B. TECH. PROJECT REPORT

On

Gender Recognition Using Partial Face Images

BY

SAURAB AGRAWAL

DISCIPLINE OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY INDORE

December 2016

Gender Recognition Using Partial Face Images

A PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the degrees

of
BACHELOR OF TECHNOLOGY
in
ELECTRICAL ENGINEERING

Submitted by:
Saurabh Agrawal

Guided by:
Dr. Vivek Kanhangad

INDIAN INSTITUTE OF TECHNOLOGY INDORE
December 2016

CANDIDATE'S DECLARATION

I hereby declare that the project entitled **"Gender Recognition Using Partial Face Images"** submitted in partial fulfillment for the award of the degree of Bachelor of Technology in 'Electrical Engineering' completed under the supervision of **Dr. Vivek Kanhangad, Assistant Professor Electrical Engineering,** Indian Institute Of Technology, Indore is an authentic work.

Further, I declare that I have not submitted this work for the award of any other degree elsewhere.

Saurabh Agrawal
(130002033)

CERTIFICATE by BTP Guide

It is certified that the above statement made by the students is correct to the best of my knowledge.

Dr. Vivek Kanhangad
Assistant Professor, Electrical Engineering
Indian Institute Of Technology, Indore

Preface

This report on "Gender Recognition using Partial Face Images" is prepared under the guidance of Dr. Vivek Kanhangad.

Through this report I have tried to provide detailed analysis of gender classification using partial face Images. This method, in general can be applied to identify gender of any person using the face parts used in this report.

I have tried to the best of my abilities and knowledge to explain every part of the content in a lucid manner. I have also added sample database images, figures and graphs to make it more illustrative and easier to understand.

Saurabh Agrawal(130002033)
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

Acknowledgements

I wish to thank **Dr. Vivek Kanhangad** for his kind support and valuable guidance.

It is his help and support, due to which I became able to complete the design and technical report. Also, providing his precious time to me even when I went unannounced without any prior mail and guiding me, not as a professor but as a mentor, did create very friendly environment.

Without his support this report would not have been possible.

Saurabh Agrawal (130002033)
B.Tech. IV Year
Discipline of Electrical Engineering
IIT Indore

Abstract

In today's world, where every task is done by machine, computer vision and machine learning plays a crucial role in all image related mechanization whether it is about searching for the image of particular person in thousands of data images or about processing some image to reduce its size before saving in database. But sometimes, searching for the person in a big database can be cumbersome and time consuming and any help to reduce this time would be highly appreciated. This can be done by dividing the database based on gender which can ideally reduce the database search time to half. Therefore, this report presents methods to examine the effects of facial features on gender classification and extract those features for classification. Face images were firstly decomposed by 2-D Discrete Wavelet Transform (DWT2) to determine the geometric based facial features. Different wavelets and different number of filter levels were applied to see the effect of Wavelet Transform. After DWT2, for dimension reduction, Principal Component Analysis (PCA) was were applied to decomposed coefficients. For appearance based features, Gray-Level Co-occurrence Matrix (GLCM) was used to calculate feature values. After getting the features, Support Vector Machine (SVM) classifier is applied to the features coefficients. Experimental results indicated that the eyes are the most influential part for gender classification. Moreover Wavelet Transform decreases process time maintaining the error rate of PCA and SVM. When 1-level DWT2 is used there is no increase in error rate however there is an acceptable increase in error rate when 2-level DWT is used.

Keyboards:

- Gender recognition
- Feature extraction
- 2D - Discrete Wavelet Transform
- Principal Component Analysis
- Gray-level Co-occurrence Matrix
- Support Vector Machine

Table of Contents

List Of Figures

Chapter 1

Introduction

In the last decade, computer vision and pattern recognition field draw more attention than other related fields due to current machine dependency in every other task of daily life. Computer vision tasks include methods for acquiring, processing, analyzing and understanding digital images, and in general, deal with the extraction of high-dimensional data from the real world in order to produce numerical or symbolic information. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics, and learning theory.

1.1 Biometrics

Biometrics is the measurement and statistical analysis of people's physical and behavioral characteristics. The technology is mainly used for identification and access control, or for identifying individuals that are under surveillance. The basic premise of biometric authentication is that everyone is unique and an individual can be identified by his or her intrinsic physical or behavioral traits.

Examples of physiological characteristics used for biometric authentication include fingerprints; DNA; face, hand, retina or ear features; and odor. Behavioral characteristics are related to the pattern of the behavior of a person, such as typing rhythm, gait, gestures and voice. Certain biometric identifiers, such as monitoring keystrokes or gait in real time, can be used to provide continuous authentication instead of a single one-off authentication check.

The accuracy and cost of readers has until recently been a limiting factor in the adoption of biometric authentication solutions but the presence of high quality cameras, microphones, and fingerprint readers in many of today's mobile devices means biometrics is likely to become a considerably more common method of authenticating users, particularly as the new FIDO specification means that two-factor authentication using biometrics is finally becoming cost effective and in a position to be rolled out to the consumer market.

The quality of biometric readers is improving all the time, but they can still produce false negatives and false positives. One problem with fingerprints is that people inadvertently leave their fingerprints on many surfaces they touch, and it's fairly easy to copy them and create a replica in silicone. People also leave DNA everywhere they go and someone's voice is also easily captured. Dynamic biometrics like gestures and facial expressions can change, but they can be captured by HD cameras and copied. Also, whatever biometric is being measured, if the measurement data is exposed at any point during the authentication process, there is always the possibility it can be intercepted. This is a big problem, as people can't change their physical attributes as they can a password. While limitations in biometric authentication schemes are real, biometrics is a great improvement over passwords as a means of authenticating an individual.

1.2 Pattern Recognition

Pattern recognition is a branch of machine learning that focuses on the recognition of patterns and regularities in data, although it is in some cases considered to be nearly synonymous with machine learning. Pattern recognition systems are in many cases trained from labeled "training" data (supervised learning), but when no labeled data are available other algorithms can be used to discover previously unknown patterns (unsupervised learning).

Computers don't "see" photos and videos in the same way that people do. When you look at a photo, you might see your best friend standing in front of her house. From a computer's perspective, that same image is simply a bunch of data that it may interpret as shapes and information about color values. While a computer won't react like you do when you see that photo, a computer can be trained to recognize certain patterns of color and shapes. For example, a computer might be trained to recognize the common patterns of shapes and colors that make up a digital image of a face. This process is known as facial detection, and it's the technology that helps Google to protect your privacy on services like Street View, where computers try to detect and then blur the faces of any people that may have been standing on the street as the Street View car drove by.

Google uses pattern recognition: Beyond facial detection technology, Google also uses facial recognition in certain features. Facial recognition, like the name suggests, can help a computer to compare known faces against a new face and see if there is a probable match or similarity. For example, facial recognition helps users of the Find my Face feature to see suggestions about who they might want to tag in a photo or video they've uploaded and would like to share.

1.3 Feature Selection

Feature selection, also known as variable selection, attribute selection or variable subset selection, is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques are used for three reasons:

- simplification of models to make them easier to interpret by researchers/users,

- shorter training times,
- enhanced generalization by reducing overfitting

The central premise when using a feature selection technique is that the data contains many features that are either redundant or irrelevant, and can thus be removed without incurring much loss of information.[2] Redundant or irrelevant features are two distinct notions, since one relevant feature may be redundant in the presence of another relevant feature with which it is strongly correlated.

1.4 Feature Extraction

Feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations. Feature extraction is related to dimensionality reduction. Feature extraction involves reducing the amount of resources required to describe a large set of data. When performing analysis of complex data one of the major problems stems from the number of variables involved. Analysis with a large number of variables generally requires a large amount of memory and computation power, also it may cause a classification algorithm to overfit to training samples and generalize poorly to new samples. Feature extraction is a general term for methods of constructing combinations of the variables to get around these problems while still describing the data with sufficient accuracy.

1.5 Facial Features

First impressions of people — such as whether they are trustworthy, dominant or attractive — can develop from a glimpse as brief as 100 milliseconds or less. Brain scans suggests that such judgments are made automatically, probably outside of people's conscious control. But now, a computer system that mimics the human brain has identified which facial features most influence how others first perceive a person, scientists say. These findings could lead to computer programs that automatically see which photographs would help people give the best first impressions they can, the researchers added

Facial expressions are widely used in the behavioral interpretation of emotions, cognitive science, and social interactions. Facial features are broadly classified in two types: Geometric based Features and Appearance based feature.

