Tutorial: Introduction to Emotion Recognition for Digital Images

Vinay Kumar, Arpit Agarwal, Kanika Mittal

Tutorial: Introduction to Emotion Recognition for Digital Images

By

Vinay Kumar

Arpit Agarwal

Kanika Mittal

Department of Electronics and Communication Engineering

Jaypee University of Information Technology
Waknaghat, Solan - 173 234, Himachal Pradesh

Table of Contents

ABSTRACT

Humans can use vision to identify objects quickly and accurately. Computer Vision seeks to emulate human vision by analyzing digital image inputs. For humans to detect an emotion will not be a difficult job to perform as humans are linked with emotions themselves but for a computer detecting an emotion will be difficult job to perform. Detecting emotion through voice, for example: detecting 'stress' in a voice by setting parameters in areas like tone, pitch, pace, volume etc can be achieved but in case of digital images detecting emotion just by analyzing images is a novel way.

The algorithm we proposed first detects facial regions in the image using a skin color model using RGB and HSV color space. Then lip region is extracted from the face region using the lip color model YCrCb color space. All the above color space uses a definite threshold value to differentiate between the regions of interest. Finally after the extraction of lip region from the image, it is compared with the series of templates and on the basis of best correlated template emotion is recognized. The proposed method is simple and fast compared to neural analysis of facial region as a whole. A simple pre defined database will be needed to help detecting various emotions that can be recognized using lip region. Size of database will affect the effectiveness of the proposed algorithm.

CHAPTER 1
INTRODUCTION

Human vision can experience emotion as associated with mood, temperament, personality and disposition. Computer Vision seeks to emulate the human vision by analyzing digital image as input. The fact that world is three- dimensional while computer vision is two-dimensional is basically one of the main problems that complicate Computer Vision.

1.1 Emotion Detection Problem

Everyday almost everyone in this world interact with other in one or another way either directly (for e.g. face to face) or indirectly (for e.g. phone calls). In some profession interaction with people are the main deed to perform like call centers, sale executives etc. With great advancement in technology in terms of different techniques of people interacting with each other it is quite necessary that one should be aware of current emotions of the person he/she is interacting. With the advancement of 3G technology in mobile communication field one may be capable to interact face to face with other while talking so if one is aware of mood of other in advance that interaction will certainly result in social as well as professional benefits.

Firstly, what are emotions? A mental state that arises spontaneously rather than through conscious effort and is often accompanied by physiological changes and these physiological changes are recognized from outer world.

Secondly, major emotions that humans face in day to day life :

- Happiness
- Worry
- Grief
- Anger
- Surprise
- Fear
- Boredom and many more.

Thirdly, as a Scientific view: Emotion lies behind much of the richness of human life and is one of the main drivers behind our choices and decisions. In order to avoid attributing hard categories, many researchers prefer to use a continuous space such as that in Figure 1 rather than discrete labels to describe emotion.

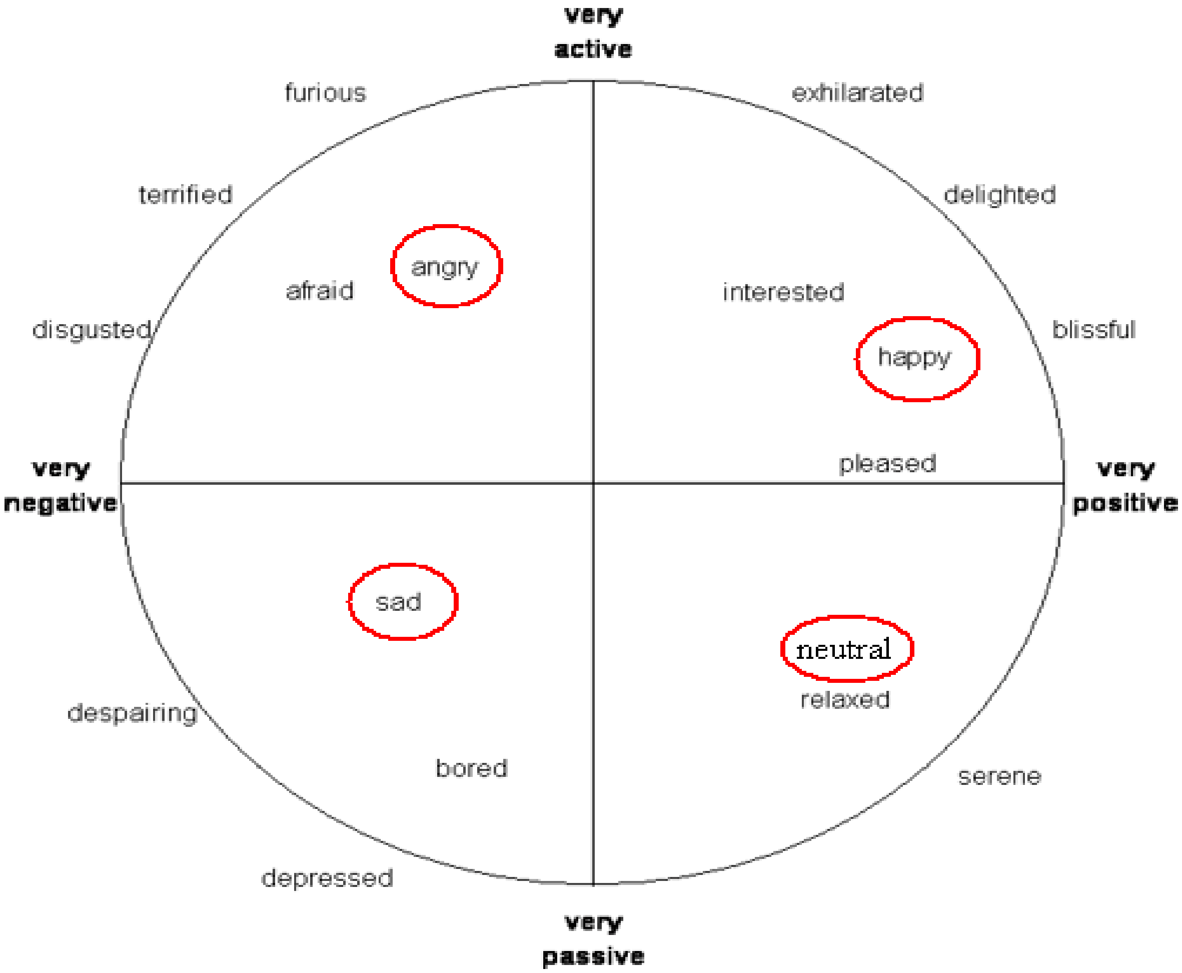


Figure 1: A two - dimensional representation of emotion.

The advantage of this representation is that it is possible to express as numbers the continuous scale from “mildly irritated” to “incandescent with rage” and also to capture the shades of grey between related pairs of emotions.

One major way to detect emotions is by analyzing the voice by setting parameters in areas like tone, pitch, pace, volume etc. but this is a complex algorithm to work out and takes a lot of computing time as well as cost. This technique demands a Interactive Voice Response (IVR) systems which help only to detect emotions while on talk.

If any novel way is generated that helps to detect emotion before a user interacts, will certainly help. In this project we will elaborate the methods by which user identifies the emotion without interacting.

1.2 Existing methods of Emotion Detection

Solutions are available but not to the extent that will benefit common people. Emotion recognition so far achieved, is done by using two major scientific fields

- Using Interactive Voice Response (IVR) system using audio signal analysis: Emotion recognition solutions depend on which emotions we want a machine to recognize and for what purpose. Emotion recognition has applications in talking toys, video and computer games, and call centers. Particularly interested in the application of emotion recognition technologies for Interactive Voice Response (IVR) systems with specific application to call centers. An obvious example is the automatic call routing of angry customer to agents (customer representative) and the automatic quality monitoring of agents performance. These systems are conversational and hence utterances are usually short.

  This technique though effective but has a flaw that some customers are very cool, calm and collected when conversing although deep down they may be angry. They may be able to clearly outline within the call what will happen if their problem is not rectified and these calls may go undetected when applying emotion detection using IVR.

  Another drawback of using above stated method is that when looking at the frequency range of the human voice, it is generally understood that electronic telephony can only handle around 20 percent of what is said.

  Thus the system using (IVR) for emotion recognition is not reliable.
- Emotion Recognition using the Brain Activity: Recognizing emotion from human brain activity, measured by Electroencephalography (EEG) signals proposes a system to analyze EEG signals and classify them into 5 classes on two emotional dimensions, valence and arousal. This system was designed using prior knowledge from other research, and is meant to assess the quality of emotion recognition using EEG signals in practice. In order to perform this assessment, a dataset with EEG signals is used. This is

done by measuring EEG signals from people that were emotionally stimulated by pictures. This method enabled to teach our system the relationship between the characteristics of the brain activity and the emotion. The EEG signals contain enough information to separate five different classes on both the valence and arousal dimension. However, using a 3-fold cross validation method for training and testing, we reached classification rates of 32% for recognizing the valence dimension from EEG signals and 37% for the arousal dimension. Much better classification rates were achieved when using only the extreme values on both dimensions, the rates were 71% and 81%.

Comparing the above two stated method no doubt the second method using Electroencephalography i.e. EEG is better but is out of reach of common man. EEG is far expensive to be used in common practice and needs a lot of computational expertise. Therefore an easy and reliable solution to detect emotions is needed that is done in this project to detect emotions using facial expressions using digital images which are easy to acquire and analyze.

Industrial use of our approach will also be beneficial for common public in a way that with the advancement of 3G technology users will interact with each other just by watching each other in their respective cell phones and even their cell phones will be able to recognize the emotions prior to interaction. This technology will also help call centers to monitor customer as well as employee emotions and thus in better operation. Even lie detectors can be upgraded using all the above three methods thus better results could be expected. In the industry of digital cameras methods of detecting smiling faces can be elaborated further.

1.3 Project Aim

- The aim of this project is to use computer vision techniques to automatically detect and analyze the emotions from the digital images.

- To develop a system that is easy to use, can be easily adaptable, modified, reproduced, and even improved.

- The ultimate goal is to develop a widely acceptable piece of software that will work for the benefit of masses. The software designed will avoid user interaction. The user will run the picture in the software and the software which is fully automated will result in the emotion detection.

1.4 Overview

The image will be systematically broken down and analyzed by the series of algorithms to determine the pixels that represent facial region. After this a second algorithm is applied to first crop the facial region and then next algorithm will detect lips from facial region. The automatic algorithm must correctly identify all pixels correctly included in lips while not incorrectly classifying the other regions as lips or lip colored coat.

Use of emotion recognition from digital images has a large opportunity and upcoming market. This is the primary reason to adopt a general and easy to apply approach towards the entire process. The approach is based on the assumption that there are not multiple faces in the image.

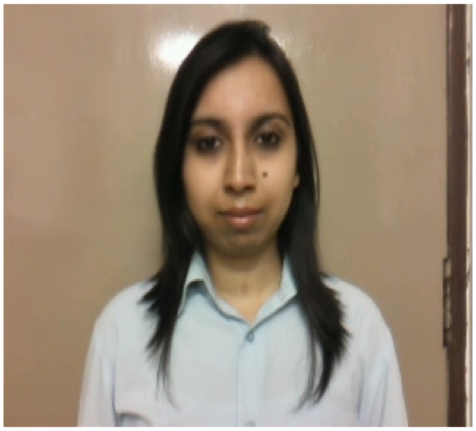
1.5 Motivation

The algorithms used in this project are very general in form. The idea behind this is to allow the system to be sensitive enough to detect the instances of facial regions as well as lip region which can occur in background of image. In such cases faces will be detected and that will result in chances where several false detections will be done. This will result from any lip colored marks in the image.

