Unsupervised face analysis from multi-view

Seyed Mohammad Hassan Anvar

2014

Seyed Mohammad Hassan Anvar. (2013). Unsupervised face analysis from multi-view. Doctoral thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/59221

https://doi.org/10.32657/10356/59221

Unsupervised Face Analysis from Multi-View

Seyed Mohammad Hassan ANVAR

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

2013

Acknowledgements

In the name of GOD the most compassionate the most merciful.

I would like to express my great honour and sincere gratitude to my supervisors Dr. Yau Wei Yun and Associate Professor Teoh Eam Khwang who have supported me with their guidance, suggestions and valuable comments. Without their assistance and direction, this research would not be possible. They have encouraged me enthusiastically and patiently during my entire PhD research project.

I am also sincerely and deeply grateful to Dr. Karthik Nandakumar, Dr. Li Jun and all the staff members and friends in Computer Vision Lab at the School of Electrical and Electronic Engineering, Nanyang Technological University and Institute for Infocomm Research (I^{2}R) for their kind support and help rendered.

I want to give my special thank and honour to my father, my mother, my brothers and sisters who have incessantly supported me with passion and encouragement in my life and during this research.

Finally, I would like to acknowledge the Agency for Science, Technology And Research (A*STAR) Singapore whom has awarded me the scholarship to pursue my PhD and has supported my research and also the School of Electrical and Electronic Engineering, Nanyang Technological University, which has accepted me as a PhD candidate and provided me with the research facilities and conducive environment for my research work.

Contents

List of Figures

List of Tables

List of Symbols

$\|.\|$	The L2-norm of a vector
${}^{\partial}/_{\partial\mu_x}$	Differentiation based on μ_x
f^A	The appearance descriptor of a feature
f^G	The Geometry information of a feature
$\int_Z \quad dz$	Integration over the region Z
$\Gamma(\,)$	Gamma function
$\hat{\theta}^{MLE}$	Probability parameter θ estimated using Maximum Likelihood Estimation
$p(\,)$	Probability function
$Area(\,)$	Area of the face

Summary

Face detection, localization and recognition from multi-poses are one of the most challenging topics in the area of computer vision and pattern recognition. During the past decade several methods have been proposed for face detection and recognition but they mainly concentrate on frontal faces. Although in recent years, some methods have been proposed based on sliding windows to exhaustively search through the image in many scales and views, they are slow and their performance on multi-view or multi-pose faces are not as good as frontal face detections. Training these methods is also a bottleneck. Most of the proposed methods require thousands of positive and negative samples from different face poses for training. These images for training have to be manually cropped, aligned and labeled, thus is manually intensive and expensive to perform. Note that since the effect of variation in the face pose captured on a fix camera and change in camera view but with a fixed face pose is the same, we used both words interchangeably,

In this research project, a probabilistic approach is proposed for face detection, localization and identification from multi-views. The proposed method does not supervision during the training stage. The main focus of the project is on multi-view analysis as well as unsupervised or automated learning. The proposed approach provides a unified framework to learn a multi-view model for face detection, localization and identification.

For face detection and localization from multi poses, given the images of different people in multi-poses, it obtains a multi-view face model obtained through a constellation of corresponding face features in the training set. The obtained model is pruned such that only the most distinctive features are retained. The model is then used to detect and localize multiple face images in multi-poses. Two versions of model construction are proposed for face detection and localization. One requires manually labeling only two control points in one image of the training set regardless of the number of images in the training set. The other version is completely automatic. Even the control points are estimated automatically and the system only requires some training images that include face of different people from multi-views. The trade-off is that the accuracy is slightly lower compared to the manually labeled approach.

Another effort is face identification and localization from multi poses. The proposed method is completely automatic. The data collection is also performed by the system using web images obtained from web search. It requires only the textual name of the queried person, which is used as input to the search engine. Using the images returned by the search engine, the system classifies these images and constructs a multi-view face model for the query. With the model, the system is then able to identify the query candidate in his/her digital photo album or gallery. The user does not have to manually tag the images or provide images to train the face recognition module. Since system is completely automatic, it has widely application including search for people in news video or tagging actors and actresses in movies or targeted advertisement for interactive televisions (IPTV).

.

