20Face%20Detection/RGB-H-CbCr%20Skin%20Colour%20Mode%20for%20Human%20Face%20Detection.pdf>>. [24 August 2015].
- Chuong, BD & Batzoglou, S 2008, 'What is the expectation maximization algorithm?', *Nature Biotechnology*, vol. 26, no. 8, pp. 897-899. Available from: Nature Biotechnology. [20 August 2015].
- Columbia University n.d., *Multispectral Image Database*. Available from: <<http://www1.cs.columbia.edu/CAVE/databases/multispectral/>>. [27 February 2015].
- Garcia, JA, Fdez-Valdivia, J, Rodriguez-S´anchez, R & Fdez-Vidal, XR 2002, 'Performance of the Kullback-Leibler information gain for predicting image fidelity', *Proceedings of 16th International Conference*, vol. 3, pp. 843-848. Available from: IEEE Xplore Digital Library [9 November 2014].
- Garcia, JA, Fdez-Valdivia, J, Rodriguez-S´anchez, R, Fdez-Vidal, XR & Fuertes, JM 2001, 'Minimum error gain for predicting visual target distinctness', *Society of Photo-Optical Instrumentation Engineers*, vol. 40, pp. 1794-1817. Available from: ResearchGate [9 November 2014].
- Grasemann, U & Miikkulainen, R 2005, 'Effective Image Compression using Evolved Wavelets', *Proceedings of the Genetic and Evolutionary Computation Conference 2005*, pp. 1961-1968. Available from: <<http://www.genetic-programming.org/hc2005/f472-grasemann.pdf>>. [26 December 2014].
- Kautsky, J, Zitova, B, Flusser, J & Peters, G 1998, 'Feature point detection in blurred images', *Image and Vision Computing, International Conference*. Available from: <[http://www.ini.rub.de/PEOPLE/rolf/sb-articles/IVCNZ\\_MIT\\_KOPF.pdf](http://www.ini.rub.de/PEOPLE/rolf/sb-articles/IVCNZ_MIT_KOPF.pdf)>. [25 December 2014].
- Lakshmi, HCV & PatilKulakarni, S 2010, 'Segmentation Algorithm for Multiple Face Detection in Color Images with Skin Tone Regions using Color Spaces and Edge Detection Techniques', *International Journal of Computer Theory and*

## Chapter 5 Reference

- Engineering*, vol. 2, pp. 552-558. Available from: International Journal of Computer Theory and Engineering [12 November 2014].
- MathWorks n.d., *Wavelet Transform in MATLAB*. Available from: <<http://www.mathworks.com/discovery/wavelet-transforms.html>>. [23 December 2014].
- Maraq, M, Al-Zboun, F, Dhyabat, M & Abu Zitar, R 2012, 'Recognition of Arabic Sign Language (ArSL) Using Recurrent Neural Networks', *Journal of Intelligent Learning Systems and Applications*, vol. 4, no. 1, pp. 41-52. Available from: <[http://file.scirp.org/Html/4-9601063\\_17553.htm](http://file.scirp.org/Html/4-9601063_17553.htm)>. [28 February 2015].
- Martins, TG 2013, Kullback-Leibler divergence. 10 July 2013. *Thiago G. Martins*. Available from: <<https://tgmstat.wordpress.com/2013/07/10/kullback-leibler-divergence/>>. [21 August 2015].
- Nain, N, Laxmi, V & Bhadviya, B 2008, 'Feature Point Detection for Real Time Applications', *Proceedings of the World Congress on Engineering 2008*, vol. I. Available from: <[http://www.iaeng.org/publication/WCE2008/WCE2008\\_pp643-648.pdf](http://www.iaeng.org/publication/WCE2008/WCE2008_pp643-648.pdf)>. [13 December 2014].
- Ray, R 2012, Using Photoshop CS6 to frame your pictures. 4 December 2012. *Russel Ray Photos*. Available from: <<http://russelrayphotos2.com/2012/12/04/using-photoshop-cs6-to-frame-your-pictures/>>. [17 November 2014].
- Resize & Watermark Multiple Images Automatically in Photoshop CS6, 2013 (video file), Available from: <<https://www.youtube.com/watch?v=E3C7LE0H4-o>>. [17 November 2014].
- Ruchika, Singh, M & Singh, AR 2012, 'Compression of Medical Images Using Wavelet Transforms', *International Journal of Soft Computing and Engineering*, vol. 2, no. 2, pp. 339-343. Available from: <<http://www.ijscce.org/attachments/File/v2i2/B0614042212.pdf>>. [24 December 2014].
- scikit-learn n.d., *Gaussian mixture model*. Available from: <<http://scikit-learn.org/stable/modules/mixture.html>>. [20 August 2015].