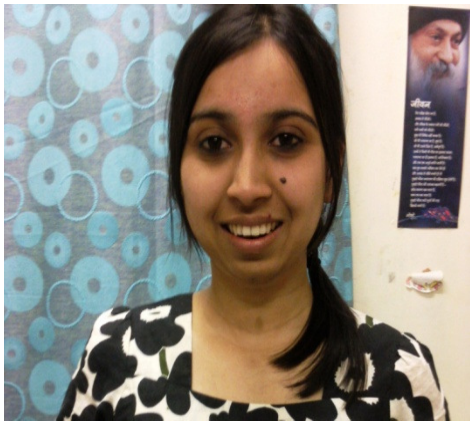
1.6 Emotions considered in this Project

In this project we are considering five major emotions which are mainly centering toward lips in facial region. These emotions are:

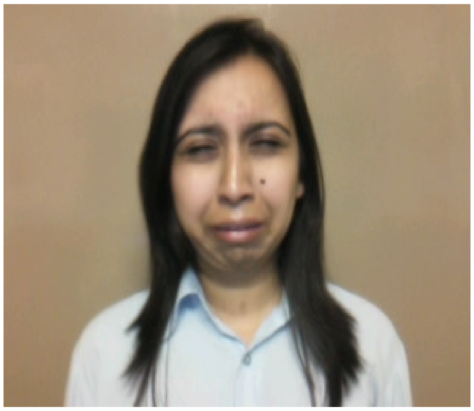
Emotion 1: NEUTRAL



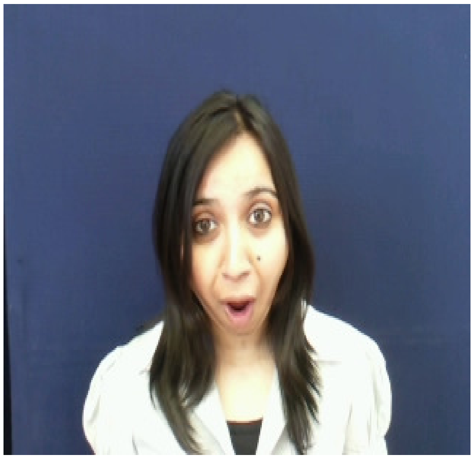
Emotion 2: HAPPINESS



Emotion 3: GRIEF



Emotion 4: SURPRIZED



Emotion 5: ANGER

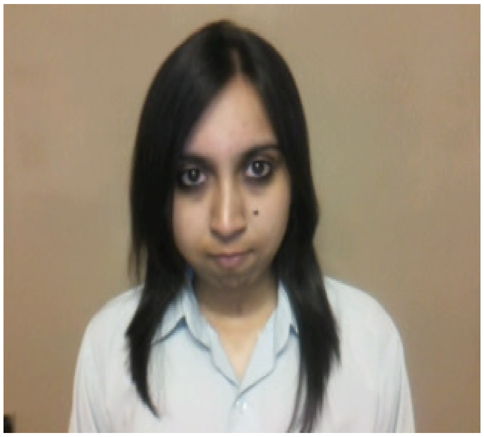


Figure 2: List of emotions used in this project

CHAPTER 2
AUTOMATING THE PROCESS

In order to implement a system that can automatically detect emotion, a series of logical steps have to be developed. The system was divided into 5 main stages/steps. These are as follows:-

1. Face Detection.
2. Cropping Facial Region.
3. Lip Region Detection.
4. Lip Segmentation.
5. Template Database Generation.
6. Comparison with Templates.

2.1 Sequence of Events

1. Image opened or read.
2. Apply possible point operations on image for better results.
3. Face detection algorithm applied to identify facial region.
4. Facial region is cropped.
5. Discrete Hartley Transform is applied on cropped image.
6. Lip detection algorithm is applied.
7. Lip region is segmented.
8. Separate database of emotion templates is generated.
9. Cropped lip area is compared individually by templates to identify exact emotion.

Note: We are considering images with single faces only and with relative dark background

CHAPTER 3
FACE DETECTION

We now give a definition of face detection: Given an arbitrary image, the goal of face detection is to determine whether or not there are many faces in the image and if present, return the image location and extent of each face.

The challenges associated with face detection can be attributed to the following factors:

- Pose: The images of a face vary due to the relative camera-face pose (frontal, 45 degree, profile, upside down) and some facial features such as an eye or the nose may become partially or wholly occluded.
- Presence or absence of structural components: Facial features such as beards, mustaches and glasses may or may not be present and there is a great deal of variability among these components including color, shape and size.
- Facial expression: The appearances of face are directly affected by a person's facial expression.
- Occlusion: Faces may be partially occluded by other objects. In an image with a group of people, some faces may occlude other faces.
- Image orientation: Face images directly vary for different rotations about the camera's optical axis.
- Imaging conditions: When the image is formed , factors such as lighting (spectra, source distribution and intensity) and camera characteristics (sensor, response, lenses) affect the appearance of a face.

There are many closely related problems of face detection. Face localization aims to determine the image position of a single face; this is a simplified detection problem with the assumption that an input image contains only one face.

Given a single face images, the main concern of face detection is to identify all image regions which contain a face regardless of its orientation, background and lighting conditions. Such task is tricky since faces can have a vast assortment in terms of shape, color, size or texture. At present time a lot of automatic approaches involve detecting faces in an image and subsequently, detecting lips within each detected face. But most of the face detection algorithms are only able to detect faces that are oriented in upright frontal view; these approaches cannot detect faces that are rotated in-plane or out-of plane with respect to the image plane, also cannot detect faces in case when only part of face is visible. As a result using threshold to separate skin region from an image for face detection was chosen in this algorithm.

3.1 Skin Color Classification

While different ethnic groups have different levels of melanin and pigmentation, the range of colors that human facial skin takes on is clearly a subspace of the total color space. With the assumption of a typical photographic scenario, it would be clearly wise to take advantage of face color correlations to limit our face search to areas of an input image that have at least the correct color components. In pursuing this goal, we looked at three color spaces that have been reported to be useful in the literature, RGB and HSI spaces, as well as YCrCb. Below we will briefly describe each color space and how it is used for skin color classification.

3.2 Skin Based Segmentation

Human skin color is a very efficient feature for face detection. Although different people may have different skin color, several studies show has shown that the major difference lies largely between their intensity rather than their chrominance.

There has been much research entailing skin based segmentation, which incidentally is the basis for some face detecting algorithms. Most of the algorithms use a range of color values to define skin color.

3.3 Segmentation Rules

Segmenting an image is the process by which a computer attempts to separate objects within an image from the background as well as from other objects. The segmentation rules are the rules that will determine the formation of regions. The segmentation rules are based on analyzing the color and edge properties of a region.

The advantages of skin based segmentation is the loss of the sizable overhead of first determining faces within an image coupled with the fact that even faces are occluded, profiled and partially out of frame or simply not recognized, will all be further processed.

This project bases its skin color using the RGB, HSI/HSV and YCrCb color models. The values are arbitrary, unique to this project and obtained by extensive color sampling from flesh tone colors from digital photos.

Using the HSI model accounted for a greater range of skin tones. The problem however was the HSI model made it difficult to constrain the colors to specifically the flesh-like tones. The solution adopted was to use the HSI model to allow for a large range and the RGB values to constrain over dominance of specific colors e.g. too ‘red’. Skin color seems to occupy small range of hue values.

3.4 Color Models

3.4.1 RGB

The RGB color model is the most widely recognized color model. It comprises of three components namely the red, green and blue color channels. The RGB model is used to a great extent in solving computer vision problems, but is better known for color representation in the displays of television sets and monitors. The value of a color by this model is best described as being a vector in three-space where red, green and blue represent the axis as shown in figure 3. Color is thus a result of the combination of the red, green and blue components. The origin of the cube (0, 0 and 0) represents pure black while its polar opposite (255, 255 and 255) represents pure white.

From studying and experimenting with various thresholds in RGB space, we found that the following rule worked well in removing some unnecessary pixels:

$$\mathbf{0.836G - 14 < B < 0.836G + 44 \Rightarrow Skin}$$

&

$$\mathbf{0.79G - 67 < B < 0.78G + 42 \Rightarrow Skin}$$

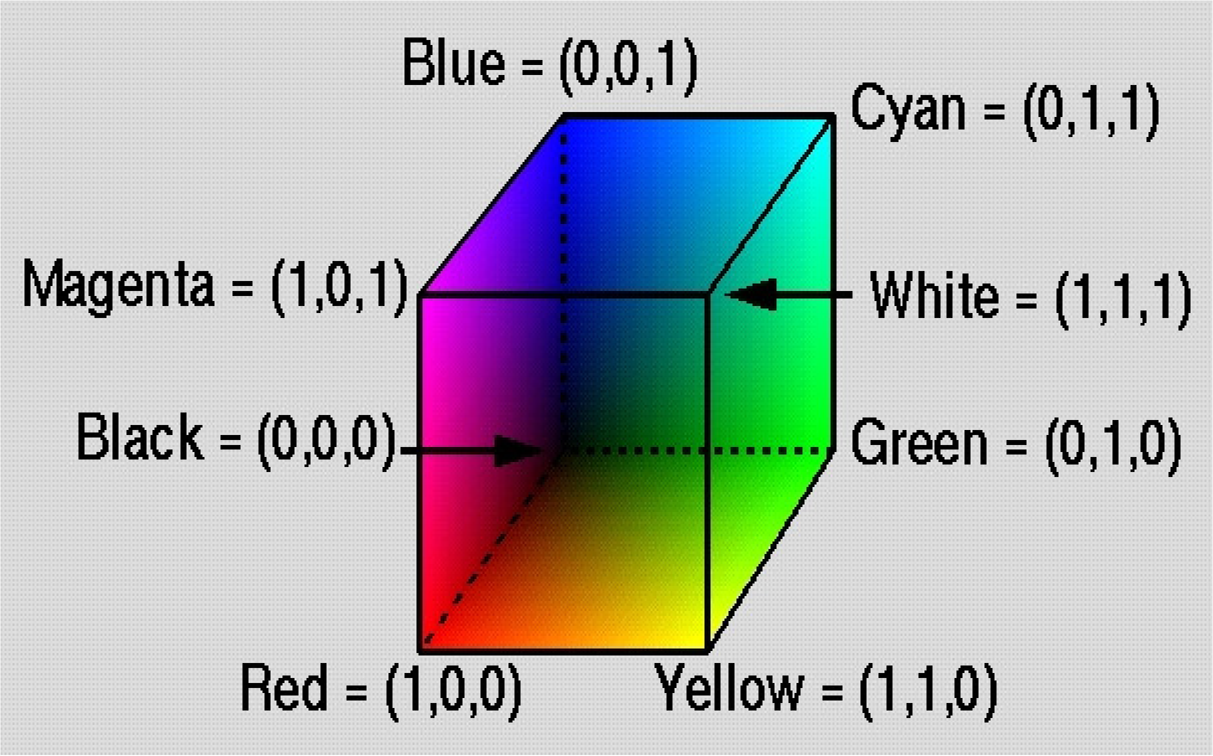


Figure 3: RGB color cube

3.4.2 HSI/HSV

While RGB may be most commonly used basis for color descriptions, it has the negative aspect that each of the co-ordinates (red, green and blue) is subject to luminance effects from the lighting intensity of the environment, an aspect which does not necessarily provide relevant information about whether a particular image "patch" is skin or not skin. The HSI color space, however, is much more intuitive and provides color information in a manner more in a line how human think of colors and how artists typically mix color information. This model is best presented as a color cone or cylinder. The HSI model comprises of three components. These are hue, saturation and intensity/value. This model is a more intuitive representation of colors.

The hue represents a person's representation of a color, for example green or orange. Hue changes as one move around the cone. Saturation is a measure of color's dilution by white light. This provides us with light or dark shades of a color. Saturation increases from the centre of the cone to the outside. Finally, the intensity is a measure of the brightness of the color. Intensity increases along the cone's vertical axis. The great advantage of using this color model is that it separates intensity from the color components unlike the RGB model that couples intensity with color information. It is the first two, H and S that will provide us with useful discriminating information regarding skin.