Chapter 1

Introduction

1.1 Motivation

Emerging new media technologies such as internet protocol televisions (IPTV), digital photo albums and social network image services require a more general face detection, localization and identification method which is able to handle the rotated, tilted and occluded faces. As such, face detection and identification from multi-poses have attracted the attentions of many researchers, making it a hot research topic in the area of the computer vision and pattern recognition. Consider the scenario where you are watching a movie, sports or news and would like to know who is the person in the content or a digital photo album with many images, which you would like to label names in the presented pictures automatically or to search for someone in a photo collection on the web and wonder is there any way to do these jobs automatically? Recently, there are several valuable studies on face detection from multi-views [1-3]. However, almost all the efforts require a large number of training images to be labeled and the faces in them cropped manually. There are also some reported studies on face identification in the media [4-10]. However, all the approaches assumed the availability of face detection and alignment. They mainly focus their contributions on the face identification and classification stages. Such cascading of various modules have the advantage of simplicity but when some stages are put together in the cascaded form, the overall performance is limited to the weakest stage in the framework. For instance, if the face detector performs poorly in handling the rotated faces or faces in multi-pose, the framework will not be able to deal with these types of images even though the face classification can support faces in multi-pose for example.

1.2 Objectives

The objective of this research is to develop a framework for multi-view face detection and identification. To achieve this goal, first we have developed a statistical framework for face detection and localization from multiple views. Then, we extended our proposed framework to handle face identification from multiple views.

1.2.1 Developing an Framework for Face Detection and Localization from Multi-Poses

The current conventional approaches for face detection and localization use a sliding window with predefined feature such as Haar, Gabor, Eigenface etc to detect face images in the given images. To train these methods, a huge number of face and non-face images must be gathered, labeled and cropped manually. The other disadvantage with the current methods is that they yield a coarse detection as they only indicate a window region which may include a face. For further processing, it is necessary to apply the face alignment techniques and then extract the face features from the detected region.

In this research work we proposed a novel framework which minimizes the manual intervention in the training stage and is able to simultaneously detect and localize the face in a give image. It can locate the face area in a more accurate manner without requiring an additional face alignment stage.

1.2.2 An Automatic Single Framework for Face Identification from Multi-Views

Another challenge is to train a model for face from multi-poses to support face recognition. The current conventional approaches cascade some previous algorithms such as Viola-Jones [1] to detect the face region. Then they use face alignment algorithms to specify the face region and crop the images to the facial area. Once the face area and facial features are extracted, the identification algorithm is applied. Although the reported performance for each stage is very high individually, the overall performance is still limited by the limitation of each stage and the performance of the weakest stage used. For example, most face recognition uses the available face detector. Even though the face recognition can support non-frontal faces, the overall system will not work if the face detector cannot support non-frontal face detection.

In this thesis, we proposed a method which fuses different stages of face identification in a single and integrated framework that can support multi-view face recognition. Our proposed method is also fully automatic and does not require any supervision. It uses the ordinary images from the web obtained from the query result produced by a web search engine and then develops a multi-view model of the person.

1.3 Original Contributions of the Thesis

The followings are the original contributions for the thesis:

1. A matching technique is proposed which is able to find the most number of real correspondence points between the comparing images of different views. The main advantage of the proposed method is that it is able to despite the real correspondences are much lower than the wrong correspondences.
2. A constellation connection is introduced which is used to make connection between face images of multi-views and register them to a reference map. We showed how faces of multi-poses find their ways to link to a frontal face image through this constellation connection. Once the connection between images of multi-views has been established, a multi-view face model from the projection of 3D into 2D space is built and for each feature in the model a probability value is determined. Using the probability, the redundant features are discarded from the model. A clustering method is proposed to remove redundant features

from the model as much as possible. Consequently, face detection is fast due to the small distinctive model.

3. A multi-view face detection and localization is proposed which makes use of the obtained model. The method is general and is able to detect multiple faces in a single image. It is also able to detect the in-plane rotated faces and estimate a rotation angle for each face individually. The detector is able to tolerate substantial occlusion and scale variations.
4. A method is proposed to totally remove the need for manual labeling from the training stage and the model construction stage. The automatic model construction is then used to train the faces of different people from multi-views and employs the obtained model to identify them in their digital photo albums.
5. The ordinary images from the web are used to train the classifier. Unlike the conventional methods which require a clear image to train their models, our proposed approach uses the images obtained from the web search engines, which could have partial faces, pose and scale variations as well as wrong images, to develop a multi-view model for face recognition.
6. The proposed approach for face detection and localization has a unique form which performs face detection, alignment and identification in a single unified framework and done in a single continuous stage. Consequently, the overall performance of the method would not be affected by the limitation of any other third party algorithms for detection or alignment.

The summary of contributions and the related publications are tabulated in Table 1.1.

Table 1.1. Contributions achieved in this project and the papers published based on these contributions.