## Chapter 5 Reference

- Singh, SK, Chauhan, DS, Vatsa, M & Singh, R 2003, 'A Robust Skin Color Based Face Detection Algorithm', *Tamkang Journal of Science and Engineering*, vol. 6, pp. 277-234. Available from: Tamkang University [10 November 2014].
- Smith, S 2012, How to Add your Logo or Text to a Photo using Photoshop. 16 January 2012. *The House of Smiths*. Available from: <<http://www.thehouseofsmiths.com/2012/01/how-to-add-your-logo-or-text-to-photo.html>>. [17 November 2014].
- Zangana, HM & Al-Shaikhli, IF 2013, 'A New Algorithm for Human Face Detection Using Skin Color Tone', *IOSR Journal of Computer Engineering*, vol. 11, pp. 31-38. Available from: ResearchGate [11 November 2014].
- 'Wikipedia', *Expectation-maximization algorithm*, (wiki article), August 11, 2015. Available from: <[https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization\\_algorithm](https://en.wikipedia.org/wiki/Expectation%E2%80%93maximization_algorithm)>. [20 August 2015].
- 'Wikipedia', *Feature Detection (computer vision)*, (wiki article), December 18, 2014. Available from: <[http://en.wikipedia.org/wiki/Feature\\_detection\\_%28computer\\_vision%29](http://en.wikipedia.org/wiki/Feature_detection_%28computer_vision%29)>. [20 December 2014].
- 'Wikipedia', *Wavelet transform*, (wiki article), December 8, 2014. Available from: <[http://en.wikipedia.org/wiki/Wavelet\\_transform](http://en.wikipedia.org/wiki/Wavelet_transform)>. [23 December 2014].



**Appendix A Gaussian Mixture Model and Expectation-Maximization Algorithm**

(In this section, all the information is from Expectation-maximization algorithm 2015)

**A-1 Gaussian Mixture Model**

Let  $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n)$  be a sample of  $n$  independent observations from a mixture of two multivariate normal distributions of dimension  $d$  and let  $\mathbf{z} = (z_1, z_2, \dots, z_n)$  be the latent variables that determine the component from which the observation originates.

$$X_i | (Z_i = 1) \sim \mathcal{N}_d(\boldsymbol{\mu}_1, \Sigma_1)$$

and

$$X_i | (Z_i = 2) \sim \mathcal{N}_d(\boldsymbol{\mu}_2, \Sigma_2)$$

where

$$P(Z_i = 1) = \tau_1$$

and

$$P(Z_i = 2) = \tau_2 = 1 - \tau_1$$

The aim is to estimate the unknown parameters representing the "mixing" value between the Gaussians and the means and covariance of each:

$$\boldsymbol{\theta} = (\boldsymbol{\tau}, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \Sigma_1, \Sigma_2)$$

where the incomplete-data likelihood function is

$$L(\boldsymbol{\theta}; \mathbf{x}) = \prod_{i=1}^n \sum_{j=1}^2 \tau_j f(\mathbf{x}_i; \boldsymbol{\mu}_j, \Sigma_j)$$

and the complete-data likelihood function is

$$L(\boldsymbol{\theta}; \mathbf{x}, \mathbf{z}) = P(\mathbf{x}, \mathbf{z} | \boldsymbol{\theta}) = \prod_{i=1}^n \sum_{j=1}^2 \mathbb{I}(z_i = j) f(\mathbf{x}_i; \boldsymbol{\mu}_j, \Sigma_j) \tau_j$$

or

## Appendix

$$L(\theta; \mathbf{x}, \mathbf{z}) = \exp \left\{ \sum_{i=1}^n \sum_{j=1}^2 \mathbb{I}(z_i = j) \left[ \log \tau_j - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_j)^\top \Sigma_j^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j) - \frac{d}{2} \log(2\pi) \right] \right\}.$$

where  $\mathbb{I}$  is an indicator function and  $f$  is the probability density function of a multivariate normal.