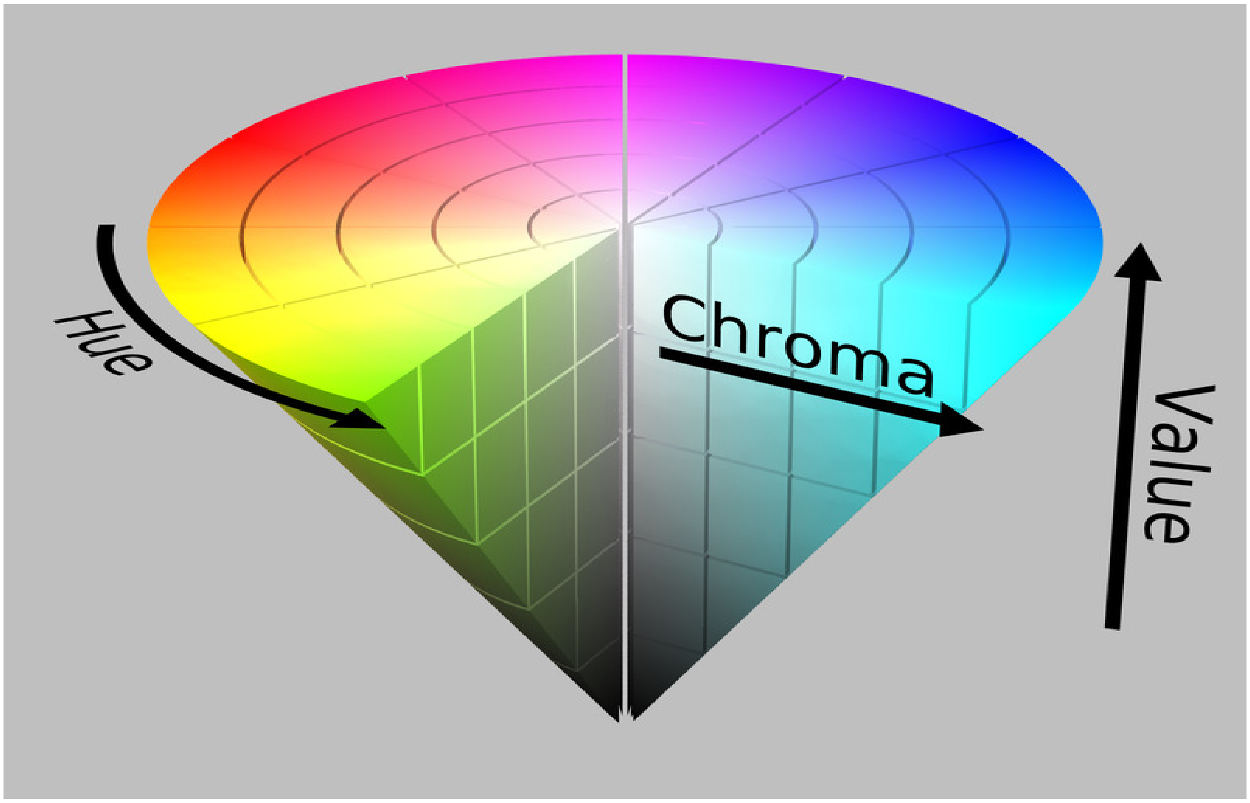


Figure 4: The HSV color cone

The threshold values used by us to derive the following rule used in our face skin detection block:

19 < H < 240 => Not skin

Otherwise we consider it as skin.

3.4.3 YCrCb

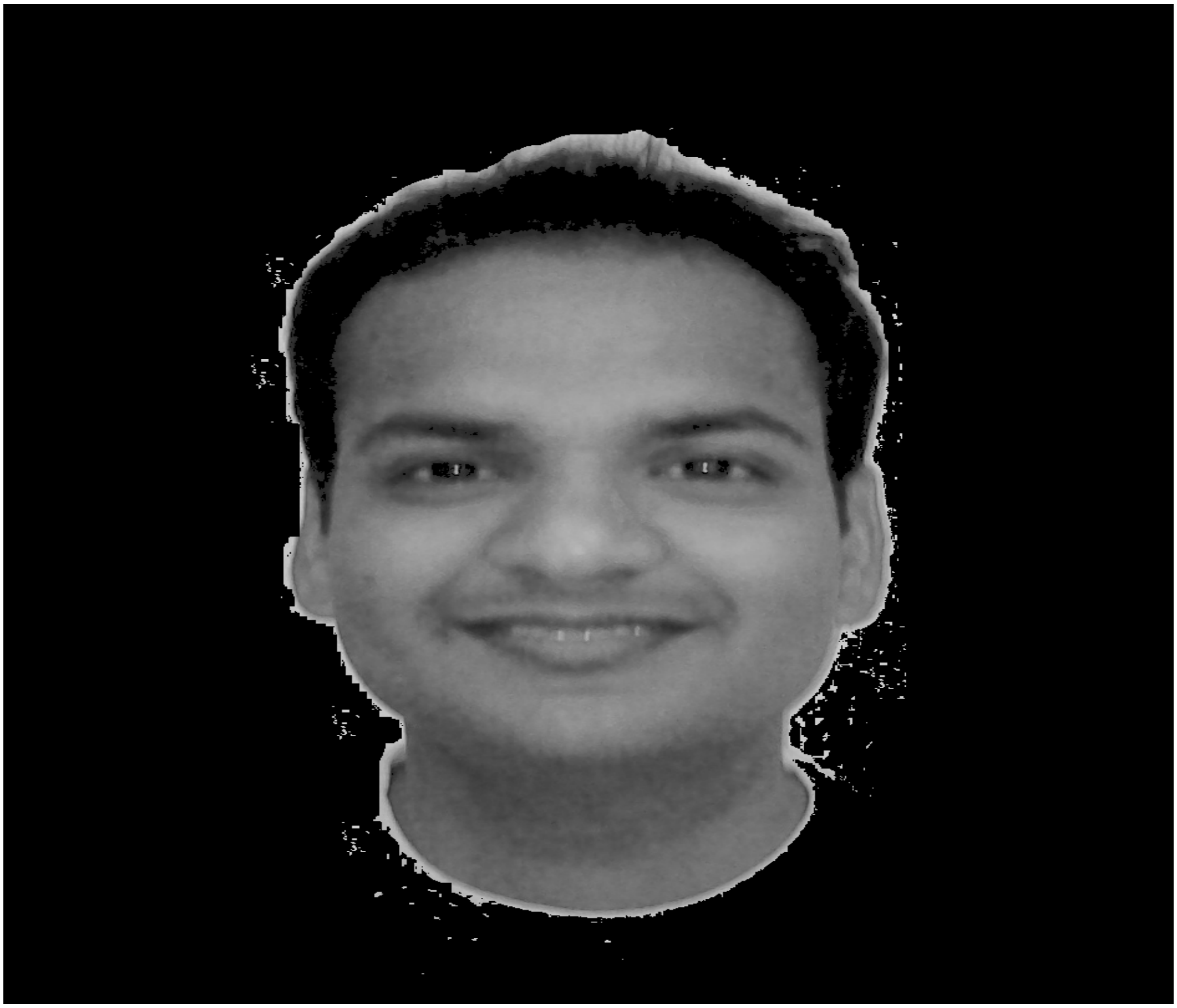
The YCrCb color space is widely used for digital video. In the format, luminance information is stored as single component (Y) and chrominance information is stored as two color-difference components (Cb and Cr). Cb represents the difference between the blue component and a reference value. Cr represents the difference between the red component and a reference value.

We analyzed the YCrCb color space for any trends that we could take advantage of to remove areas that are likely to be skin. After experimenting with various thresholds, we found that the best results could be found by using the following thresholds:

$$\left.\begin{array}{c} \mathbf{140 < Cr < 165} \\ \\ \mathbf{140 < Cr < 195} \end{array}\right\} \textbf{Skin}$$

Using these three color models we localize the skin color of the image.

Original Image



Skin Color Separated Image

Figure 5: Skin region separated from the non skin regions

3.5 Erosion and Dilation

The basic morphological operations:- erosion and dilation procedure give contrasting results when applied to either grayscale or binary images. Erosion shrinks image objects while dilation expands them. Te specific actions of each operation are covered in the following sections.

3.5.1 Characteristics of Erosion

- Erosion generally decreases the sizes of objects and removes small anomalies by subtracting objects with a radius smaller than the subtracting element.

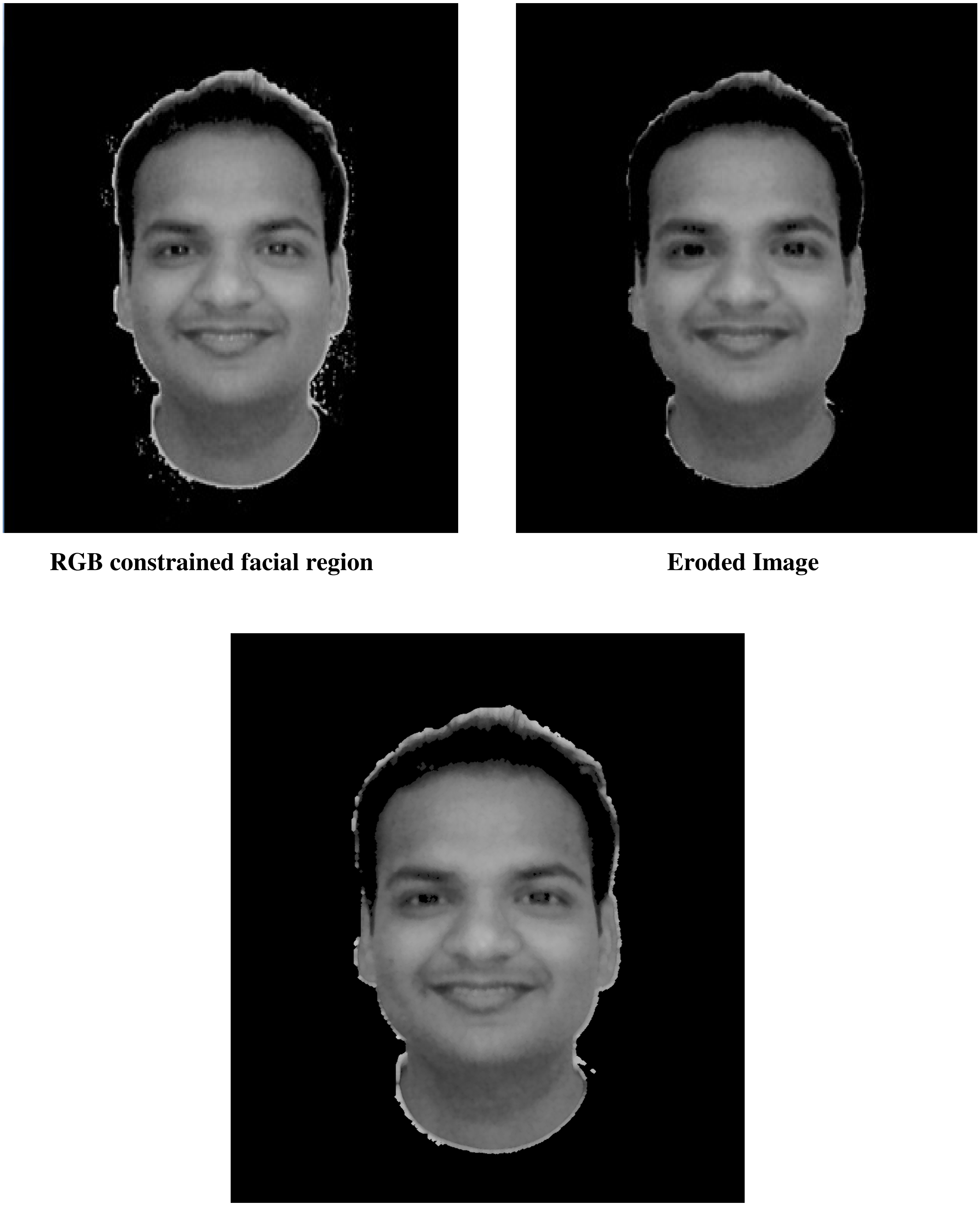
- With grayscale images, erosion reduces the brightness (and therefore the size) of bright objects on a dark background by taking the neighborhood minimum when passing the structuring element over the image.
- With binary images, erosion completely removes objects smaller than the structuring element and removes perimeter pixels from larger image objects.

3.5.2 Characteristics of Dilation

- Dilation generally increases the sizes of objects, filling in holes and broken areas and connecting areas that are separated by spaces smaller than the size of the structuring element.
- With grayscale images, dilation increases the brightness of objects by taking the neighborhood maximum when passing the structuring element over the image.
- With binary images, dilation connects areas that are separated by spaces smaller than the structuring element and adds pixels to the perimeter of each image object.