1.6 Geometric Based Features

Geometry-based features describe the shape of the face and its components, such as the mouth or the eyebrow. Geometric feature learning methods extract distinctive geometric features from images. Geometric features are features of objects constructed by a set of geometric elements like points, lines, curves or surfaces. These features can be corner features, edge features, Blobs, Ridges, salient points image texture and so on, which can be detected by feature detection methods

In the geometric feature-based approach the primary step is to localize and track a dense set of facial points. Most geometric feature-based approaches use the active appearance model (AAM) or its variations, to track a dense set of facial points. The locations of these facial landmarks are then used in different ways to extract the shape of facial features, and movement of facial features, as the expression evolves.

1.7 Appearance Based Features

Features which are based on texture of the image like expression, emotions etc. The appearance features that have been successfully employed for emotion recognition are local binary pattern (LBP) operator , histogram of orientation gradients (HOG) , local Gabor binary patterns (LGBP) , local directional pattern (LDP) , non-negative matrix factorization (NMF) based texture feature , Gabor filter based texture information , principle component analysis (PCA) , linear discriminant analysis (LDA), etc.

Chapter 2

2.1 Discrete Wavelet Transform

Wavelet Transform has gained widespread acceptance in signal processing and image compression. It is a powerful technique for representing data at different scales and frequencies. The fundamental idea of wavelet transform is that a given function f(x) is analyzed to different resolution by projecting the function on the wavelet plane. The wavelet plane consists of the set of basis functions that have different resolution and f(x) is transformed into sub-functions with different resolutions.

The wavelet transform is similar to the Fourier transform (or much more to the windowed Fourier transform) with a completely different merit function. The main difference is this: Fourier transform decomposes the signal into sines and cosines, i.e. the functions localized in Fourier space; in contrary the wavelet transform uses functions that are localized in both the real and Fourier space. Generally, the wavelet transform can be expressed by the following equation:

$$F(a,b)=\int_{-\infty}^{\infty} f(x)\psi^{*}_{(a,b)}(x)dx$$

,Where $\psi_{(a,b)}(x)$ is the mother wavelet.

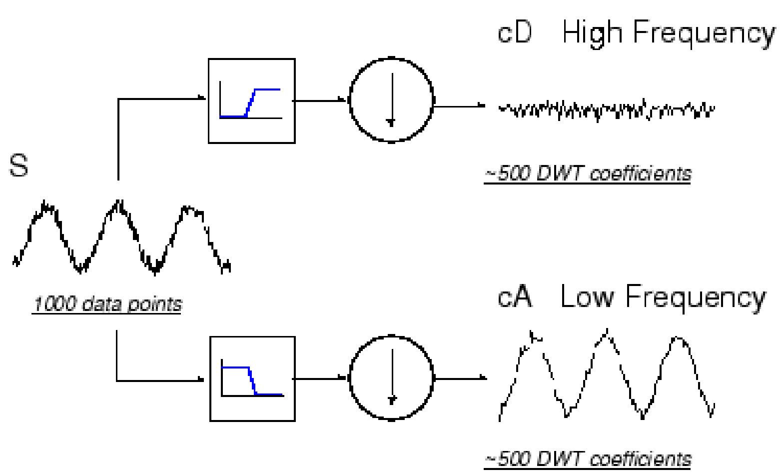


Figure 2.1: DWT of signal S

The two-dimensional DWT can be implemented using digital filters and downsamplers. Figure 2 shows the structure of filter bank that performs the decomposition of 2-D wavelet transform. As seen in the figure, firstly rows are filtered and down-sampled then columns are filtered and down-sampled. Due to downsampling image dimensions decrease.

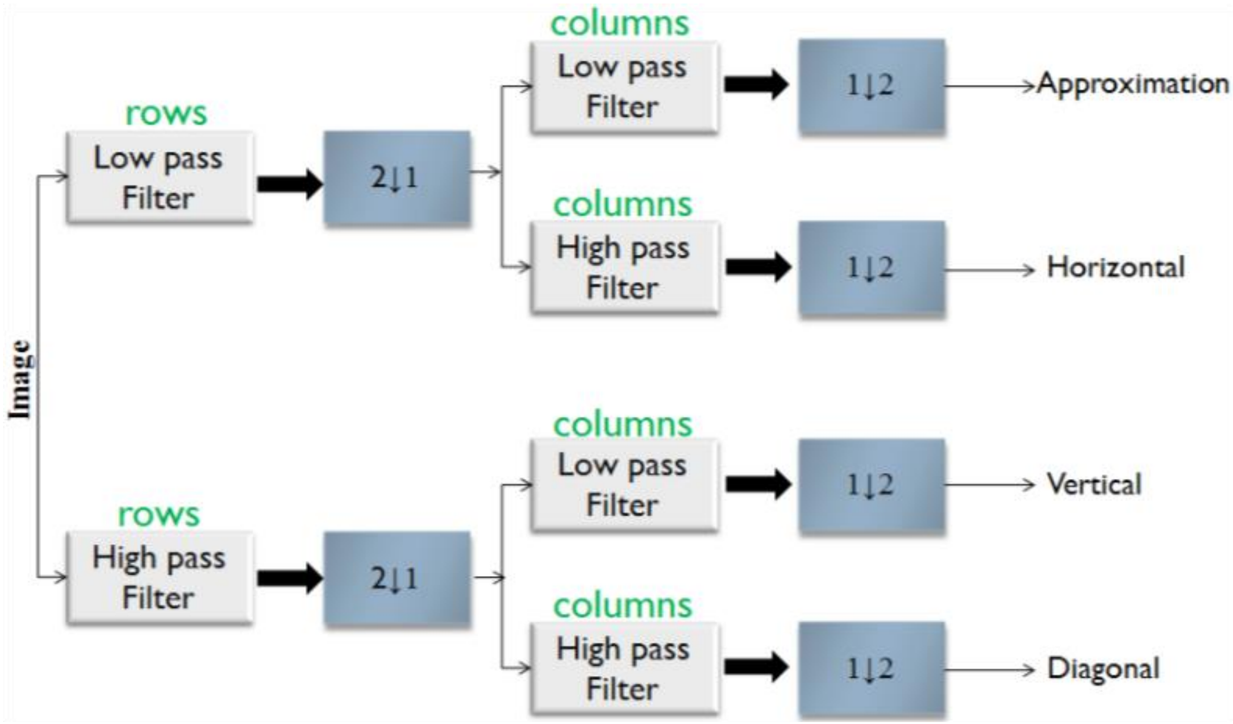


Figure 2.2: 2-D DWT of image

Single level discrete 2-D wavelet transform computes the approximation coefficient matrix and detail coefficient matrices, which are horizontal, vertical and diagonal respectively. Generally, in recognition algorithms, low frequency components of faces are used or image is firstly filtered with low-pass filters to get rid of noise and redundant data. In this study, approximation, which can be regarded as low-pass filter output of the image, was used. Approximation coefficient matrix carries more information about face that can be targeted again by single level discrete 2-D wavelet transform to reduce the dimensions.

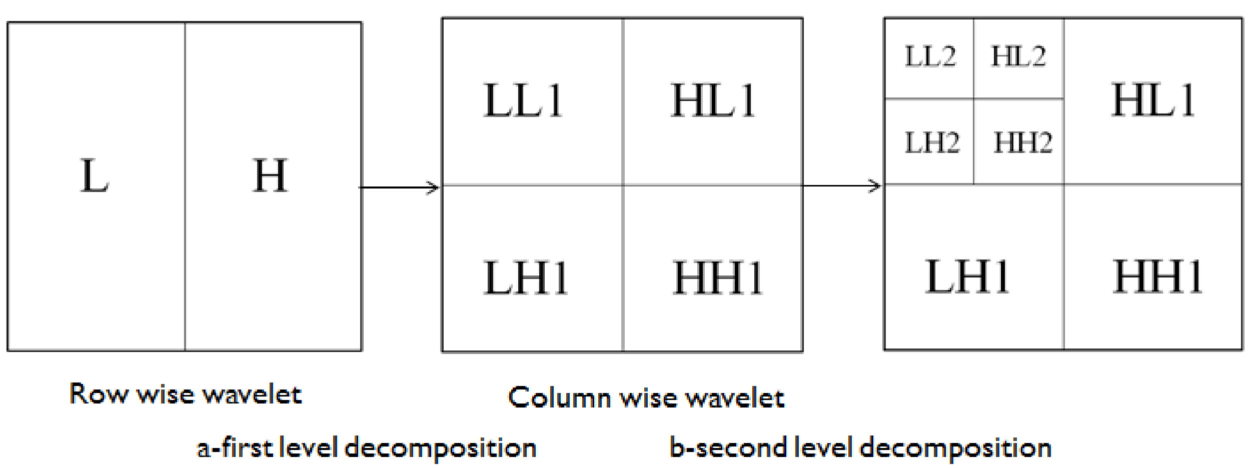


Figure 2.3: 2-Level Decomposition of image

2.2 Principal Component Analysis

In many real world problems, reducing dimension is an essential step before any analysis of the data can be performed. The general criterion for reducing the dimension is the desire to preserve most of the relevant information of the original data according to some optimality criteria. In pattern recognition and general classification problems, methods such as Principal Component Analysis (PCA), Independent Component Analysis (ICA) and Fisher Linear Discriminate Analysis (LDA) have been extensively used. These methods find a mapping from the original feature space to a lower dimensional feature space.