**A-2 Expectation (E Step)**

Given the current estimate of the parameters  $\theta^{(t)}$ , the conditional distribution of the  $Z_i$  is determined by Bayes theorem to be the proportional height of the normal density weighted by  $\tau$ .

$$T_{j,i}^{(t)} := P(Z_i = j | X_i = \mathbf{x}_i; \theta^{(t)}) = \frac{\tau_j^{(t)} f(\mathbf{x}_i; \boldsymbol{\mu}_j^{(t)}, \Sigma_j^{(t)})}{\tau_1^{(t)} f(\mathbf{x}_i; \boldsymbol{\mu}_1^{(t)}, \Sigma_1^{(t)}) + \tau_2^{(t)} f(\mathbf{x}_i; \boldsymbol{\mu}_2^{(t)}, \Sigma_2^{(t)})}$$

These are called the "membership probabilities" which are normally considered the output of the E step (although this is not the Q function of below).

Note that this E step corresponds with the following function for Q:

$$\begin{aligned} Q(\theta | \theta^{(t)}) &= E[\log L(\theta; \mathbf{x}, \mathbf{Z})] \\ &= E[\log \prod_{i=1}^n L(\theta; \mathbf{x}_i, \mathbf{z}_i)] \\ &= E[\sum_{i=1}^n \log L(\theta; \mathbf{x}_i, \mathbf{z}_i)] \\ &= \sum_{i=1}^n E[\log L(\theta; \mathbf{x}_i, \mathbf{z}_i)] \\ &= \sum_{i=1}^n \sum_{j=1}^2 T_{j,i}^{(t)} [\log \tau_j - \frac{1}{2} \log |\Sigma_j| - \frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_j)^\top \Sigma_j^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j) - \frac{d}{2} \log(2\pi)] \end{aligned}$$

This does not need to be calculated, because in the M step we only require the terms depending on  $\tau$  when we maximize for  $\tau$ , or only the terms depending on  $\mu$  if we maximize for  $\mu$ .

**A-3 Maximization (M Step)**

The fact that  $Q(\theta|\theta(t))$  is quadratic in form means that determining the maximizing values of  $\theta$  is relatively straightforward. Note that  $\tau$ ,  $(\mu_1, \Sigma_1)$  and  $(\mu_2, \Sigma_2)$  may all be maximized independently since they all appear in separate linear terms.

To begin, consider  $\tau$ , which has the constraint  $\tau_1 + \tau_2 = 1$ :

$$\tau_j^{(t+1)} = \frac{\sum_{i=1}^n T_{j,i}^{(t)}}{\sum_{i=1}^n (T_{1,i}^{(t)} + T_{2,i}^{(t)})} = \frac{1}{n} \sum_{i=1}^n T_{j,i}^{(t)}$$

For the next estimates of  $(\mu_1, \sigma_1)$ :

$$\boldsymbol{\mu}_1^{(t+1)} = \frac{\sum_{i=1}^n T_{1,i}^{(t)} \mathbf{x}_i}{\sum_{i=1}^n T_{1,i}^{(t)}}$$

and

$$\Sigma_1^{(t+1)} = \frac{\sum_{i=1}^n T_{1,i}^{(t)} (\mathbf{x}_i - \boldsymbol{\mu}_1^{(t+1)}) (\mathbf{x}_i - \boldsymbol{\mu}_1^{(t+1)})^\top}{\sum_{i=1}^n T_{1,i}^{(t)}}$$

and, by symmetry

$$\boldsymbol{\mu}_2^{(t+1)} = \frac{\sum_{i=1}^n T_{2,i}^{(t)} \mathbf{x}_i}{\sum_{i=1}^n T_{2,i}^{(t)}}$$

and

$$\Sigma_2^{(t+1)} = \frac{\sum_{i=1}^n T_{2,i}^{(t)} (\mathbf{x}_i - \boldsymbol{\mu}_2^{(t+1)}) (\mathbf{x}_i - \boldsymbol{\mu}_2^{(t+1)})^\top}{\sum_{i=1}^n T_{2,i}^{(t)}}$$

**A-4 Termination**

Conclude the iterative process if  $\log L(\theta^t; \mathbf{x}, \mathbf{Z}) \leq \log L(\theta^{(t-1)}; \mathbf{x}, \mathbf{Z}) + \epsilon$  for  $\epsilon$  below some pre-set threshold.