Once the skin color is localized the resultant image is converted to a binary image so that the processing is faster. Using a suitable size window, the noises in this black and white image are improved. The process of erosion and dilation is carried out to remove unwanted pixels which were falsely detected as skin color pixels by moving the window over the image matrix and if the number of white pixels is more than a fixed threshold of 10 pixels the whole window is made black.

The face detection involves taking the detected skin areas. Hence face localization based on the occurrence of white region in the eroded and diluted image is accomplished. This process has been developed based on the assumption that the image being processed has a single face with minimal skin exposure.



RGB constrained facial region

Eroded Image

Dilated Image

Figure 6: Noise Removal Using Erosion And Dilation

CHAPTER 4

CROPPING FACIAL REGION

After detection of the facial region, for further implementation, it is required to crop the area of interest. We know that lip region contains a maximum of red color content in it. If red color content is present somewhere outside the facial region, it will interfere in the lip detection process, thus increasing time of computing as well as complexity, without being of any use to us. Thus, by cropping the required region i.e. focusing on the required region only, results will be more precise and accurate and also image size on which we are processing will reduce, thus reducing computational time which an important factor.

Just before cropping and after face detection, erosion and dilation is applied. They are vital functions required here, as it helps in reducing the noise thus facilitating cropping. For cropping we first find four points i.e. the top, bottom, left and the right ones. After finding these four points, major work is done. Then height and width is found of the detected region is found. This is followed by creation of a rectangle and cropping of the required area of the image i.e. the facial region.

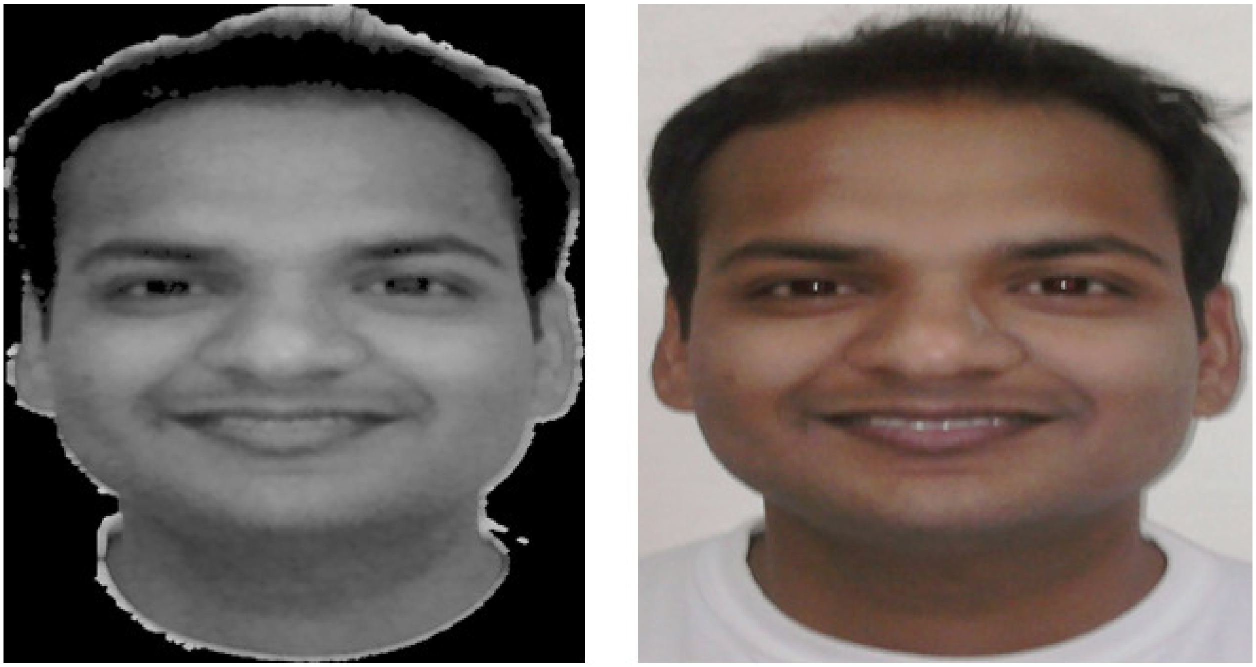


Figure 7: Cropped Facial Region

CHAPTER 5

LIP DETECTION AND SEGMENTATION

After face detection and getting our region of interest, we now move towards the most important part of the project i.e. lip detection. As already discussed in the 1st chapter, the variations seen during different emotions are around the lip region as well as the eyes region. But the maximum changes that are visible will be around the lip region, thus giving us more accurate results.

Recalling the RGB thresholds used for skin color pigments used for face detection:

$$0.836G - 14 < B < 0.836G + 44 \Rightarrow \text{Skin}$$

$$\&$$

$$0.79G - 67 < B < 0.78G + 42 \Rightarrow \text{Skin}$$

We see that, in these thresholds no restrictions have been imposed on the red color component. This is done as lips are composed of maximum of red colored pigments. Bounding the red colored pigments will result in loss of information thus, hampering the free working of the code i.e. lip detection.

The effective automatic location and tracking of a person’s lip is a problem that has been proven to be very difficult in the field of computer vision. Different methods for lip segmentation have been proposed in the last decade. The most important method for extracting lips is based on the segmentation directly from the color space. This kind of algorithm often uses a color transformation or color filter to enlarge the difference between the lips and the skin. The processing time is a prominent advantage of these algorithms. However, low color contrast between the lip and the skin for unadorned faces makes the problem difficult. The extraction of the lip is sensitive to color change. For images with weak color contrast, the method cannot satisfactorily outline the boundary of the lip as outer labial contour of the mouth has very poor color distinction when compared against its skin background.

5.1 Discrete Hartley Transform

The discrete Hartley transform (DHT) is an invertible linear transform or integral transform closely related to the discrete Fourier transform (DHT). It was proposed as an alternative to the Fourier transform by R. V. L. Hartley in 1942, and is one of many known Fourier-related transforms. The discrete version of the transform, the Discrete Hartley transform, was introduced by R. N. Bracewell in 1983. DHT has many analogous applications in signal processing and related fields.

In the DFT, one multiplies each input by cos - i * sin (a complex exponential), whereas in the DHT each input is multiplied by simply cos + sin. Thus, the DHT transforms ‘n’ real numbers to ‘n’ real numbers, with no intrinsic involvement of complex numbers.. The Hartley transform has the convenient property of being its own inverse (an involution):

$$\boxed{f = \{\mathcal{H}\{\mathcal{H}f\}\}}$$

There is a misconception that discrete Fourier transform is slower than discrete Hartley transform. This is seldom true and it all merely depends on the input sizes and other factors. Now seeing the mathematical aspects of the discrete Hartley transform,

The Hartley transform of a function *f* (*t*) is defined by:

$$\boxed{H(\omega) = \{\mathcal{H}f\}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t)\,\mathrm{cas}(\omega t)\mathrm{d}t,}$$

where ω can in applications be an angular frequency and

$$\boxed{\mathrm{cas}(t) = \cos(t) + \sin(t) = \sqrt{2}\sin(t + \pi/4) = \sqrt{2}\cos(t - \pi/4)}$$

is the cosine-and-sine or *Hartley* kernel. In engineering terms, this transform takes a signal (function) from the time-domain to the Hartley spectral domain (frequency domain). Now following shows how DHT is practically carried out:

$$C=\frac{1}{\sqrt{p}}\begin{bmatrix}1 & 1 & \dots & 1\\ 1 & \cos(2\pi/p)+\sin(2\pi/p) & \dots & \mathrm{cas}(2\pi(p-1)/p)\\ \vdots & \vdots & \vdots & \vdots\\ 1 & \mathrm{cas}(2\pi(p-1)/p) & \dots & \mathrm{cas}\left(2\pi(p-1)^2/p\right)\end{bmatrix}$$

In the case of the three channel signals ($\rho = 3$), which is of special interest in color image processing, the unitary transform for three channel signals is as follows :

$$\mathrm{C} = \begin{bmatrix}0.5773 & 0.5773 & 0.5773\\ 0.5773 & 0.2113 & -0.7886\\ 0.5773 & -0.7886 & 0.2113\end{bmatrix}$$

Much has been talked about DHT, now let us consider how this will be applied in this project. The three channel DHT has been applied to the color image as follows:

$$\begin{bmatrix}\mathrm{C}_1\\ C_2\\ C_3\end{bmatrix} = \begin{bmatrix}0.5773 & 0.5773 & 0.5773\\ 0.5773 & 0.2113 & -0.7886\\ 0.5773 & -0.7886 & 0.2113\end{bmatrix}\begin{bmatrix}R\\ G\\ B\end{bmatrix}$$

$C_1C_2C_3$ consists of two major components: luminance (C_1) and chrominance (C_2C_3). Luminance C_1 describes the brightness of the pixel color. Chrominance C_2C_3 describes the color portion of the pixel which includes hue and saturation information. The chrominance component C_3 of $C_1C_2C_3$ has high value around the lip by our observation.

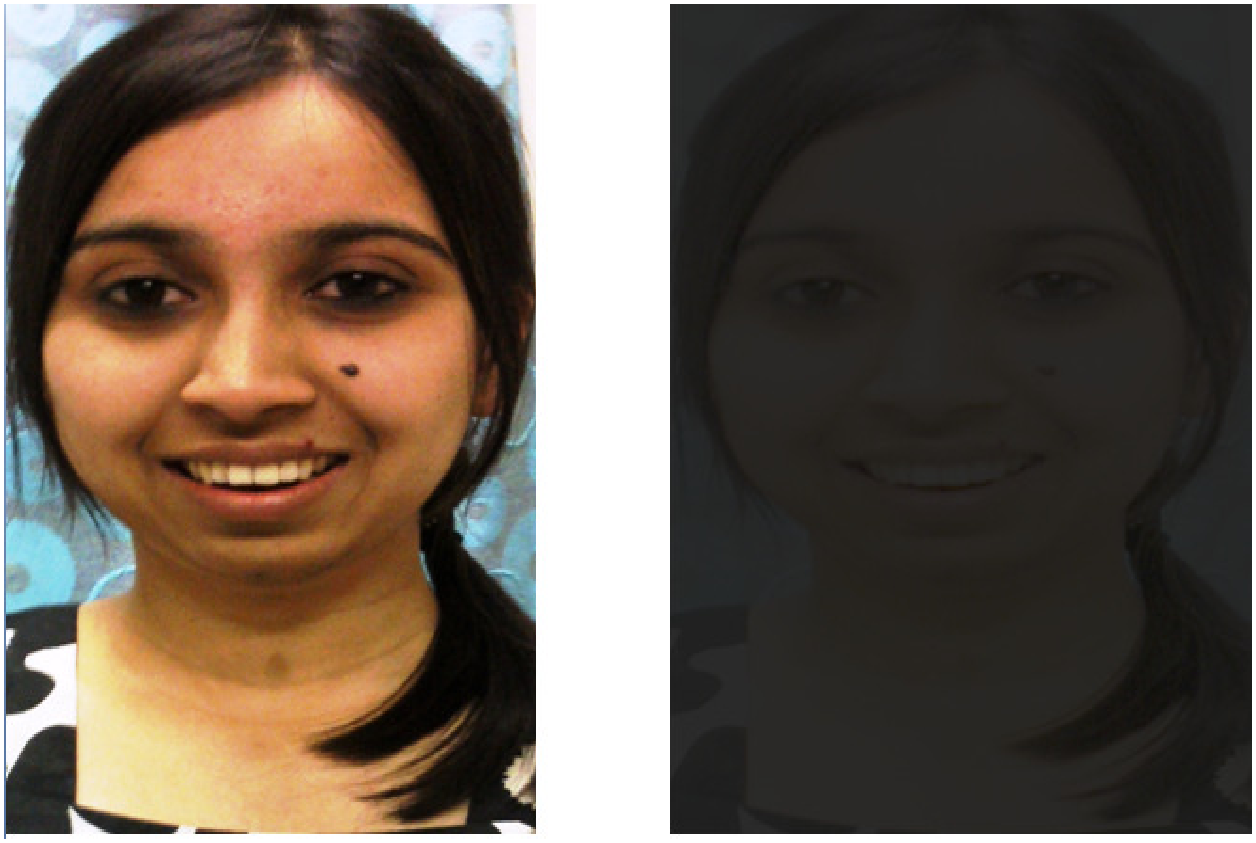
5.2 Preprocessing

Before applying DHT, some preprocessing is done. As it is seen the appearance of the skin color can be changed due to different lighting conditions. In order to increase the lip segmentation robustness to different lighting conditions, it is necessary to perform illumination compensation before converting the RGB values to its corresponding $C_1C_2C_3$.