Contribution	Paper Title	Place of Issue
Matching Technique	Finding the Correspondence Points in Images of Multi-Views	*Proc. of* IEEE SITIS 2012
Model Shrinking and Enhancing the Face Detection Speed	Fast Face Detection and Localization from Multi-Views Using Statistical Approach	*Proc. of* IEEE ICICS 2011
In-Plane Rotation Angle Estimation	Estimating In-Plane Rotation Angle for Face Images from Multi-Poses	*Proc. of* IEEE CIBIM 2013
1. Matching Technique 2. Constellation Connection 3. Multi-View Face Detection and Localization	Multi-View Face Detection and Registration Requiring Minimal Manual Intervention	*Trans. on* IEEE PAMI 2013
1. Removing the Manual Labeling 2. Using the Ordinary Images from the Web 3. Face Identification from Multi-Views	An Automatic Multi-View Model Learning for Character Identification and Face Tagging in Digital Photo Albums	*Submitted to Trans. on* IEEE TCSVT

1.4 Organization of the Thesis

The organization of the thesis is as follows:

Chapter 2 reviews the tools and materials required to develop our algorithms. It also investigates the previous related works and significant researches to the area of our project. In the beginning feature extraction and selection are introduced and probabilistic functions for feature analysis are investigated which are the important and basic tools in the area of image processing and pattern analysis. The chapter proceeds by introducing the state of the art matching algorithms and face detection and localization methods from multi poses. Finally the state of the art face identification applied to multimedia and web-based learning is investigated.

Chapter 3 describes our proposed algorithm for finding the reliable correspondence points between comparing images. It starts with feature selection and develops a method for refining the potential correspondence points between image pairs. The proposed method finds the maximum number of true correspondence points among images of different views.

Chapter 4 illustrates the face registration from multi-views. It describes the proposed constellation approach which connects the faces of multi-poses to resister to a predefined reference map. Once all the possible faces have been registered to the reference map, a multi-view face model from the projection of 3D into 2D space is formed and redundant features are discarded from the model using the probability values of the features in the model.

Chapter 5 uses the constructed multi-view face model in chapter 4 to detect and localize multi-view face images in general images. It also describes the experiments conducted which validate the ability of the proposed probabilistic classifier to detect and localize occluded and in-plane rotated faces in the given images and also estimates the rotation angle for the detected faces. The result of the proposed face detection framework is also compared to the state of the art face detection methods.

Chapter 6 extends the approach to a totally automatic approach for training a multi-view model and then uses the model to detect and identify people in their digital photo album. Experiments conducted are described to show that the proposed approach is able to use the ordinary images from the web without any supervision and enhances a multi-view model for the query.

Chapter 7 represents the conclusion and some suggestion for the future work.

Chapter 2

A Literature Review

2.1 Introduction

In this chapter, some important works and tools for face detection, localization and recognition from multi-poses are investigated. Then, the state of the art methods on face detection and localization from different views are described in detail. Finally, we investigated the state-of-the-art in character identification and face tagging from multimedia, including the web.

2.2 Keypoints and Features

Objects are usually recognized by their special characteristics or features. Up to now, several different feature extraction methods have been proposed in the literature. Some methods compute the attributes for each pixel in the entire image such as the correlation [11], mutual information [12] and Fourier [13] methods. However, these methods are time consuming and in general are not able to handle scale, distortion and pose variations. Some other methods determine the salient points in the images which decrease the processing time significantly. Among them the Scale Invariant Feature Transform (SIFT) introduced by Lowe [14] is the most popular. It has been shown [15] that this feature extraction method could tolerate wide range of distortion and viewpoint changes. They are highly distinctive and can provide robust matching between different views of an object. Because of its popularity, several extensions such as GLOH [15], SURF[16] and SCARF [17] have been introduced recently.

2.2.1 Scale Invariant Feature Transform

Lowe [14] proposed a method to extract features from an image which are invariant to scale and rotation. There are four major stages to extract these features. The first stage is finding keypoints in the image. All possible scales and the entire image is searched for distinctive points using the difference of Gaussian (DOG) operation. Input image is convolved as in (2.1) with different smoothing levels (σ) of a Gaussian function (2.2), and then the two nearby smoothed images are subtracted using (2.3) to obtain the difference of Gaussian image.

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.1}$$

where $I(x, y)$ is the input image and "$*$" is the convolution operation.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \; e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{2.2}$$

where σ is the smoothing level of Gaussian function.

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma) \tag{2.3}$$

where k is the multiplicative constant for different smoothing levels and $D(x, y, \sigma)$ is the difference of Gaussian image. Every smoothed image is down sampled by half to produce the Difference of Gaussian (DOG) images as shown in Figure 2.1. This procedure is repeated until no down sampling is possible.