### A-5 Kullback-Leibler Divergence Algorithm

(In this section, all the information is from Expectation-maximization algorithm 2015)

In Univariate Gaussian distributions, the Kullback-Leibler divergence between a Gaussian distribution  $p$  with mean  $\mu_1$  and variance  $\sigma_1^2$  and a Gaussian distribution  $q$  with mean  $\mu_2$  and variance  $\sigma_2^2$  is given by

$$\text{KL}(p, q) = \log \frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1 - \mu_2)^2}{2\sigma_2^2} - \frac{1}{2}$$

The equation above shows the transmission value from Gaussian distribution  $p$  to Gaussian distribution  $q$ . In order to show how much the information that will be lost from Gaussian distribution  $p$  to Gaussian distribution  $q$ , let

$$x = \text{KL}(p, q)$$

and

$$y = \text{KL}(q, p),$$

therefore,

Amount of information lost from Gaussian distribution  $p$  to distribution  $q$

$$= \frac{x}{x + y}$$

## Appendix B Determination of Skin Pixel in the Image

(In this section, all the information is from Abdul Rahman, Kit & See n.d., ‘RGB-H-CbCr Skin Colour Model for Human Face Detection’)

In order to determine the skin pixel in an image, there is a set of bounding rules derived from three colour spaces, there are RGB, YCbCr and HSV, which rules are needed to be met so that the system can accurately determine the skin region.

### B-1 RGB Colour Space

In RGB colour space, the skin colour at uniform daylight illumination rule is defined as:

$$\begin{aligned} & (R > 95) \text{ AND } (G > 40) \text{ AND } (B > 20) \text{ AND} \\ & (\max\{R, G, B\} - \min\{R, G, B\} > 15) \text{ AND} \\ & (|R - G| > 15) \text{ AND } (R > G) \text{ AND } (R > B) \end{aligned} \quad \text{[Equation 1]}$$

While the skin colour under flashlight or daylight lateral illumination rule is given by:

$$\begin{aligned} & (R > 220) \text{ AND } (G > 210) \text{ AND } (B > 170) \text{ AND} \\ & (|R - G| \leq 15) \text{ AND } (R > B) \text{ AND } (G > B) \end{aligned} \quad \text{[Equation 2]}$$

To consider both conditions when needed, a logical OR will be used to combine both Equation 1 and Equation 2. The RGB bounding rule is denoted as:

$$\text{Rule A: Equation 1} \cup \text{Equation 2}$$

## **B-2 Cb-Cr Subspace from YCbCr Colour Space**

Cb-Cr subspace is a strong discriminant of skin colour, there are 5 bounding planes from its 2-D subspace distribution. The five bounding rules that enclosed the Cb-Cr skin colour region are formulated as below:

$$Cr \leq 1.5862 \times Cb + 20 \quad \text{[Equation 3]}$$

$$Cr \geq 0.3448 \times Cb + 76.2069 \quad \text{[Equation 4]}$$

$$Cr \geq -4.5652 \times Cb + 234.5652 \quad \text{[Equation 5]}$$

$$Cr \leq -1.15 \times Cb + 301.75 \quad \text{[Equation 6]}$$

$$Cr \leq -2.2857 \times Cb + 432.85 \quad \text{[Equation 7]}$$

The Equation 3 to Equation 7 are combined by using logical AND to get the CbCr bounding rule, denoted as:

$$\text{Rule B: Equation 3} \cap \text{Equation 4} \cap \text{Equation 5} \cap \text{Equation 6} \cap \text{Equation 7}$$

### **B-3 HSV Colour Space**

In HSV colour space, the hue value exhibit the most noticeable separation between skin and non-skin regions. Therefore there are two cut-off levels as H subspace skin boundaries.

$$H < 25 \quad \text{[Equation 8]}$$

$$H > 230 \quad \text{[Equation 9]}$$

Both Equation 8 and Equation 9 are combined by using a logical OR to obtain the H bounding rule, denoted as:

$$\text{Rule C: Equation 8} \cup \text{Equation 9}$$

Thereafter, each pixel that fulfils Rule A, Rule B and Rule C is classified as a skin colour pixel. So:

$$\text{Rule A} \cap \text{Rule B} \cap \text{Rule C}$$