There is physiological evidence that the response of cells in the retina is nonlinear in the intensity of the incoming image, which can be approximated as a log function of the intensity. The form of the log transform function used to normalize the luminance level is:

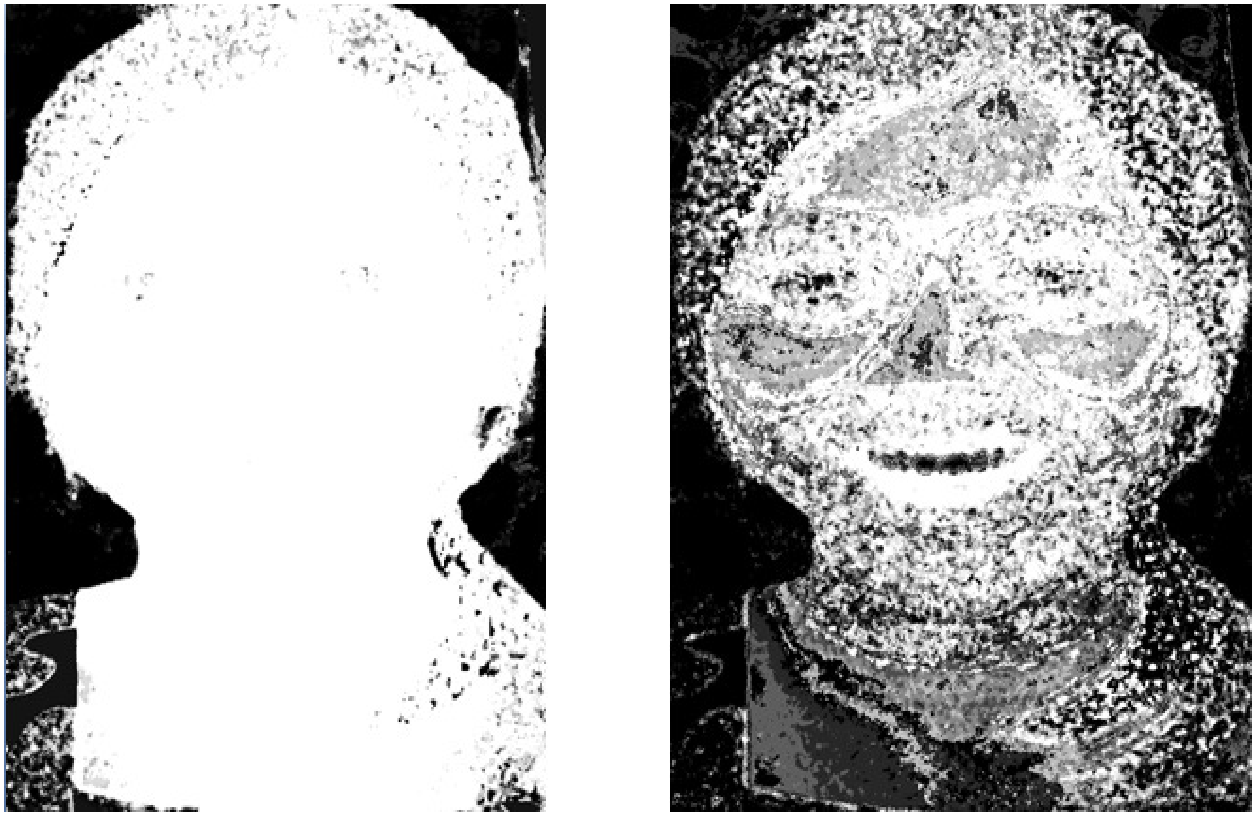
$$g\ (x\ ,y) = a + \frac{\log(f(x,y) + 1)}{b\log(c)}$$

Where f (x, y) is an original image, and a, b and c are parameters that control the location and shape of the curve. Transform the original image f (x, y) and then normalize g(x, y) to the range of [0, 255]. In the experiment, a, b and c are chosen to be 10, 0.25 and 2, respectively, as different selections of parameters have the same effect on the illumination compensation except b = 0 and c = 1 or c <= 0.



Original image **Normalized image**

Figure 8: Pre-processing



C2 (Chrominance) **C3 (Chrominance)**

Figure 9: Hartley transform of Normalized Image

5.3 Implementation

After doing preprocessing and applying discrete Hartley transform and converting RGB values to $C_1C_2C_3$ values it has been seen that around lip region C_3 has maximum value. So as to standardize the entire process a value has been set. This value is some factor of maximum value of C_3 and it is set after doing many experiments. It is seen that C_3(maximum)/4 gives the best result covering the entire lip region i.e. the required region. Similarly as done in face detection, functions of erosion and dilation are applied to remove noise. But before erosion and dilation, just to be more accurate Gaussian filter is applied on the image.

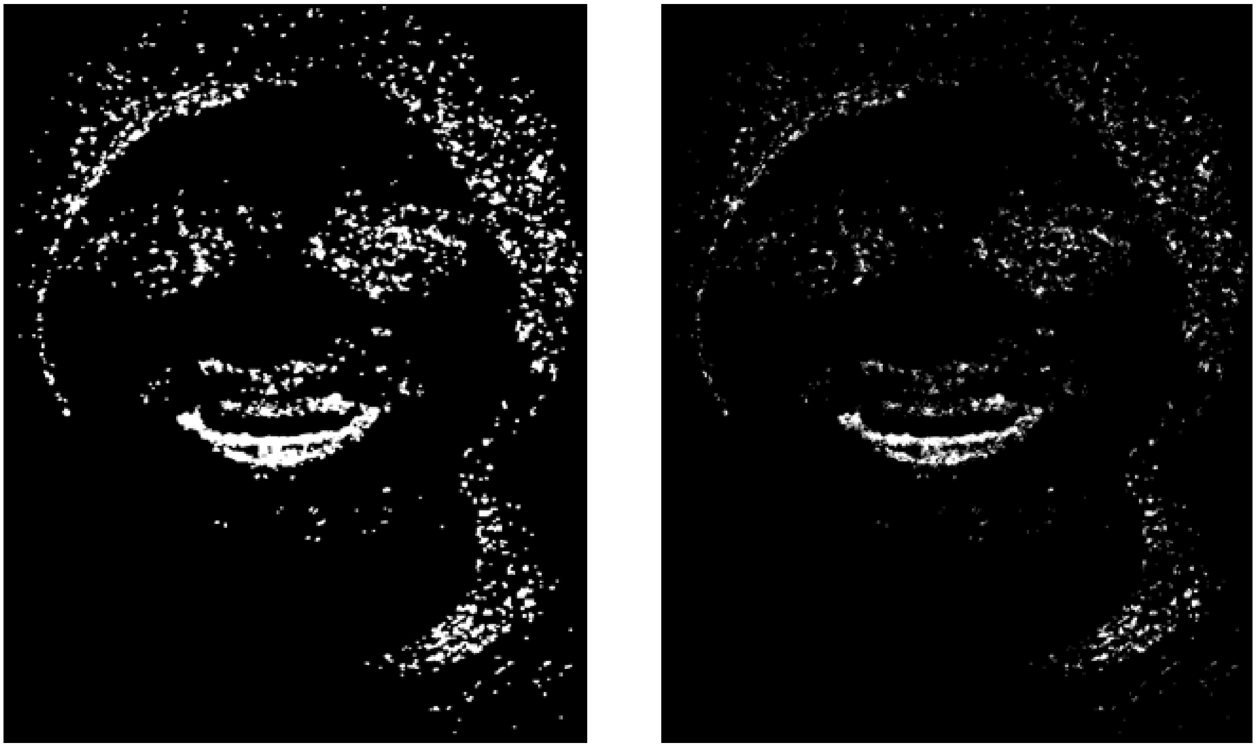
5.4 Gaussian Filter

In electronics and signal processing, a Gaussian filter is a filter whose impulse response is a Gaussian function. The Gaussian filter is a linear filter that is usually used as a smoother. The output of the Gaussian filter at the moment t is the weighted mean of the input values, and the weights are defined by formula:

$$\boxed{\omega\ (\tau) = C(\sigma) \cdot \exp\left(\frac{\tau^2}{2\sigma^2}\right)\ ;\quad \tau = \ldots, -1, 0, +1, \ldots\quad ,}$$

Where, τ is the "distance" in time from the current moment;

- σ is the parameter of the Gaussian filter;
- C (σ) is the normalization constant chosen to make the sum of all weights equal to the unit value.



C3 constrained image **C3 after passing through Gaussian filter**

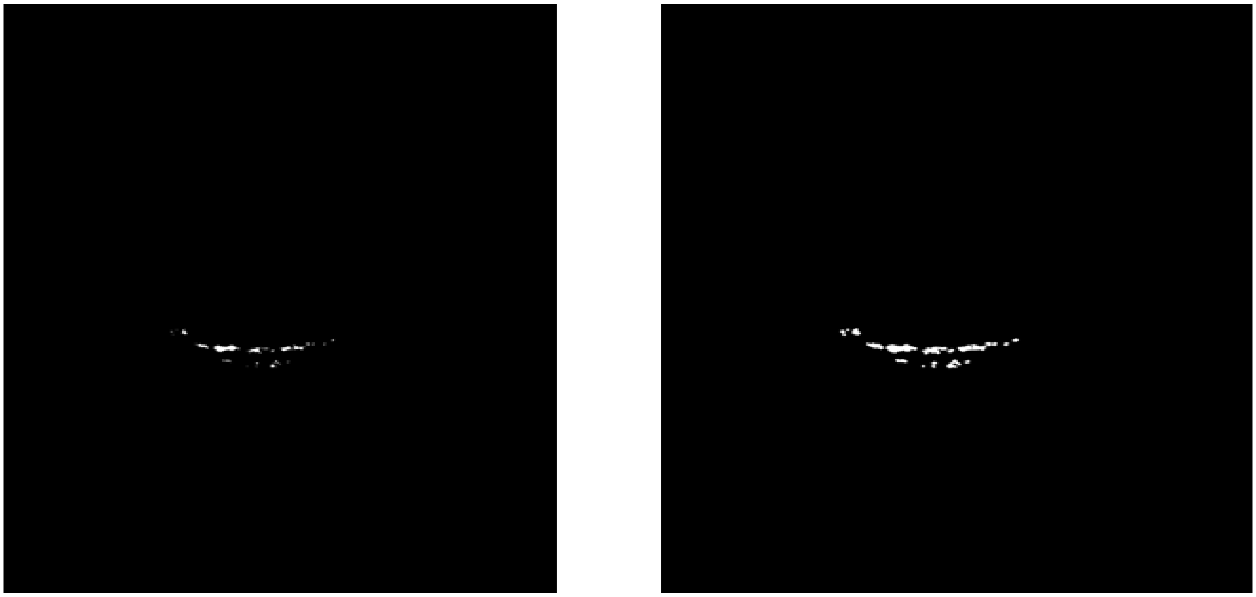
Figure 10: Image smoothing using Gaussian filter

If you plot the values of ω (τ) against τ, then the plot coincides with the famous bell-like curve describing the density of the Gaussian distribution. This explains the word "Gaussian" in the name of the filter. The Gaussian filter is completely defined by a single parameter σ. The greater is the value of σ, the wider the window function ω (τ), and, hence, greater the degree of smoothing. The Gaussian filter provides better suppression of higher frequencies than the rectangular filter and the triangular filter. Besides the one-dimensional Gaussian filter, there are extensions to the case of two dimensions, say, (x, y). Such two-dimensional Gaussian filters are widely used in image processing.