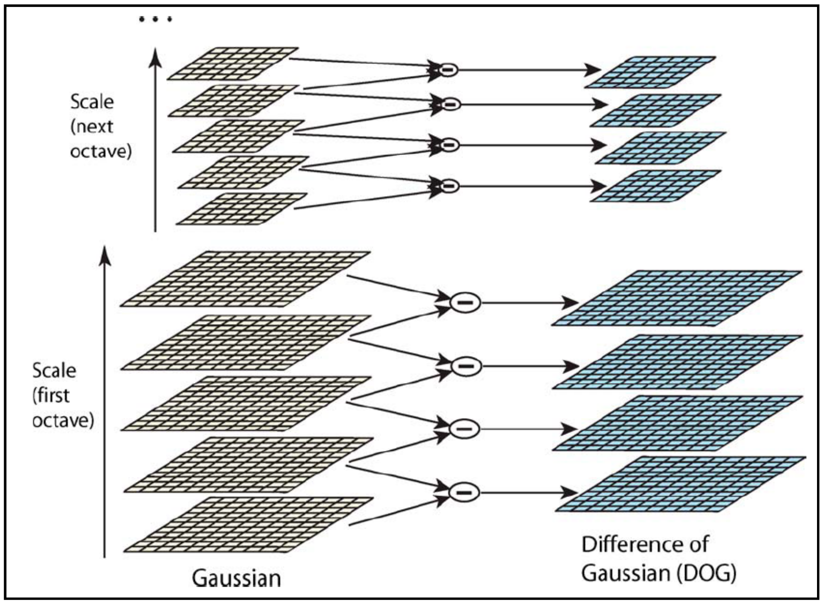


Figure 2.1. Shows the convolution between input image and the Gaussian filters in different smoothing levels. The filtered images are downsampled to lower scales until no downsampling is possible. The difference between neighbor levels shapes the Difference of Gaussian layers. (Extracted from [14])

Potential keypoints are those points placed in the local maxima and minima of the DOG images. These candidates are detected by comparing the eight neighbours of each pixel in the current, above and below images as shown in Figure 2.2.

In the second stage, the algorithm fits a model to each potential keypoint in order to identify the location and scale of the features. This information can help to reject low contrast points. Using up to the quadratic terms of Taylor expansion (2.4) around each point for the scale-space function $D(x, y, \sigma)$, the location of the extremum (2.6) is

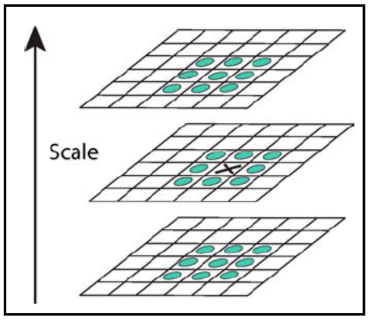


Figure 2.2. Depicts the local minima and maxima points obtained by comparing each pixel with its eight neighbour pixels in the current, above and below levels. (Extracted from **[14]**)

determined by taking the derivative on (2.4). Substituting (2.5) into (2.4) gives the function value at the extremum (2.6).

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \tag{2.4}$$

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} \tag{2.5}$$

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x} \tag{2.6}$$

Each point with the extremum value $D(\hat{x})$ magnitude less than a threshold (we used 0.03 in all our experiments) were rejected as it is considered low contrast point.

Since the DOG algorithm has significant response to the edge points, it is necessary to eliminate edge points for stability. Those points which have a large principal value through the edge but a small in the perpendicular direction are removed. Hessian matrix (2.7) is used to determine the principal curvature values at keypoints in the different locations and scales. Computational complexity could be reduced by computing only a ratio of trace and determinant (2.8) of this matrix and reject the points which are less than a threshold (2.8).

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{2.7}$$

$$\frac{Trace(H)^2}{Det(H)} < \frac{(r+1)^2}{r} \tag{2.8}$$

Once stable keypoints are determined, a consistent orientation is assigned to them based on image properties in the vicinity region of keypoints. The gradient magnitude

(2.9) and orientation value (2.10) in the Gaussian smoothed image are calculated within a region around the keypoints to shape the orientation histogram of 36 bins for 360 degree range of orientations.

$$m(x,y) = \sqrt{(\ L(x+1,y) - L(x-1,y)\ )^2 + (\ L(x,y+1) - L(x,y-1)\ )^2} \tag{2.9}$$

$$\theta(x,y) = \arctan\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right) \tag{2.10}$$

Peak values in this histogram show the dominant direction of local gradient. Each peak value greater than 80 percent of maximum peak value is considered as individual keypoint and direction. In this way, it is able to determine more than one feature in each keypoint location with different orientations and scales. Following the above procedure gives stable keypoints such that if the picture is rotated or resized, the keypoint location, scale and orientation remain unchanged. The area around the keypoints are segmented into 4×4 sub regions and feature descriptors are formed by computing the gradient orientation histograms within each sub region in 8 directions. To obtain a more orientation invariant descriptor, the coordinate of the descriptor is rotated with respect to the keypoint’s global orientation.