PCA is one of the most effective methods in data compression and pattern recognition. The aim of the PCA is to reduce the dimension of the data. PCA is used to omit redundant data for feature extraction, data compression and prediction. Since the PCA works in linear domain, it is used in linear applications, such as, signal processing, image processing, system and control theory and communication. In some applications it might be desired to pick a subset of the original features rather then find a mapping that uses all of the original features. The benefits of finding this subset of features could be in saving cost of computing unnecessary features.

Principal component analysis in signal processing can be described as a transform of a given set of n input vectors (variables) with the same length *K* formed in the n-dimensional vector $x = [x_1, x_2, x_3, ...x_n]^T$ into a vector y according to

$$y = A(x - m_x) \tag{1}$$

This point of view enables to form a simple formula (1) but it is necessary to keep in the mind that each row of the vector *x* consists of *K* values belonging to one input. The vector m_x in Eq. (1) is the vector of mean values of all input variables defined by relation

$$m_x = E\{x\} = \frac{1}{K}\sum_{k=1}^{K} x_k \tag{2}$$

Matrix $\boldsymbol{A}$ in Eq. (1) is determined by the covariance matrix $\boldsymbol{C_x}$. Rows in the A matrix are formed from the eigenvectors **e** of $\boldsymbol{C_x}$ ordered according to corresponding eigenvalues in descending order. The evaluation of the $\boldsymbol{C_x}$ matrix is possible according to relation

$$C_x = E\{(x-m_x)(x-m_x)^T\} = \frac{1}{K}\sum_{k=1}^{K} x_k x_k^{\ T} - m_x m_x^{\ T} \tag{3}$$

As the vector x of input variables is n-dimensional it is obvious that the size of C_x is n x n. The elements $C_x(i, i)$ lying in its main diagonal are the variances

$$C_x(i,i) = E\{(x_i - m_i)^2\} \tag{4}$$

of $\boldsymbol{x}$ and the other values $C_x(i, j)$ determine the covariance between input variables $\boldsymbol{x_i}$, $\boldsymbol{x_j}$.

$$C_x(i,j) = E\{(x_i - m_i)(x_j - m_j)\} \tag{5}$$

between input variables $\boldsymbol{x_i}$, $\boldsymbol{x_j}$. The rows of A in Eq. (1) are orthonormal so the inversion of PCA is possible according to relation

$$x = A^T Y + m_x \tag{6}$$

The algorithm can be summarized in the following steps:

1. Compute the sample covariance matrix, or use the true covariance matrix if it is available. In some cases it is preferred to use the correlation matrix instead of the covariance matrix. The correlation matrix is defined as the n×n matrix whose i, j'th entry is

$$\rho_{ij} = \frac{E(x_i x_j)}{E(x_i^{\ 2})E(x_j^{\ 2})}$$

This representation is preferred in cases where the features have very different variances from each other, and using the regular covariance form will cause the PCA to put very heavy weights on the features with the highest variances.

2. Compute the Principal components and eigenvalues of the Covariance/Correlation matrix.

3. Choose the subspace dimension q and construct the matrix A_q from A. This can be chosen by deciding how much of the variability of the data is desired to be retained.

4. Cluster the vectors to p ≥ q clusters using K-Means algorithm. The distance measure used for the K-Means algorithm is the Euclidean distance. The reason to choose p greater then q in some cases is if the same retained variability as the PCA is desired, a slightly higher number of features are needed.

5. For each cluster, find the corresponding vector which is closest to the mean of the cluster. Choose the corresponding feature, as a principal feature. This step will yield the choice of p features.

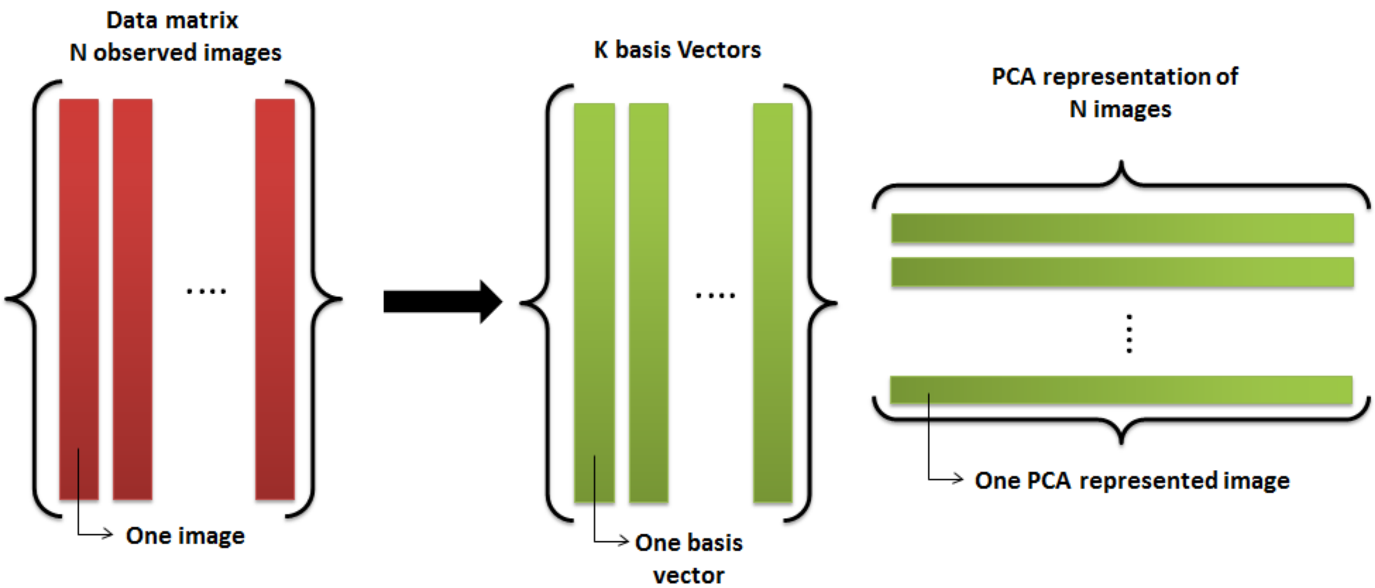


Figure 2.4: PCA of N images reduced size of set to K images

<u>Drawbacks</u>: By rearranging pixels column by column to a 1D vector, relations of a given pixel to pixels in neighboring rows are not taken into account. Another disadvantage is in the global nature of the representation; small change or error in the input images influences the whole Eigen-representation. However, this property is inherent in all linear integral transforms.

In general, PCA allows us to obtain a linear M-dimensional subspace of the original N-dimensional data, where M ≤ N. Furthermore, if the unknown, uncorrelated components are Gaussian distributed, then PCA actually acts as an independent component analysis since uncorrelated Gaussian variables are statistically independent. However, if the underlying components are not normally distributed, PCA merely generates decorrelated variables which are not necessarily statistically independent.

2.3 Gray-level Co-occurrence Matrix

One of the simplest approaches for describing texture is to use statistical moments of the intensity histogram of an image or region. Using only histograms in calculation will result in measures of texture that carry only information about distribution of intensities, but not about the relative position of pixels with respect to each other in that texture. Using a statistical approach such as co-occurrence matrix will help to provide valuable information about the relative position of the neighboring pixels in an image.

In statistical texture analysis, texture features are computed from the statistical distribution of observed combinations of intensities at specified positions relative to each other in the image. According to the number of intensity points (pixels) in each combination, statistics are classified into first-order, second order and higher-order statistics. The Gray Level Co-occurrence Matrix (GLCM) method is a way of extracting second order statistical texture features. The approach has been used in a number of applications, Third and higher order textures consider the relationships among three or more pixels. These are theoretically possible but not commonly implemented due to calculation time and interpretation difficulty.