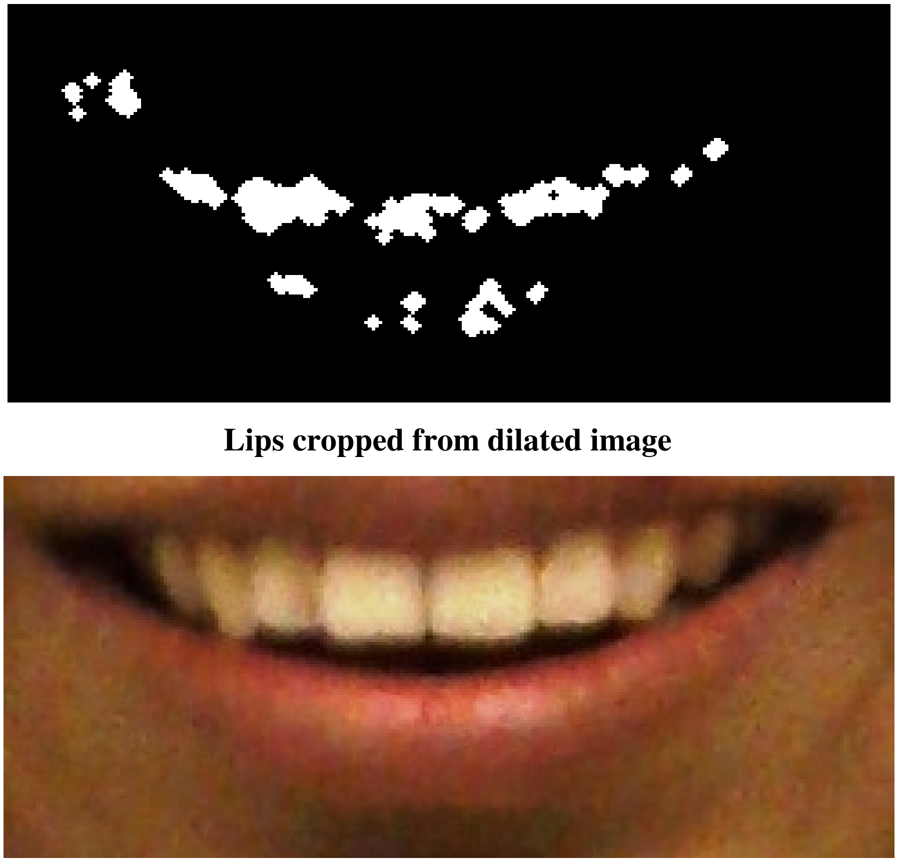
5.5 Lip Segmentation

With the completion of lip detection we are just left with lip segmentation and comparison with templates to detect emotions. The lip segmentation will precisely give us the cropped lip region, thus making us step closer to the ultimate goal of the project. For lip segmentation same procedure is used as that used for face segmentation. This process is started with finding of four points i.e. the top, bottom, left and right points of the detected lips. Then height and width of the detected lips are found.

There may be possibility of noise occurring in the case of lip detection and another area on the face may be confused for lips. To avoid this, there is a scientifically proven fact that the width of lip is approximately three times that of lip height. To be on the safer side, considering the worst case situation will be better. So instead of taking lip width three times that of lip height, lip width is taken greater than equal to two times that of lip height. Thus, if this condition is satisfied the rectangular cropping window is made and hence we get the cropped lip region.



Erosion **Dilation**



Lips cropped from dilated image

Lips cropped from original image

Figure 11: Process of Lip Segmentation from Original Image

CHAPTER 6
DATABASE GENERATION

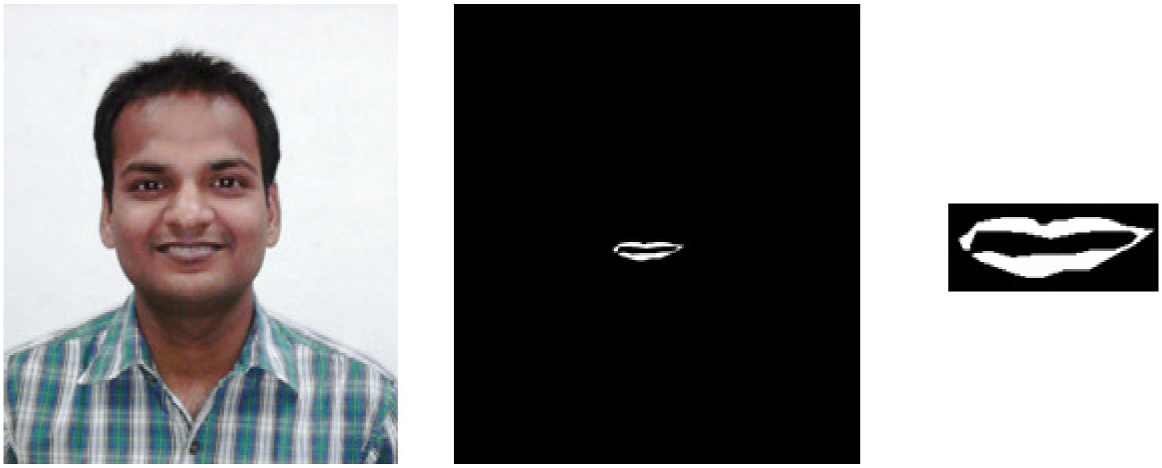
After, lip detection this is a necessary step for driving the project further. Lips so far detected are result of series of algorithms and conditions. For emotion detection through facial expressions in digital images, variation in lips constitutes 70-80 percent of emotions.

Lips have been detected but to reach towards the ultimate goal of the project the lips so detected have to be compared to something, in order to find what kind of emotion a particular face is depicting. So, a collection of binary images will help in detection of emotions that will be done by comparison of detected lips with these binary images in database. This implies, that database is merely a collection of images or templates.

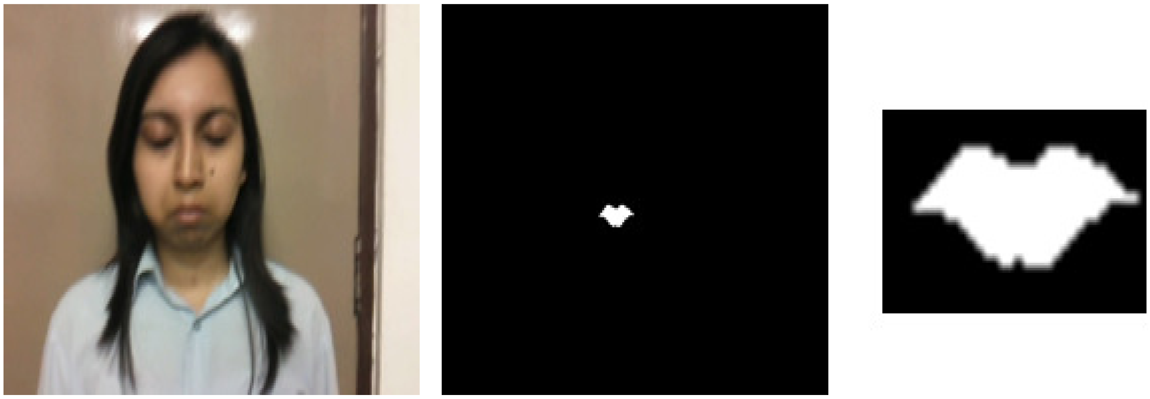
6.1 Template generation

In order to generate database, templates have to be generated. Templates are generated by a function called Roipoly. It is an in-built Matlab command. Roipoly is used to specify a polygonal region of interest (ROI) within an image. Roipoly returns a binary image that you can use as a mask for masked filtering.

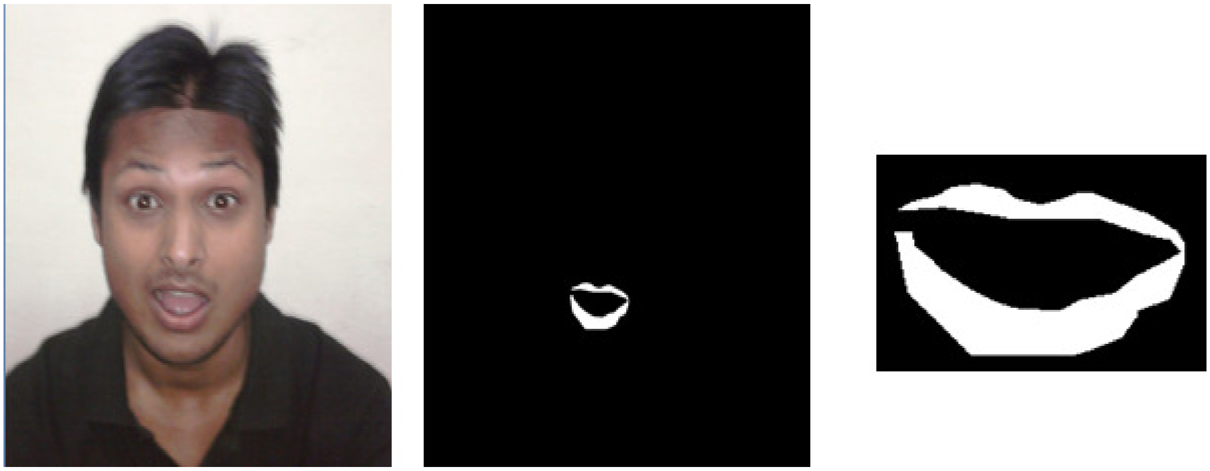
BW = roipoly creates an interactive polygon tool, associated with the image displayed in the current figure, called the target image. With the polygon tool active, the pointer changes to cross hairs, †, when you move the pointer over the image in the figure. Using the mouse, you specify the region by selecting vertices of the polygon or any area of interest. You can move or resize the polygon or figure using the mouse. When you are finished positioning and sizing the polygon/figure, create the mask by double-clicking, or by right-clicking inside the region and selecting Create mask from the context menu. Roipoly returns the mask as a binary image, BW, the same size as I. In the mask image, roipoly sets pixels inside the region to 1 and pixels outside the region to 0. This binary image BW so returned is in logical format and as such cannot be used for implementation. It needs to be converted to a format that can be processed on i.e. double or uint8. After this the masks are cropped in the similar manner used for face cropping and lip cropping i.e. finding four points and creating rectangle and finally cropping.