2.2.2 Evaluation of Features

A comprehensive survey have been conducted by Mikolajczyk and Schmid [15] to evaluate the performance of the different interest region detection and local feature extraction methods in various scenarios. Their benchmark test included the rotated, zoomed and rotated, viewpoint changed, blurred, JPEG compressed, and changed illumination conditions. They concluded that Scale Invariant Features Transform (SIFT) [14] and Gradient Location and Orientation Histogram features (GLOH) [15], which is an extension to the SIFT features, have better performance. They also have

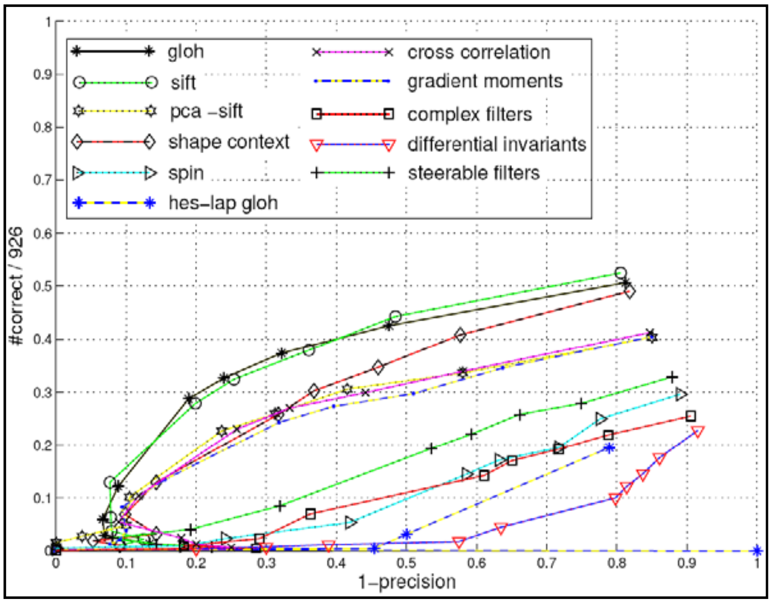


Figure 2.3. Compares the performance of different features used to find the correspondence points between images in various scenarios. (Extracted from **[15]**)

showed that SIFT feature with nearest neighbor matching strategy is more robust and distinctive among the region-based descriptors such as GLOH, SIFT, PCA-SIFT [18], shape context [19], spin images [20], cross correlation, gradient moments [21], complex filters [22], differential invariants [23] and steerable filters [24]. Figure 2.3 illustrates the comparison between these features.

Based on the above comparison, we chose the original SIFT feature in all our experiments. Our evaluations also showed that the SIFT is suitable meet our project’s needs. However, our proposed approach was not built using the foundation of SIFT but using it only as a feature. Thus, our approach is general and other scale invariant features could be used.

2.2.3 Feature Selection from Objects

Feature selection is very important in object detection and recognition. Selecting proper features helps to improve the performance and speed of the algorithm. Usually many SIFT features are found from different parts of an image to ensure that the desired features are found. However, it is very important that only those features which come from the desired objects are selected and to reject those from the background or due to noise. For instance, after extracting the features for the image in Figure 2.4, it is found that the two features shown in Figure 2.4(a) have a similar appearance but one of them is part of the desired object and the other is from the background. This implies that the appearance of the mentioned feature is not fully discriminative to represent the object.

Therefore, a mechanism must be considered to select only those features which represent the object among all the extracted features from the image as shown in Figure 2.4(b). For this purpose a three step approach introduced by Dorko and Schmid [25] is adopted. Keypoints and feature descriptors are found in all the images which contain the object using the method similar to the Lowe's method [26]. But instead of Difference of Gaussian interest point detector, the Harris-Laplace detectors [27] is used. This detector computes the Harris interest points in different scales and selects points in local maxima. After detecting the keypoints, and assigning descriptors to them, they are clustered using the K-means algorithm with 300 to 600 number of clusters. These clusters are then used to learn a face model using the linear Support Vector Machine (SVM) or Gaussian Mixture Model (GMM). Object area must then be marked manually in the training images so as to separate the keypoints obtained into positive and negative groups. In SVM, the labeled keypoints and cluster centers of the objects are then further processed using likelihood ratio or mutual information to determine the discriminative features among the other features of the objects. In GMM, Expectation Maximization (EM) algorithm is initialized using the cluster centers to estimate the means, covariance and weights of the objects that satisfy a Gaussian mixture model. The discriminative level of the features of the objects are determined using the likelihood ratio or mutual information.