Haralick et all first introduced the use of co-occurrence probabilities using GLCM for extracting various texture features. Gray Level Co-occurrence Matrix (GLCM) takes an image and estimates its properties related to the second-order statistics. It takes two pixels and a spatial relation between them (distance d and direction) and gives the corresponding number of occurrences of the pair of gray levels.

The GLCM for an Image *I* of size NxM can be calculated using the equation given below,

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

Where *i* and *j* are the pixel values; *x* and *y* are the spatial positions in the image *I*; the offsets (Δx, Δy) define the spatial relation for which this matrix is calculated; and *I(x, y)* indicates the pixel value at pixel *(x, y)*.

Whether considering the intensity or grayscale values of the image or various dimensions of color, the co-occurrence matrix can measure the texture of the image. Because co-occurrence matrices are typically large and sparse, various metrics of the matrix are often taken to get a more useful set of features. Texture analysis is often concerned with detecting aspects of an image that are rotationally invariant. To approximate this, the co-occurrence matrices corresponding to the same relation, but rotated at various regular angles (e.g. 0, 45, 90, and 135 degrees), are often calculated and summed.

After making the GLCM matrix, there is still one step to take before texture measures can be calculated. The measures require that each GLCM cell contain not a count, but rather a probability. This process is called normalizing the matrix. Normalization involves dividing by the sum of values.

$$P_{i,j} = \frac{V_{i,j}}{\sum_{i,j=0}^{N-1} V_{i,j}}$$

, where *i, j* are the rows and column number respectively

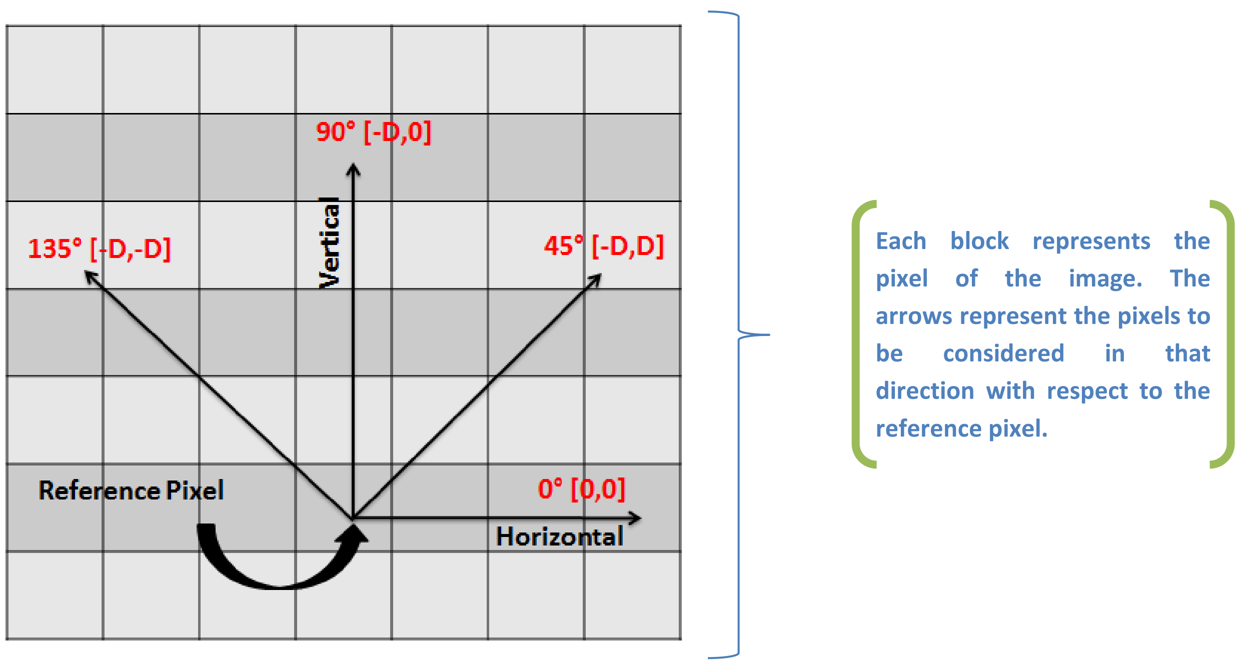


Figure 2.5: GLCM calculation along 4 directions

To illustrate consider a 4x4 image represented by figure 5 with four gray-tone values 0 through 3 (N=4). A generalized GLCM for that image is shown in figure 6 where #(i, j) stands for number of times gray tones i and j have been neighbors satisfying the condition stated by displacement vector D.

0	0	1	1
0	0	1	1
0	2	2	2
2	2	3	3

Figure 2.6: pixel values of 4x4 image

Gray tone	0	1	2	3
0	#(0,0)	#(0,1)	#(0,2)	#(0,3)
1	#(1,0)	#(1,1)	#(1,2)	#(1,3)
2	#(2,0)	#(2,1)	#(2,2)	#(2,3)
3	#(3,0)	#(3,1)	#(3,2)	#(3,3)

Figure 2.7: General form of GLCM

The four GLCM for angles equal to 0°, 45°, 90° and 135° and radius equal to 1 are shown in following figures. These are symmetric matrices hence evaluation of either upper or lower triangle serves the purpose. Frequency normalization can be employed by dividing value in each cell by the total number of pixel pairs possible.

4	2	1	0
2	4	0	0
1	0	6	1
0	0	1	2

Figure 2.8: GLCM for D=1 and ϕ=0°

6	0	2	0
0	4	2	0
2	2	2	2
0	0	2	0

Figure 2.9: GLCM for D=1 and ϕ=90°

2	1	3	0
1	2	1	0
3	1	0	2
0	0	2	0

Figure 2.10: GLCM for D=1 and ϕ=45°

4	1	0	0
1	2	2	0
0	2	4	1
0	0	1	.

Figure 2.11: GLCM for D=1 and ϕ=135°

Choice of Distance 'D' and angle 'ϕ' will yield different GLCM. Various research studies show D values ranging from 1, 2 to 10. Applying large displacement value to a fine texture would yield a GLCM that does not capture detailed textural information. Every pixel has eight neighboring pixels allowing eight choices for θ, which are0°, 45°, 90°, 135°, 180°, 225°, 270° or 315°. However, taking into consideration the definition of GLCM, the co-occurring pairs obtained by choosing ϕ equal to 0° would be similar to those obtained by choosing θ equal to 180°. This concept extends to 45°, 90° and 135° as well. Hence, one has four choices to select the value of ϕ. Sometimes, when the image is isotropic, or directional information is not required, one can obtain isotropic GLCM by integration over all angles.

After forming the GLCM matrix, the following features can be extracted from the matrix.

- **Contrast:** Measures the local variations in the gray-level co-occurrence matrix. It is high when local gray level is uniform and inverse GLCM is high. It is the difference between the highest and the lowest values of a contiguous set of pixels. It measures the amount of local variations present in the image. A low contrast image presents GLCM concentration term around the principal diagonal and features low spatial frequencies. GLCM contrast and homogeneity are strongly, but inversely, correlated in terms of equivalent distribution in the pixel pairs population. It means homogeneity decreases if contrast increases while energy is kept constant.

$$\sum_i \sum_j (i-j)^2 g_{ij}$$

- **Correlation:** Measures the joint probability occurrence of the specified pixel pairs. Correlation measures the linear dependency of grey levels of neighboring pixels. Digital Image Correlation is an optical method that employs tracking & image registration techniques for accurate 2D and 3D measurements of changes in images. This is often used to measure deformation, displacement, strain and optical flow, but it is widely applied in many areas of science and engineering

$$\frac{\sum_i \sum_j (ij) g_{ij} - \mu_x \mu_y}{\sigma_x \sigma_y}$$

- **Energy:** Also referred to as Angular Second Moment or uniformity. It is the sum of squares of entries in the GLCM. Angular Second Moment is high when image has very good homogeneity or when pixels are very similar. It measures the textural uniformity that is pixel pair repetitions and detects disorders in textures. Energy reaches a maximum value equal to one.

$$\sum_i \sum_j g_{ij}^{\ 2}$$

- **Entropy:** Entropy shows the amount of information of the image that is needed for the image compression. Entropy measures the loss of information or message in a transmitted signal and also measures the image information. Entropy is strongly, but inversely correlated to energy.

$$\sum_i \sum_j - g_{ij} \log(g_{ij})$$

- **Homogeneity:** Measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal. This statistic is also called as Inverse Difference Moment. It measures image homogeneity as it assumes larger values for smaller gray tone differences in pair elements. It is more sensitive to the presence of near diagonal elements in the GLCM. It has maximum value when all elements in the image are same.

$$\sum_i \sum_j \frac{1}{1+(i-j)^2} g_{ij}$$

The dimension of a GLCM is determined by the maximum gray value of the pixel. Number of gray levels is an important factor in GLCM computation. More levels would mean more accurate extracted textural information, with increased computational costs. The computational complexity of GLCM method is highly sensitive to the number of gray levels and is given by $O(n^2)$.