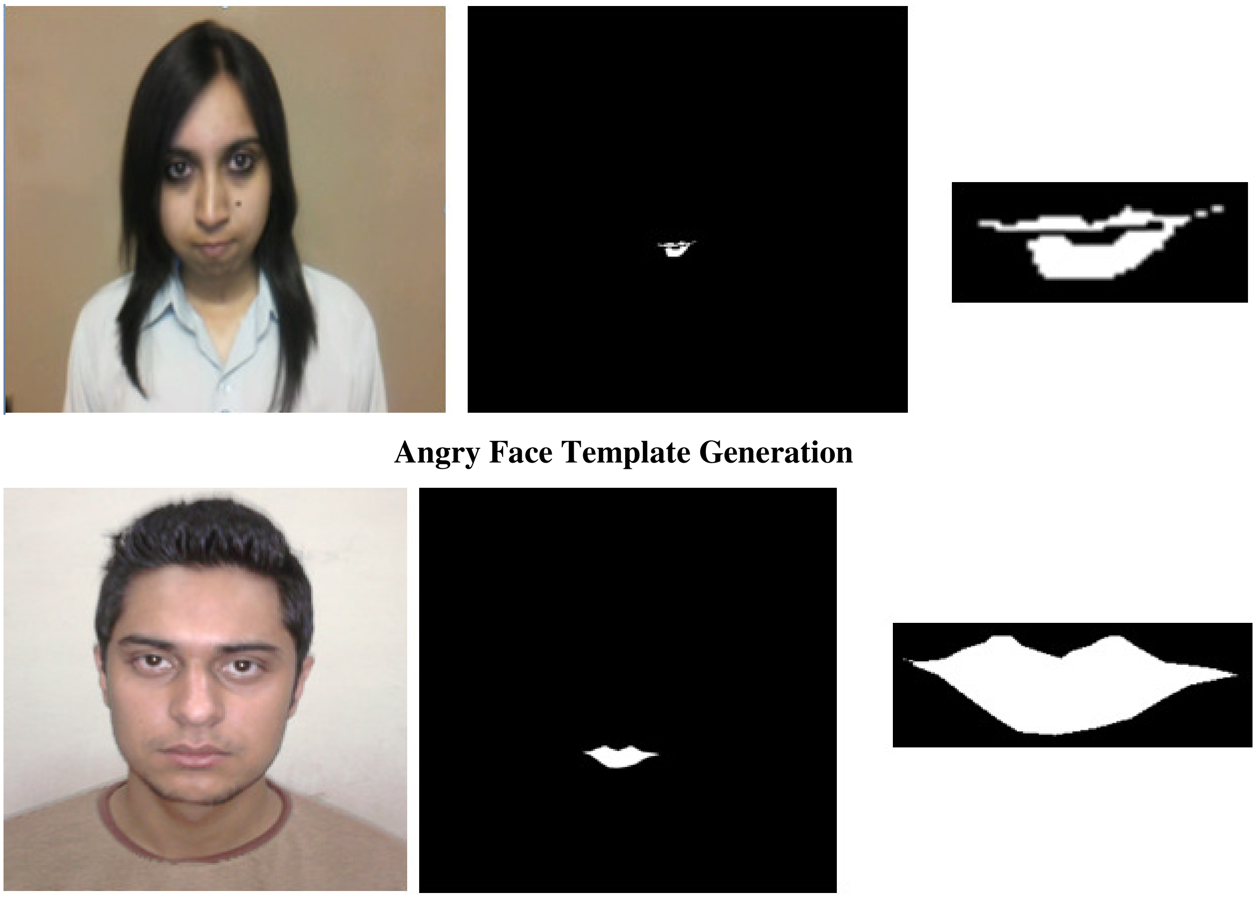
Smiling Face Template Generation



Sad Face Template Generation



Surprised Face Template Generation



Angry Face Template Generation

Neutral Face Template Generation

Figure 12: Different Emotion Templates

Hence a binary image is created that can now be compared in order to detect emotions. So, far the database used in this project contains a collection of almost 75-80 images covering the five types of emotions and all the variance that can be present in all those five types. Also, database can be enlarged by adding more images, thus making the detection more precise and accurate. Templates still need to be modified in order to be possible to compare.

6.2 Template Modifications

Templates have been created and cropped too but that isn't sufficient for processing and comparison. A lot of modifications have to be made. Firstly, out of all the templates available the template with maximum dimensions has to be searched.

This is followed by creation of a 3-D matrix of size; say m×n×p, where m×n is same as that of template with maximum dimensions and the third dimension p being the number of images in the database. Now sizes of all the templates is compared to that of image with maximum dimensions and if the dimension is less than m×n then zero-padding is done to make dimensions equal to m×n. Padding is done uniformly in all the directions. If padding is not done the image will not be of same dimensions thus comparison will become difficult and more complex, thus increasing computational cost as well as time. Then, the templates with similar dimensions are copied in the matrix and templates of similar emotion are all put together sequentially.

CHAPTER 7
EMOTION RECOGNITION

After doing face detection, lip detection, template generation and various segmentations, we now reach towards the final and the last step of our project. It is the easiest step of all the steps. It is simply the comparison of the detected lip region and the templates so made i.e. images in database. This comparison is made by using cross-correlation.

7.1 Cross-correlation

This option calculates the cross correlation function for two images. Each of the images is divided into rectangular blocks. Each block in the first image is correlated with its corresponding block in the second image to produce the cross correlation as a function of position.

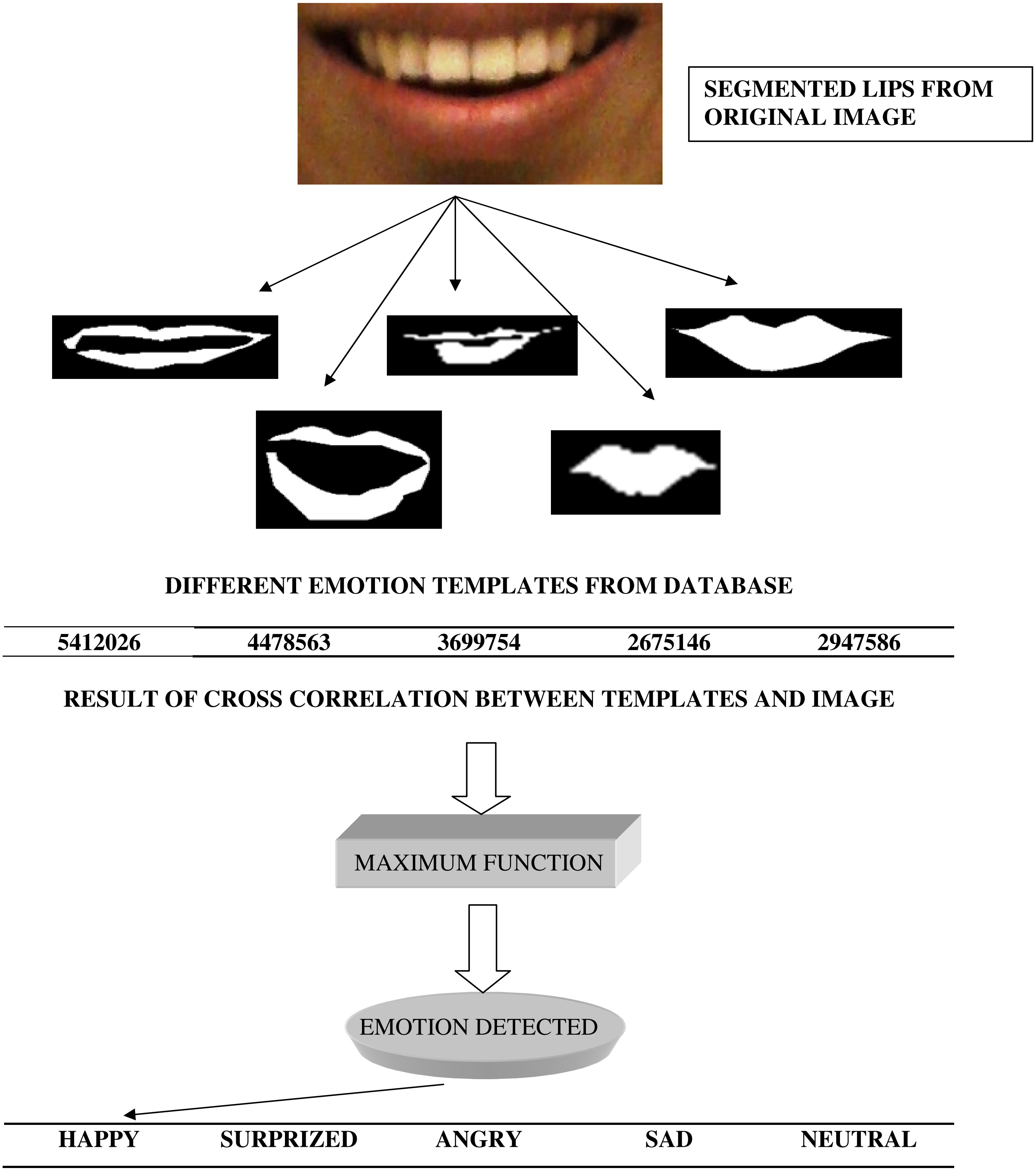
The cross correlation may be used to determine the degree of similarity between two similar images, or, with the addition of a linear offset to one of the images, the spatial shift or spatial correlation between the images. The function we have used in this project for correlation is xcorr2. It returns the cross-correlation of matrices A and B with no scaling. It has its maximum value when the two matrices are aligned so that they are shaped as similarly as possible. If matrix A has dimensions (Ma, Na) and matrix B has dimensions (Mb, Nb), the equation for the two-dimensional discrete cross-correlation is

$$\boxed{\mathrm{C(i,j)} = \sum_{\mathrm{m=0}}^{\mathrm{Ma-1}} \sum_{\mathrm{n=0}}^{\mathrm{Na-1}} \mathrm{A(m,n)} \ \cdot \mathrm{conj(B(m+i,n+j))}}$$

Where $0 \leq \ i < Ma+Mb-1$ and $0 \leq j < Na+Nb-1$

7.2 Emotion Detection

After finding cross-correlation of the detected and cropped lip region with templates, we have to find the index number (p) on which we get maximum value offered by cross-correlation of two images. Emotion is detected by checking the corresponding emotion across the index number with maximum value of cross-correlation.


SEGMENTED LIPS FROM ORIGINAL IMAGE
DIFFERENT EMOTION TEMPLATES FROM DATABASE
RESULT OF CROSS CORRELATION BETWEEN TEMPLATES AND IMAGE
MAXIMUM FUNCTION
EMOTION DETECTED

5412026	4478563	3699754	2675146	2947586

HAPPY	SURPRIZED	ANGRY	SAD	NEUTRAL

Figure 13: The whole process of emotion detection

CHAPTER 8
PROJECT EVALUATION

8.1 Results

The five emotions that we have been working on i.e. happiness, grief, anger, surprise and neutral were successfully identified on the majority of images used. The degree to how well facial region as well as lips detected varied from picture to picture depending on multiple factors. Generally no false emotion were found or wrongly interpreted. The only problem was that database generation was done only on limited faces. Most of the artifacts that were present in the image were rectified separately before applying algorithm.

8.2 Design

The development of the entire project can be broken into four parts:

1. The automation
2. The detection (face, lips)
3. The generation
4. The comparison and recognition

The design of the software was such that testing and modifications could be easily implemented.

8.3 Difficulties

Although this is a very little code, but we came across some minute difficulties at all levels. Starting with, our face detection algorithm works efficiently only for single face in the image and that too with a uniform background. in lip detection the value that is set as a standard might not work for all the faces, thus it might require some adjustments. If after face and lip detection noise is not removed, it can hamper the cropping of the respective regions and might not give accurate results. The templates made for database generation are done manually and region of interest is selected manually using mouse pointer, so, slightest of mistake may result in not so precise results. Dilation and erosion window size needs to be adjusted time to time depending on the images.

8.4 Success

The procedure has a high success rate. A test was conducted on several individuals as well arbitrary faces taken from internet and the result was satisfactory.

8.5 Future Work

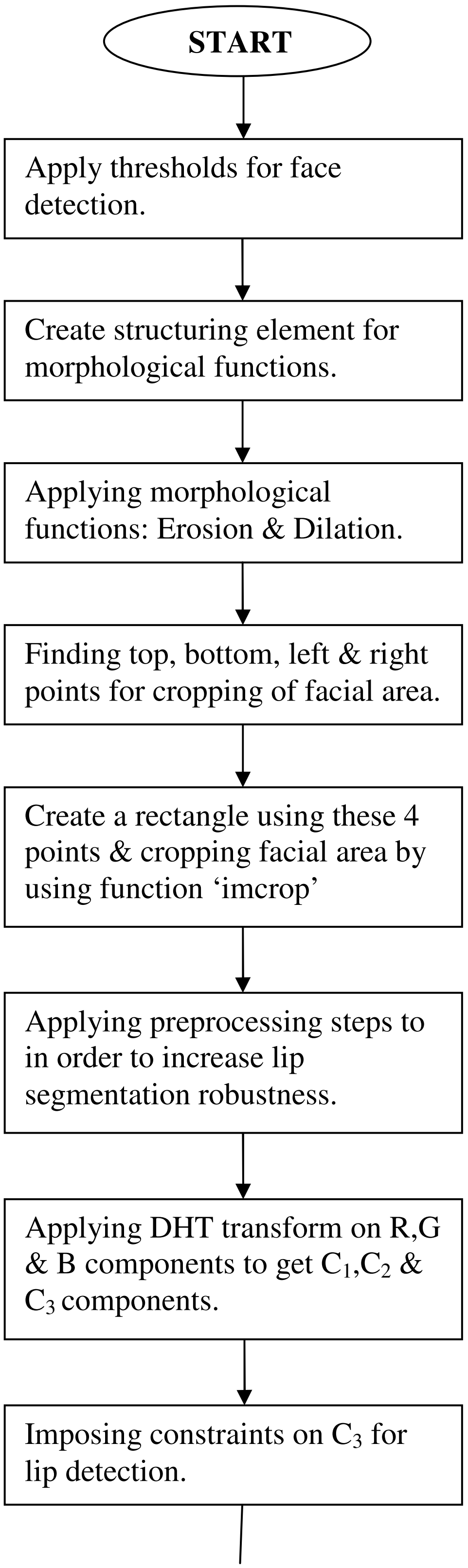
There are several aspects of this project that have high market potential, so, the concept needs to upgraded for better results. Firstly, this concept needs to be extended from a single face to multiple faces and this concept should be able to adapt itself to non-uniform backgrounds. Secondly, different poses, structural components as well as different imaging conditions should be no hurdle in the process. There should be option of automatic generations of similar dimension templates. More sensitive and adaptive thresholds can be developed for facial and lip detection. Inclusion of more templates in the database so as to make emotion recognition more precise and accurate. Many more emotions can also be introduced. Overall, the aim should be to make this algorithm more flexible and adaptive in real time applications.