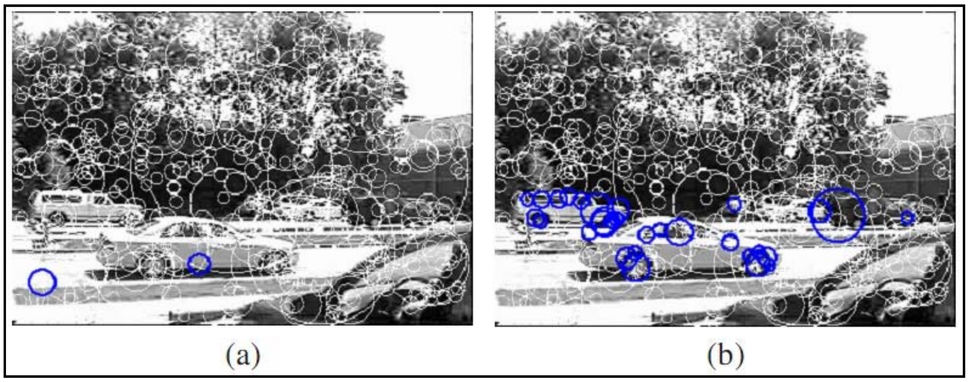


Figure 2.4. Shows the extracted features for a sample image. Two indicated features in (a) have the similar appearance but one from the object part and the other from the background clutter. It is desirable to use a method to select only those features which come from the distinctive parts of object (b). (Extracted from [25])

In the likelihood ratio method, the probability of correctly classifying the descriptor as part of the desired object to non object is used as a criterion to eliminate or keep an object part. An object part C_i is considered as a discriminative part of the object if the likelihood ratio calculated for this part is greater than a threshold. The likelihood ratio determines the probability a cluster center comes from the object to the probability that a cluster center comes from the non-object parts as given by (2.11).

$$L(C_i) = \frac{P(C_i = 1|Object)}{P\left(C_i = 1\middle|\overline{Object}\right)}, \tag{2.11}$$

In the mutual information method, the mutual information between the parts which are correctly classified and those which are wrongly classified, are calculated using equation (2.12). The parts with mutual information greater than a threshold are considered as the discriminative parts of the object.

$$I(C_i\,, Object) =$$

$$\sum_{c\epsilon\{0,1\}} \sum_{o\epsilon\{0,1\}} P(C_i = c, Object = o)\, log \frac{P(C_i = c, Object = o)}{P(C_i = c)P(Object = o)} \tag{2.12}$$

2.3 Feature Analysis Using Probability Functions

Many factors could affect the image quality such as illumination variation, noise, motion blur and image resolution. Probability functions are popularly used to help characterize the image in the presence of noise and clutter. The following subsections briefly review the posterior probability and likelihood ratio test used in our work. These two functions are also widely used by many researchers in the area of image processing [28, 29].

2.3.1 General form of Posterior Probability

The general form of posterior probability function which is commonly used in the object recognition methods is given in (2.13).

$$P(object|image) = P\left(object\middle|pixel(1,1), pixel(1,2), \dots, pixel(n,m)\right) \tag{2.13}$$

This equation is a conditional probability function which determines the probability of an existing object of interest in the given image. $pixel(i,j)$ denotes the intensity value for the pixel at location (i,j) and can be considered as either grey level or color value.

In many applications, it is not suitable to employ pixel values. Hence, it is better to change the probability function in (2.13) to the more general form in (2.14).

$$P(object|image) = P(object|feature1, feature2, \dots, featureN), \tag{2.14}$$

In this equation, $featureN$ may be considered as the intensity value of a pixel or a region of pixels. Alternatively, it can also represent image features such as gradient, intensity difference and cosine and sine coefficients.

Such probability function is called posterior which means it could estimate the probability with the given condition. However, computing it is not feasible. Using Bayesian theory, the posterior function can be changed to prior probability and likelihood as given by (2.15) which is suitable for computation as it is possible to evaluate the prior probability and likelihood terms based on previous observations.

$$P(object|image) = \frac{P(image|object)P(object)}{P(image)}, \qquad (2.15)$$

Substituting (2.14) into (2.15) gives the general forms in (2.16).

$$\begin{aligned} &P(object|image) \\ &= P(object|feature1, feature2, \dots, featureN)P(object|image) \\ &= \frac{P(feature1, feature2, \dots, featureN|object)P(object)}{P(feature1, feature2, \dots, featureN)} \end{aligned} \qquad (2.16)$$

To further simplify the above equation, it is common to assume that all the features are independent from one another, resulting in (2.17).