The general things to be kept in mind while GLCM is used in the selection of the textural features can be stated as follows:

- Energy is preferred to entropy as its values belong to normalized range.
- Contrast is associated with the average gray level difference between neighbor pixels. It is similar to variance however preferred due to reduced computational load and its effectiveness as a spatial frequency measure.
- Energy and contrast are the most significant parameters in terms of visual assessment and computational load to discriminate between different textural patterns.

So, we will take only energy, contrast, correlation and homogeneity and will not take entropy as feature.

2.4 Support Vector Machine

Support Vector Machine (SVM) is primarily a classier method that performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of different class labels. SVM supports both regression and classification tasks and can handle multiple continuous and categorical variables. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well. Support Vectors are simply the co-ordinates of individual observation.

Consider the example in Figure 2.1. Here there are many possible linear classifiers that can separate the data, but there is only one that maximizes the margin (maximizes the distance between it and the nearest data point of each class). This linear classifier is termed the optimal separating hyperplane.

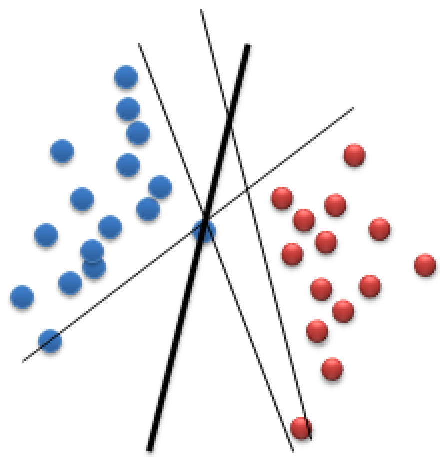


Figure 2.12: Bold black line best separates the two classes shown by colors

The set of vectors is said to be optimally separated by the hyperplane if it is separated without error and the distance between the closest vectors to the hyperplane is maximal. This incisive constraint on the parameterization is preferable to alternatives in simplifying the formulation of the problem. In words it states that the norm of the weight vector should be equal to the inverse of the distance, of the nearest point in the data set to the hyperplane.

In the case where a linear boundary is inappropriate the SVM can map the input vector, x, into a high dimensional feature space, z. By choosing a non-linear mapping a priori, the SVM constructs an optimal separating hyperplane in this higher dimensional space.

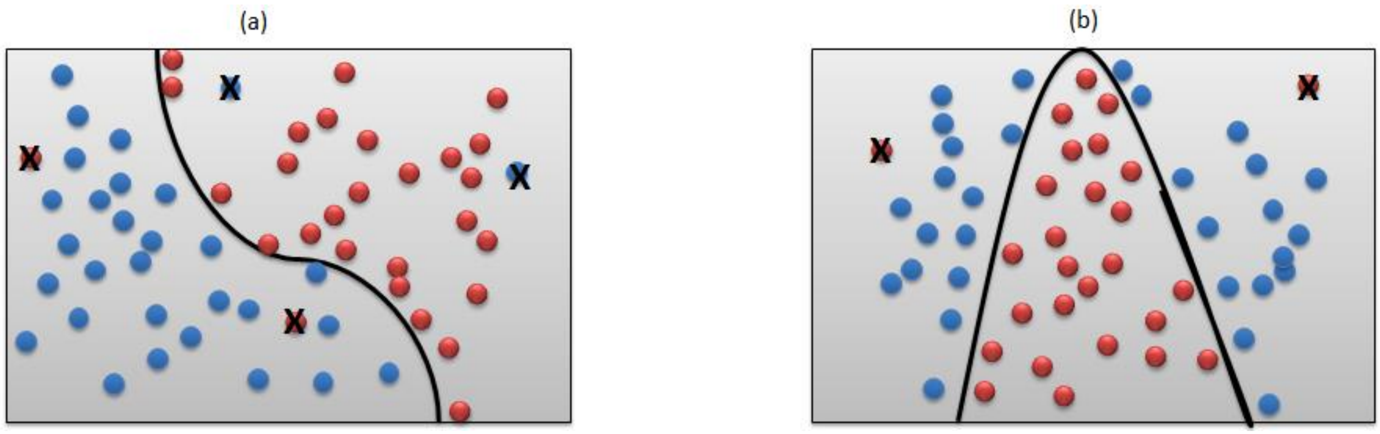


Figure 2.13: Non-linear Hyperplanes best separating two classes

To overcome the limitation of SVM we use the concept of kernels. Given a test point kernels will fit a curve on data points, specifically to points that are 'close' to the test set. Intuitively, a kernel is just a transformation of your input data that allows you (or an algorithm like SVMs) to treat/process it more easily. An RBF kernel, perhaps the mostly commonly used kernel, acts essentially as a low band pass filter that prefers smoother models. So if we are using an RBF kernel it will fit a Gaussian distribution over 'nearby' points of test point.

RBF kernel with the most commonly used Gaussian form is given by

$$K(x,x') = \exp(-\frac{\|x-x'\|^2}{2\sigma^2})$$

Polynomial kernel is given by $K(x,x') = (\langle x,x' \rangle \ +1)^d$

Classical techniques utilizing radial basis functions employ some method of determining a subset of centers. Typically a method of clustering is first employed to select a subset of centers. An attractive feature of the SVM is that this selection is implicit, with each support vectors contributing one local Gaussian function, centered at that data point.

2.5. Viola-Jones Algorithm

A face detector has to tell whether an image of arbitrary size contains a human face and if so, where it is. For our experiment, we need to detect the eyes and nose from the face and crop it so that those cropped images can be used as different sample image.

An operational algorithm must also work with a reasonable computational budget. Techniques such as integral image and attentional cascade make the Viola-Jones algorithm highly efficient: fed with a real time image sequence generated from a standard webcam, it performs well on a standard PC.

The Viola-Jones algorithm uses Haar-like features, that is, a scalar product between the image and some Haar-like templates. A simple rectangular Haar-like feature can be defined as the difference of the sum of pixels of areas inside the rectangle, which can be at any position and scale within the original image. This modified feature set is called 2-rectangle feature. Viola and Jones also defined 3-rectangle features (like (c) and (d)) and 4-rectangle features (like (e))). The values indicate certain characteristics of a particular area of the image. Each feature type can indicate the existence (or absence) of certain characteristics in the image, such as edges or changes in texture.

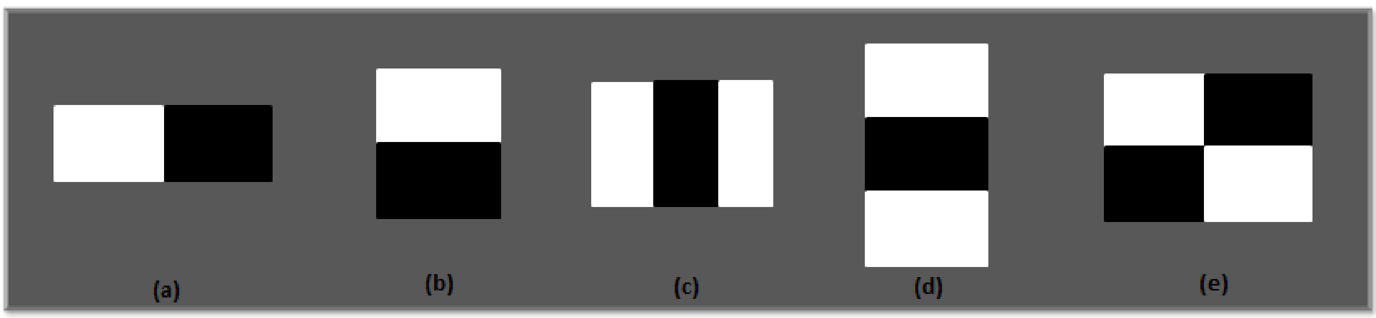


Figure 2.14: General Haar like patterns

Let I and P denote an image and a haar pattern, both of the same size N × N. The feature associated with pattern P of image I is defined by

$$\underbrace{\sum_{1\le i\le N}\sum_{1\le j\le N} I(i,j)1_{P(i,j)\text{ is white}}}_{\text{Sum of image pixels under white region}} - \underbrace{\sum_{1\le i\le N}\sum_{1\le j\le N} I(i,j)1_{P(i,j)\text{ is black}}}_{\text{Sum of image pixels under black region}}$$

Similarly, for the other haar like patterns, the features can be calculated by taking the difference of summed up pixel values under white and black region. Haar-like features consist of a class of local features that are calculated by subtracting the sum of a subregion of the feature from the sum of the remaining region of the feature.