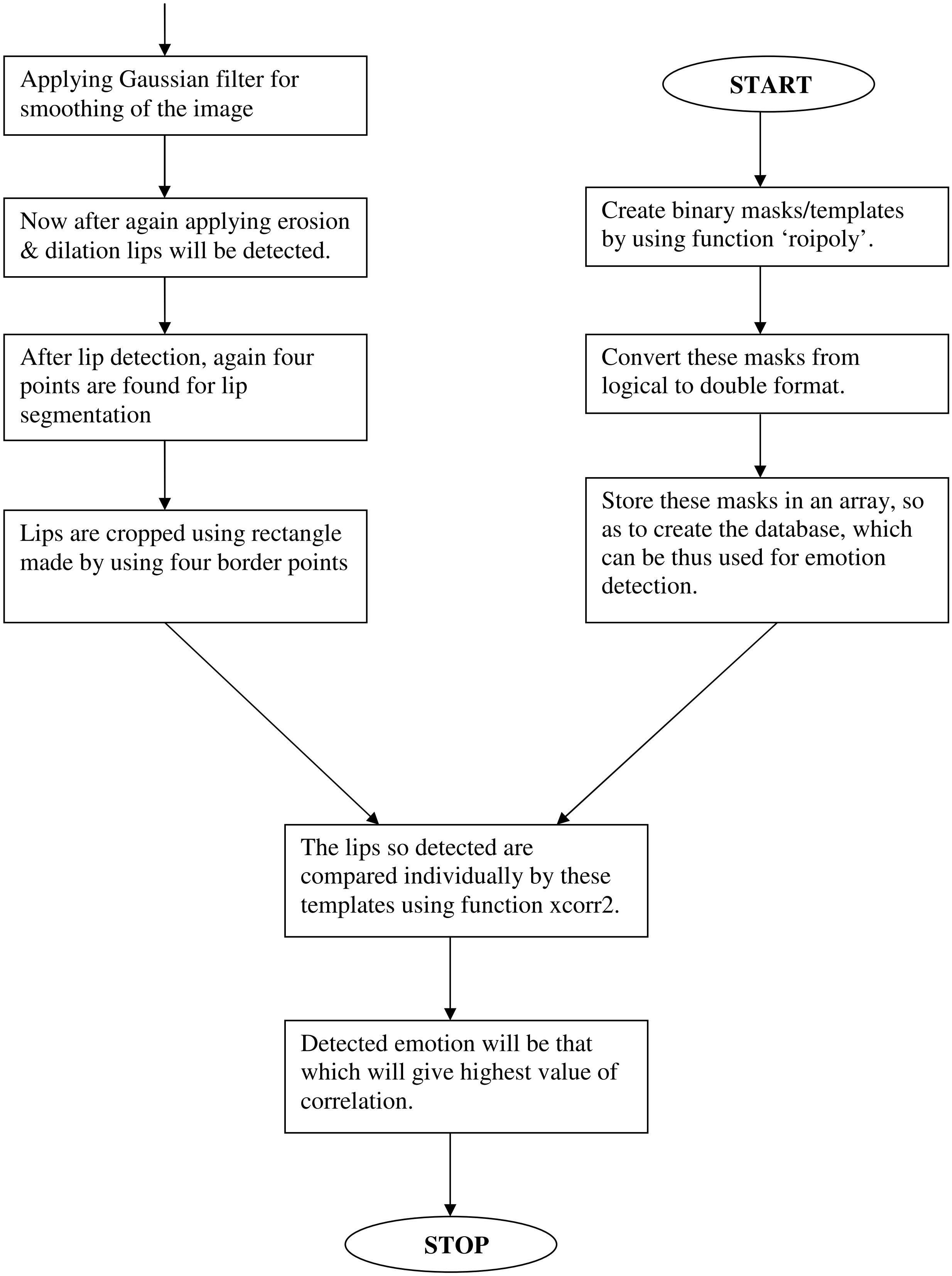
BIBLIOGRAPHY

1. A. Yuille, “Feature Extraction from Faces Using Deformable Templates,” Int. Journal of Computer Vision,(1992), 8(2):99-111.
2. Bracewell R.N.: “The Hartley transform” , Oxford University Press, 1986.
3. D.S. Berry, “What can a moving face tell us?” , J. Pers. Soc. Psychol.,1990,1004-1014
4. Gocke R., Millar J.B., Zelinsky A., Robert-Ribes J, “Automatic extraction of lip feature points”. Proc. Austral. Conf. Robotics and Automation, Melbourne, Australia, August 2000, pp. 31–36.
5. Li Y.-ZH, Kobatake K.: ‘Extraction of facial sketch images and expression transformation based on FACS’. Proc. Int. Conf. Image Processing, Washington, USA, October 1995, vol. 3, pp. 520–523.
6. M.H. Yang, “Detecting Faces in images: a survey”, at al IEEE trans. on PAMI, April1984, pp. 12-59.
7. M. Lievin, P.Delmas, P.Y.Coulon, F.Luthon and V.Fristot, “Automatic Lip Tracking: Bayesian Segmentation and Active Contours in a Cooperative Scheme,” IEEE Int. Conf. on Multimedia Computing and Systems, (1999), Vol.1, pp.691-696.
8. N.H. Frijda, “The understanding of facial expression of emotion”, Acta Psychol. 9 (1953) 294– 362.
9. P. Ekman, W.V. Friesen,” Pictures of Facial Affect.”, Consulting Psychologist Press, 1976, 256 - 264.
10. R.L. Buckner, M.M. Strauss, S.E. Hyman, B.R. Rosen, H.C. Breiter, N.L. Etcoff, P.J. Whalen, W.A. Kennedy, S.L. Rauch, “Response and habituation of the human amygdale during visual processing of facial expression”,1996,875 – 887.
11. Sanjay Kr. Singh,” A Robust Skin Color Based Face detection Algorithm”, Published in IET Computer Vision, 1997, Vol.3, pp.65-103.
12. U. Dimberg, M. Thunberg, K. Elmehed, “Unconscious facial reactions to emotional facial expressions”, Psychol. Sci. 11, 2000, 86– 89.
13. Woods, G. & Digital Image Processing.
14. Y.P. Guan,” Automatic extraction of lips based on multi-scale wavelet edge detection”, Published in IET Computer Vision, 2008, Vol. 2, No. 1, pp. 23–33.

APPENDIX A

FLOWCHART


START
Apply thresholds for face detection.
Create structuring element for morphological functions.
Applying morphological functions: Erosion & Dilation.
Finding top, bottom, left & right points for cropping of facial area.
Create a rectangle using these 4 points & cropping facial area by using function ‘imcrop’
Applying preprocessing steps to in order to increase lip segmentation robustness.
Applying DHT transform on R,G & B components to get C_1,C_2 & C_3 components.
Imposing constraints on C_3 for lip detection.


Applying Gaussian filter for smoothing of the image
Now after again applying erosion & dilation lips will be detected.
After lip detection, again four points are found for lip segmentation
Lips are cropped using rectangle made by using four border points
START
Create binary masks/templates by using function ‘roipoly’.
Convert these masks from logical to double format.
Store these masks in an array, so as to create the database, which can be thus used for emotion detection.
The lips so detected are compared individually by these templates using function xcorr2.
Detected emotion will be that which will give highest value of correlation.
STOP

APPENDIX B

PSEUDO CODE

NOTE: The platform used for implementation of algorithms is in Matlab R2008Ra

The whole project is divided in eight modules:

1) Facedetect.m

- Input the image.
- The image is converted into different planes of respective RGB color space.
- Use selective thresholds, retrieve the image is scanned pixel wise:

$0.836G - 14 < B < 0.836G + 44$
$79G - 67 < B < 0.78 + 42$ ⇨ Image in RGB model

$19 < H < 240$ ⇨ For image in HSV model

$140 < Cr < 165$
$140 < Cb < 195$ ⇨ Image in YCrCb model

- Pixels satisfying these conditions are termed as skin pixels.
- These pixels are separated out from rest of the image for further operations.
- Store the skin.bmp

2) Dilation.m

- Retrieve the image skin.bmp.
- To avoid noise the image is first erodes with a diamond shaped window of radius 3 pixels and then dilated with a diamond shaped window of radius 3-4 pixels.
- All kind of unwanted small noise is removed from the image.
- The image after dilation is stored as dilation.bmp

3) Facecrop.m

- The image dilation.bmp is retrieved along with original image.
- For cropping the image so as to reduce the net working area whole image is scanned pixel wise for non zero intensity values.
- Firstly, image is scanned column wise for each row from top and the first pixel that results in nonzero value is stored in a flag (top).
- Secondly, image is scanned row wise for each column from left and the first pixel that results in nonzero value is stored in a flag (left).
- Thirdly, image is scanned column wise for each row from bottom and the first pixel that results in nonzero value is stored in a flag (bottom).
- Lastly, image is scanned row wise for each column from right and the first pixel that results in nonzero value is stored in a flag (right).
- The two rows (top and bottom) and two columns (left and right) results in a rectangle with top minus bottom as height and right minus left as width of the rectangle surrounding the facial region strictly.
- Using these flags and other values derived from these flags facial region is cropped from the original image.
- The cropped image is stored in crop.bmp.

4) Normalize.m

- The image crop.bmp is retrieved.
- To normalize the image a log function that converts the scalar intensity to non linear function is used

$$g\ (x\ ,y) = a + \frac{\log(f(x,y) + 1)}{b\log(c)}$$

  Where, a b & c are constants and f(x,y) is crop.bmp.
- The above step normalizes the luminance level in the image.
- Store the image as normalize.bmp.

5) Lipdetection.m

- Retrieve the normalize.bmp
- Break the normalized image into respective RGB components.
- Apply discrete Hartley transform onto the RGB components to get C1(luminance), C2 and C3(chrominance) using the relation:

$$\begin{bmatrix} \mathrm{C}_1 \\ C_2 \\ C_3 \end{bmatrix} = \begin{bmatrix} 0.5773 & 0.5773 & 0.5773 \\ 0.5773 & 0.2113 & -0.7886 \\ 0.5773 & -0.7886 & 0.2113 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

- C3 component is maximum across lips, applying constraints over c3 will result in detection of lip region pixels.
- Noise removal and smoothing is also done using erosion and dilation of satisfactory window size and Gaussian filter is used for smoothing the image before erosion and dilation.
- Save the image as lipdetect.bmp

6) Lipsegmentation.m

- Open the image lipdetect.bmp and crop.bmp.
- Cropping is needed to avoid unnecessary computing and is done in similar way as of face.
- While cropping, dimensions are increased by 10 to 15 pixels to get whole lips from crop.bmp
- Cropped lips is stored as seglips.bmp

7) Templategenerate.m

- Open separate emotion illustrating images one by one.
- Use the function *roipoly* for selecting the region of interest i.e. lips.
- Convert the results of *roipoly* i.e. logical image into integer images.
- Modify each template separately by cropping and selecting only region of interest.
- Cluster all similar emotion templates and name them in serial order.

- Find the biggest dimension among the templates and construct all templates of that size using concept of zero padding.
- Construct a 3-d matrix (mxnxp) in which (mxn) is the size of template and (p) is the index no. of the template.
- Save the matrix as database.mat

8) Emotiondetection.m

- Retrieve the image seglips.bmp and matrix database.bmp
- Compare each template separately with seglips.bmp using correlation function.
- Save the maximum value of correlation into a flag (max) and corresponding index no. into another flag (indx).
- Check the emotion corresponding to flag (indx) with maximum value of correlation.
- Emotion is detected.