$$\begin{aligned} &P(feature_1, feature_2, \dots, feature_N|object) \\ &= P(feature_1|object).P(feature_2|object)\dots P(feature_N|object) \\ &= \prod_{j=1}^{N} P\left(feature_j|object\right), \end{aligned} \qquad (2.17)$$

Thus the equation (2.16) is simplified to the form of (2.18) as:

$$P(object|image) = \prod_{j=1}^{N} \frac{P\left(feature_j\middle|object\right).P(object)}{P(feature_j)} \tag{2.18}$$

To determine the posterior function in (2.18), past observations are used. It means that some training information is needed to calculate the prior probability function and likelihood term. $P(feature_j)$ and $P(object)$ are prior probabilities of occurrence of $feature_j$ and object of interest in the training set. $P\left(feature_j\middle|object\right)$ is called the likelihood term and is usually calculated by counting the number of occurrence of $feature_j$ in the training images that contain the object of interest.

2.3.2 Likelihood Ratio

Likelihood ratio function is commonly used to determine whether there is an object occurring in an image or not. Considering the possibility that some features represent the non-object area or noise and background clutter, their prior probability is given in (2.19).

$$\begin{aligned} &P\left(feature_1, feature_2, \dots, feature_N\middle|\overline{object}\right) \\ &= P\left(feature_1\middle|\overline{object}\right).P\left(feature_2\middle|\overline{object}\right)\dots P\left(feature_N\middle|\overline{object}\right) \\ &= \prod_{j=1}^{N} P\big(feature_j)\big|\overline{object}\big) \end{aligned} \tag{2.19}$$

where $\overline{object}$ indicates non-object. Thus, we can define the likelihood ratio function using the combination of two prior probability functions given in (2.17) and (2.19) as demonstrated in (2.20).

$$Likelihood\ Ratio = \prod_{k=1}^{N} \frac{P(feature_k)|object)}{P(feature_k)|\overline{object})} \tag{2.20}$$

$P(feature_k)|\overline{object})$ is calculated in the same way as $P(feature_k)|object)$ but instead of counting the number of occurrence of $feature_k$ in the training images which contain the object of interest, the number of occurrence of $feature_k$ in the images which do not contain the object of interest is counted. This usually happens when noise disrupt the results or the selected feature is not discriminative and response to both objects and non-objects.

2.4 Matching Techniques

Matching techniques are algorithms that search two images to find a set of similar points in both images. The pair of similar points in both images are called the correspondence points

2.4.1 Finding the Correspondence Points Using Scale Invariant Features

Lowe [14] proposed a matching method based on scale invariant features. In this method, the keypoints in two comparing images are found using the Difference of Gaussian in multi scales. Orientation is assigned to each keypoint based on the gradient of its neighboring points. Then the descriptor is computed for each keypoint using gradient histogram at specific orientations in a sub region whose size corresponds to the feature’s scale. The keypoint descriptors of the first image are compared to the keypoint descriptors of the second image by computing the Euclidean distance between the descriptors. The best match candidate for each keypoint in the first image is obtained by finding the keypoint in the second image whose descriptor has the minimum Euclidean distance to the descriptor in the first image. However, such matching process does not ensure that the matching is always correct. Therefore, a method is needed to discard the keypoints which are not reliable or wrong. In addition, the distinctiveness of each feature may differ and thus using a global threshold of the Euclidean distance to reject the less distinctive features may not produce good results as the some genuine matches may be removed. Lowe proposed to compare the distance of the closest match to the distance of the second closest match as a measure for the distinctive feature. He inferred that if the first match is the correct and distinctive match, then its distance should be significantly smaller compared to the second closest match which should be considered as the wrong match. In another word, the second closest match can provide a measurement to determine the level of ambiguity of a feature.

2.4.2 Matching Refinement Using RANSAC Method

Fischler and Bolles [30] introduced the Random Sample Consensus (RANSAC) method for finding the true correspondences among a set of matches. This method includes two stages. In the first stage, some potential correspondence points between two images are found and in the second stage a global property between points is estimated and those points which do not agree with this global specification are rejected as outlier.

Given a set of N potential matches, a set of M samples are randomly selected from them. For each sample, a model hypothesis and a support are defined. Then the model parameters and support are found based on the other samples in the set. Typically the model parameters are some global specifications such as Epipolar line or center of perspective and the number of inliers is considered as the model support. Among obtained hypothesis, the one with the maximum support is considered as the model for M samples. The model parameters are refined by all the inliers. Inliers are those samples which their model parameters are similar to the model within a threshold.