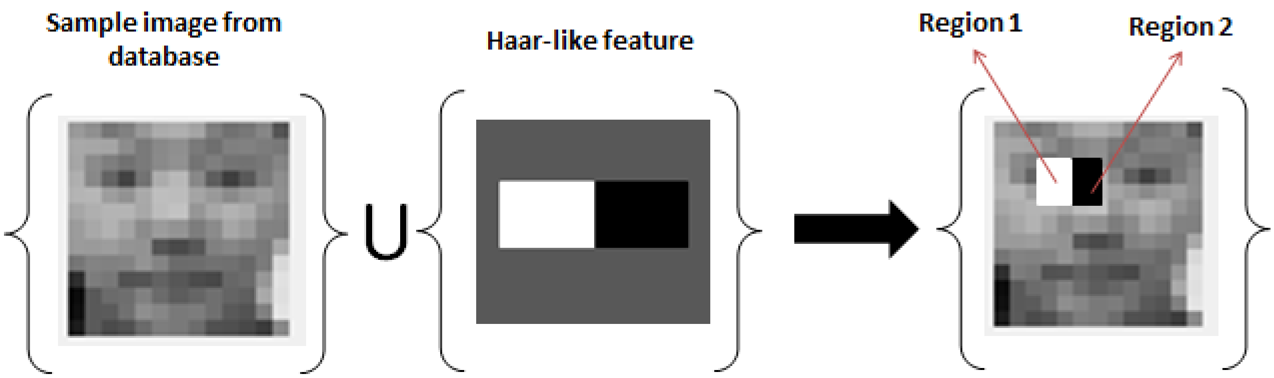


Figure 2.15: Haar like pattern applied on sample face image

The derived features are assumed to hold all the information needed to characterize a face. Since faces are by and large regular by nature, the use of Haar-like patterns seems justified. There is, however, another crucial element which lets this set of features take precedence: the integral image which allows calculating them at a very low computational cost. Instead of summing up all the pixels inside a rectangular window, this technique mirrors the use of cumulative distribution functions. The integral image X of image I can be calculated using recursion which is given by

$$X(i, j) = \begin{cases} \sum_{1 \le p \le i} \sum_{1 \le q \le j} I(p,q), & 1 \le i \le N \; and \; 1 \le j \le N \\ 0, & otherwise \end{cases}$$

Where,

$$\sum_{N_1 \le p \le N_2} \sum_{N_3 \le q \le N_4} I(p,q) = X(N_2, N_4) - X(N_2, N_3 - 1) - X(N_1 - 1, N_4) + X(N_1 - 1, N_3 - 1)$$

holds for all $N_1 \le N_2$ and $N_3 \le N_4$. As a result, computing an image's rectangular local sum requires at most four elementary operations given its integral image. Moreover, obtaining the integral image itself can be done in linear time: setting $N_1 = N_2$ and $N_3 = N_4$, we find

$$I(N_1, N_3) = X(N_2, N_4) - X(N_2, N_3 - 1) - X(N_1 - 1, N_4) + X(N_1 - 1, N_3 - 1)$$

Which lead us to the recursion algorithm. A computer program can be written to solve this recursion for higher values of N in Matlab or any other programming language.

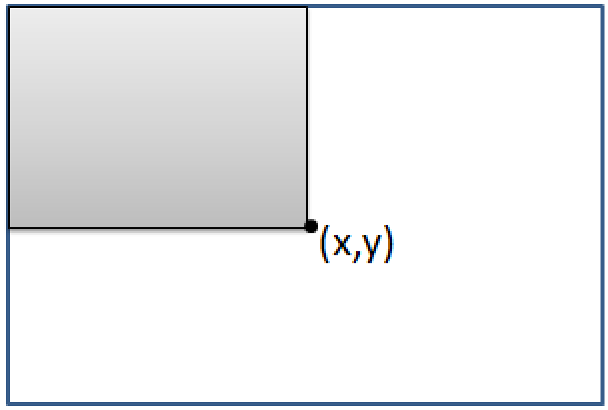


Figure 2.16: Integral image X(x,y) is given by sum of all the pixels in highlighted area

For illustration, consider the figure given below. The sum of the pixels within rectangle D can be computed with four array references: The value of the integral image at location 1 is the sum of the pixels in rectangle A. The value at location 2 is A + B, at location 3 is A + C, and at location 4 is A + B + C + D. The sum within D can be computed as 4 + 1 − (2 + 3).

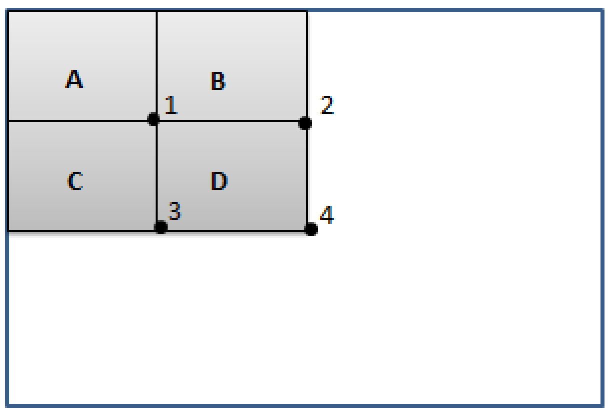


Figure 2.17: Integral image calculation for D

The building block of the Viola-Jones face detector is a decision stump, or a depth one decision tree, parameterized by a feature $f \in \{1, \cdot\cdot\cdot, d\}$, a threshold $t \in R$ and a toggle $T \in \{-1, 1\}$.The values above and below threshold will toggle the classifier value. Toggle T set carries two values as the given figure can be either a face (1) or not a face (-1). A very small number of these features can be combined to form an effective classifier. The main challenge is to find these features. A variant of AdaBoost is used both to select the features and to train the classifier. We will use cascade classifiers to speed up the process. Smaller, and therefore more efficient, classifiers can be constructed which reject many of the negative sub-windows while detecting almost all positive instances: simpler classifiers are used to reject the majority of sub-windows then, more complex classifiers are called upon to achieve low false positive rates.

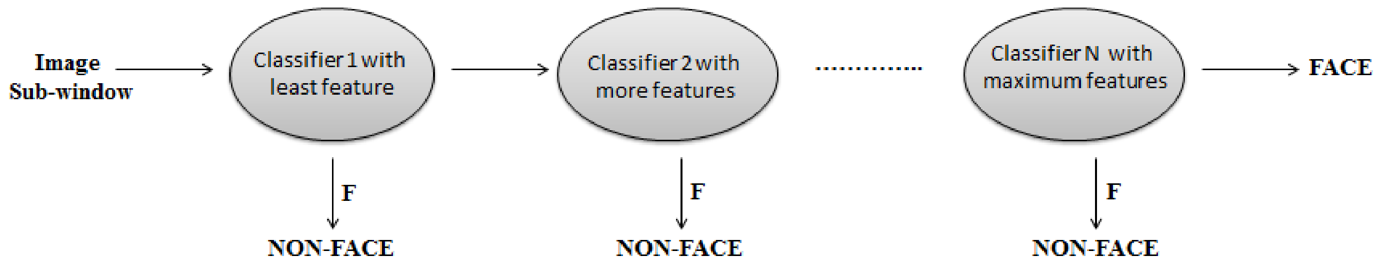


Figure 2.18: Cascade classifier with N weak classifiers

Appending a layer to the cascade means that the algorithm has learned to reject a few new negative patterns previously viewed as difficult, all the while keeping more or less the same positive training pool. To build the next layer, more negative examples are thus required to make the training process meaningful. To replace the detected negatives, we run the cascade on a large set of gray images with no human face and collect their false positive windows. The same procedure is used for constructing and replenishing the validation set.

The final result after passing through the N^{th} will determine if the given image is a face or non-face.

Chapter 3

Matlab Implementation

3.1 Objective

- To extract different features from facial images using various methods.
- To train the classifier used with those extracted features.
- To decrease the process time of feature extraction and training as much as possible by removing the redundancy from training set.
- To determine the accuracies of the methods using different parameters and maximize the accuracy percentage by trying various classifier kernels.

3.2 Database Used

- 45 Males and 45 Females
- 10 images per person were used
- 5 images per person used for training and remaining used for testing purpose
- Hence 225 male and 225 female images for Training
- Remaining 225 male and 225 female images for Testing
- 128x128 pixel image (BMP image)

Figure 3.1: Female Database

Figure 3.2: Male Database

3.3 Pre-processing of image

Images are cropped using Viola-Jones algorithm and they are resized so that the input image can be of same size for all images

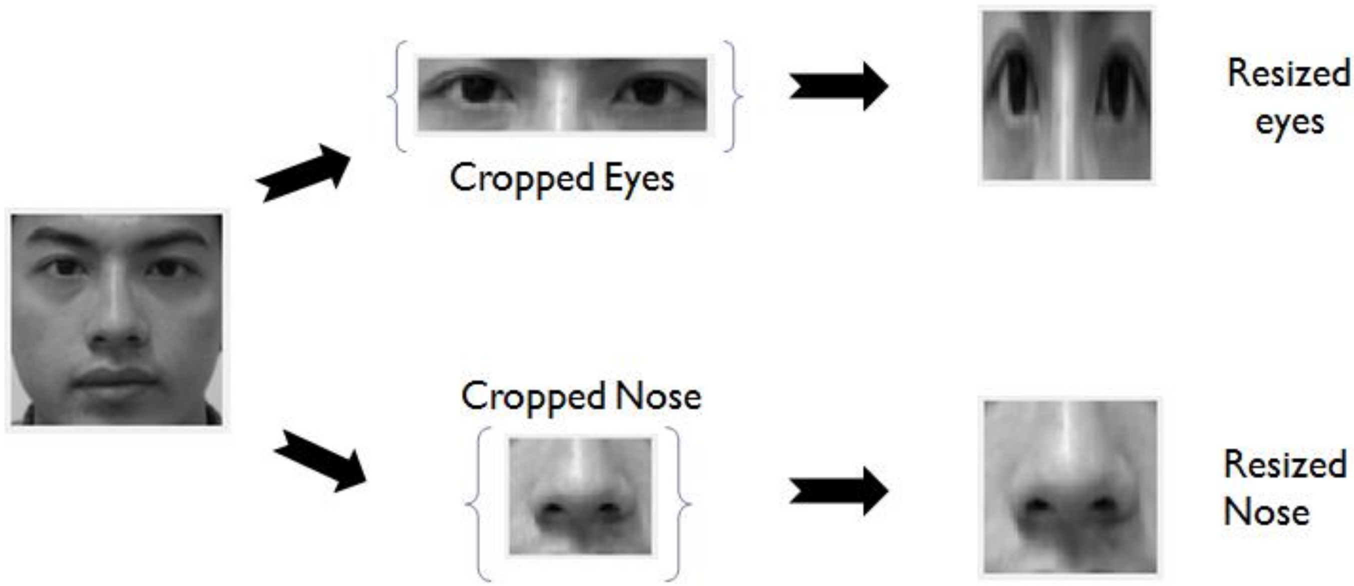
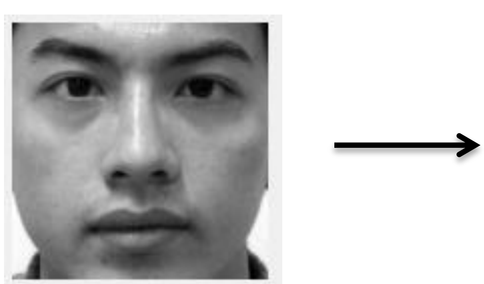


Figure 3.3: Pre-Processing of image

3.4 GLCM of sample image

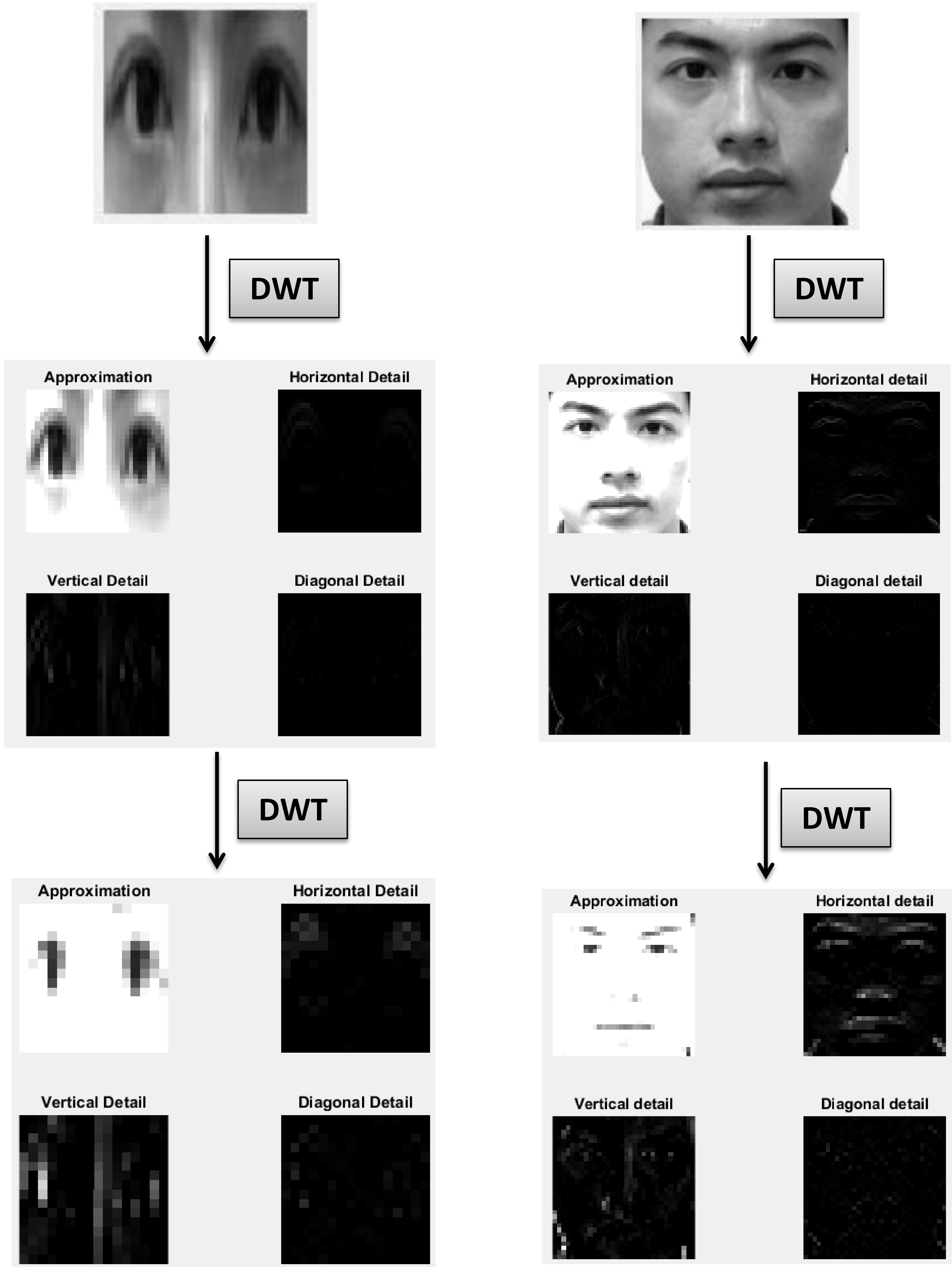
GLCM of sample face image from database with 8 gray levels.



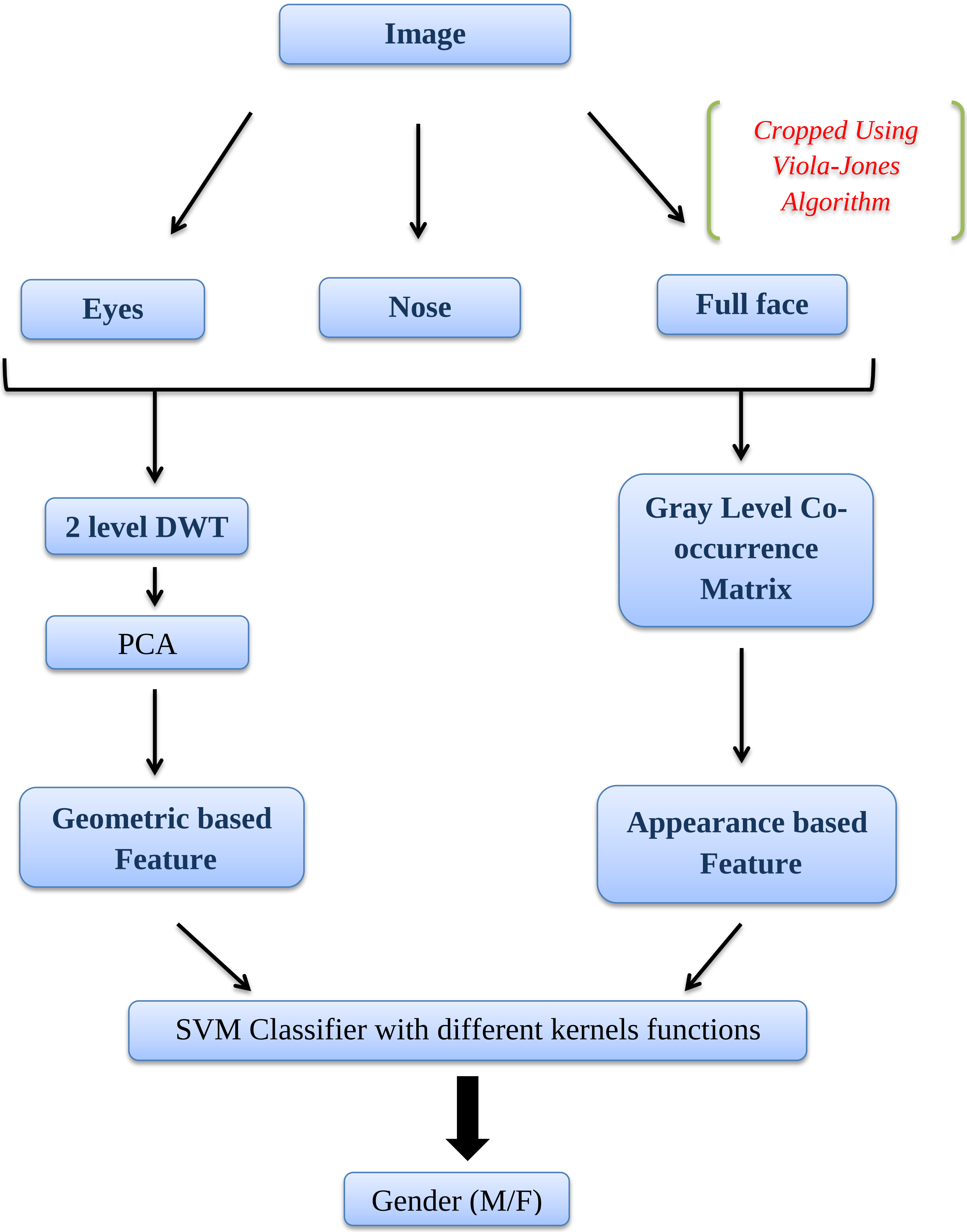
68	8	9	18	3	0	0	0
14	2	19	54	11	0	0	0
12	24	121	165	80	9	0	0
14	39	180	576	230	52	0	1
11	23	100	254	530	200	12	0
11	8	42	48	237	535	21	1
4	5	4	2	8	11	116	4
0	0	0	0	0	1	5	0

Figure 3.4: GLCM matrix of given image

3.4 2-Level DWT implementation of face and resized eyes



3.5 Flow Diagram



Chapter 4

Results and Observations

5.1 Using Appearance based feature

The GLCM matrix will be taken with both horizontal and vertical directions at three (5, 7, 10) different distances between the pixels. Also, three different SVM kernels were used and compared for the same set of features.

5.1.1 Horizontal direction

Adjacent pixels are compared horizontally with different distances. Distance D represent the D^{th} pixel to the right of the pixel compared.

- **Full Face**

Distance	RBF	Linear	Quadratic
5	79	70	70
7	71	64	67
10	78	67	72

- **Nose**

Distance	RBF	Linear	Quadratic
5	77	56	74
7	74	58	71
10	71	62	68

- **Eyes**

Distance	RBF	Linear	Quadratic
5	71	70	71
7	75	74	83
10	73	68	74

5.1.2 Vertical direction

Adjacent pixels are compared vertically with different distances. Distance D represent the D^{th} pixel to the bottom of the pixel compared.

- **Full face**

Distance	RBF	Linear	Quadratic
5	**64**	**59**	**63**
7	**63**	**60**	**61**
10	**66**	**52**	**63**

- **Nose**

Distance	RBF	Linear	Quadratic
5	**70**	**70**	**71**
7	**77**	**73**	**75**
10	**75**	**69**	**73**

- **Eyes**

Distance	RBF	Linear	Quadratic
5	**80**	**72**	**80**
7	**76**	**61**	**71**
10	**74**	**59**	**67**

RBF gave better results than other two kernel functions therefore we will use only RBF for remaining part of the project. Also, as it can be seen, for different parts, different distances give better results and therefore any general distance can't be assumed to give best result.

Eyes are giving better results in horizontal direction while nose is giving best results in vertical direction. Face is also giving better results in horizontal directions compared to vertical direction.

5.2 From Geometric based feature:

Training set images were passed through 2-level DWT and PCA is applied thereafter. PCA will provide us the feature images (eigenfaces) which will be passed through SVM classifier. Different numbers of features were taken and the results compared. Also, DWT is performed with 2 different wavelets and compared

- **Full Face**
  - **Using db4 wavelet**

No. of features Extracted	Accuracy
10	65
20	74
30	77
35	78

  - **Using db8 wavelet**

No. of features Extracted	Accuracy
10	69
20	76
30	78
35	79

- **Nose**
  - **Using db4 wavelet**

No. of features Extracted	Accuracy
10	68
20	74
30	78
35	80

- **Using db8 wavelet**

No. of features Extracted	Accuracy
10	69
20	76
30	79
35	81

- **Eyes**
  - **Using db4 wavelet**

No. of features Extracted	Accuracy
10	69
20	76
30	79
35	81

- **Using db8 wavelet**

No. of features Extracted	Accuracy
10	70
20	77
30	80
35	82

The results are better for db8 than compared to db4 wavelet for all three facial images. As number of feature extracted are increased, the accuracy is increased and after 30 features, the change in accuracy percentage is not significant. Hence the remaining features can be avoided to decrease the process time of training the SVM classifier.

Accuracy achieved order in this experiment : Eyes > Nose > Face .

Chapter 5

Conclusion

- The accuracy obtained is highest for eyes and lowest for full face because eyes gives us the more detailed (local) feature as we are looking at smaller level while face gives the global feature as a whole face which is at bigger level. Although the feature obtained from nose are also local features but features obtained from eyes are more detailed as more edges are present in eye image. Therefore, we can conclude that eyes carries more information about gender than nose.

- RBF kernel function gives better results than other kernel functions in SVM while performing face recognition because separation of features on linear and quadratic hyperplane is too difficult thus reducing the accuracy. In fact, RBF kernel is the most popular kernel function in machine learning.

- Discrete Wavelet Transform using db8 gives more accuracy than db4 wavelet when applied to face images for gender classification.

- Accuracies obtained from geometric based features are more compared to appearance based features as texture of face image (emotion) varies hugely compared to geometry (shape, size) of the facial parts.

Future Work

- These both methods can be implemented together to get a more accurate gender analysis.
- Much more accurate SVM kernel functions can be used for better classification of two classes.
- Other classifiers can also be implemented here and compared.
- Same methods can be applied on other facial parts like lips and mouth and results can be obtained for those images in similar manner
- These methods can be tested on different and vast databases to generalize the results obtained here.

References

1. H. F. Alrashed ,M. A. Berbar.," Facial gender recognition using Eyes images", 2013.
2. S. A. Khan, M. Nazir, N. Naveed, N. Riaz "Efficient gender classification methodology using DWT and PCA", 2012.
3. O. Ozbudak, M. Tukel, S. Seker.," Fast gender classification" 2013.
4. E. Makinen, R. Raisamo.," Evaluation of Gender Classification Methods with Automatically Detected and Aligned Faces" 2008.
5. A Majumder, L. Behera,V. K. Subramanian, "Automatic and Robust Detection of Facial Features in Frontal Face Images".
6. M. M. Hasan, M. F Hossain.,"Facial Features Detection in Color Images Based on Skin Color Segmentation", 2014.
7. T. Lee, S.K. Park., 2005. "A New Facial Features and Face Detection Method for Human-Robot Interaction".
8. S. Zafeiriou, A. Tefas, I. Pitas, "Gender Determination using Support Vector Machine Variant", 2008.
9. A. H. Bishak, Z. Ghandriz, T. Taheri,"Face Recognition using Co-occurrence Matrix of Local Average Binary Pattern (CMLABP)", 2012.
10. R. C. Gonzalez, R. E. Wood (2013), Digital Image Processing, (4^{th} ed.)