After finding the model for M random samples, the number of consensus samples with the model is compared with a threshold. If the number of inliers supporting the model is less than the threshold, the obtained model is discarded. The algorithm is repeated for the new set of M random samples chosen from the N potential matches. After a limited number of trials, either the model with the maximum number of consensus is found or the operation is terminated. The assumption made is that in each set of the M random samples, there is at least one true correspondence point. Hence, it is necessary to set a constraint for the value of M to ensure that the assumption is satisfied.

2.4.3 Matching Correspondence Points between Two Set of Points Using Convex Hull Edge

Goshtasby et al. [31] proposed a method to find the transformation parameters which maximizes the number of correspondence points between two sets S_1 and S_2. Some points in the first image and their correspondences in the other image are selected as the control points. The two images of the same scene are registered based on the computed transformation. A crucial part is to define the associated transformation parameters such as translation (T), rotation (R) and scale (S). The algorithm starts by determining the convex hull for each set of points and considering the points in the edge of each set named C_1 and C_2 as illustrated in Figure 2.5.

The transformation parameters (R, S, T) are estimated by matching a pair of points in C_1 to a pair of points in C_2. The number of points in the rest of set S_1 which could be matched to the set S_2 within a threshold with respect to the estimated transformation parameters (R, S, T) are determined. The transformation parameters which yield the maximum number of matches between the two sets (S_1, S_2) is considered as the estimated parameters (R_m, S_m, T_m) and all the points in one set are mapped to the other using the estimated parameters. Finally, the obtained parameters are refined by minimizing the sum of square errors between the true match points from both sets. The main issue regarding this method is the way C_1 and C_2 are selected. When two set of points are compared to find the real correspondences, it is not necessary true that their global shapes are the same. It might just be a small region found within the two sets that are mapped to each other in a correct manner.

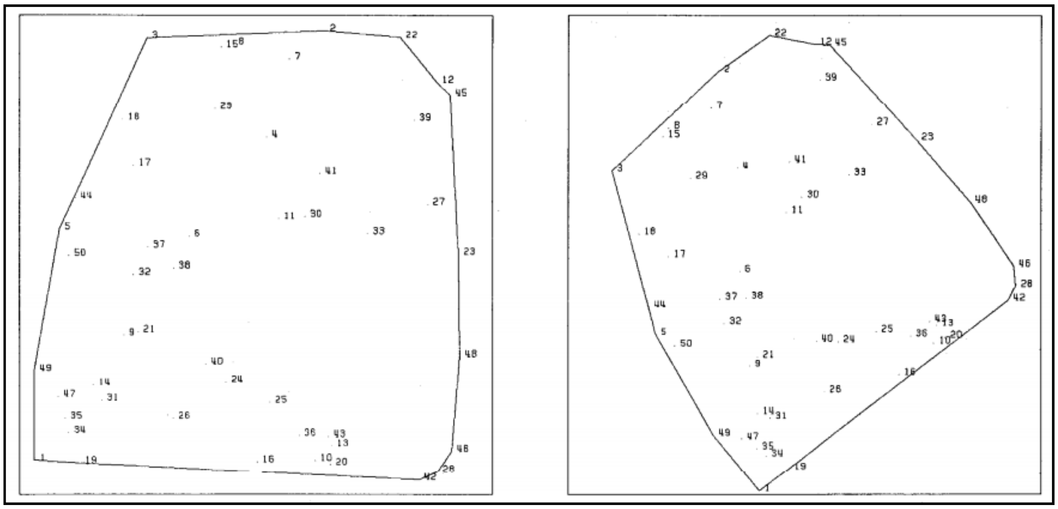


Figure 2.5. Illustrates the convex hull for each set of points by considering the points in the edge of each set. (Extracted from **[31]**)

2.4.4 Refining the Matching by Combining the Affine Invariant Interest Point Detector and RANSAC Method

Mikolajczyk and Schmid [32] proposed a matching approach to find the real correspondences between two images of similar scene using the scale and affine invariant interest point detector based on the Harris detector [27],. Their method calculates up to the 4^{th} order of Gaussian derivative around the interest points detected by Harris detector and then compute the 12 dimension descriptors. If the Mahalanobis distance between each descriptor in the first image and the most similar descriptor of the second image falls below a threshold, it is considered as a potential correct match. After finding the potential match pairs, cross correlation is applied to the images to discard the low score matches. The final correspondence points are obtained by estimating the transformation between the two images using random sample consensus (RANSAC). This method could reject the outliers and select the real correspondence points between the two images. However, the method performs well when the number of true correspondences is larger than the false correspondences. In the situations where the error is dominant and the number of real correspondence points among the all potential matches is very small, RANSAC will fail [32].

2.5 Face Detection from Different Views

Face detection is an essential part of face analysis algorithms including face recognition, face tracking, expression recognition and pose angle estimation [33]. During the recent years several face detection methods have been proposed. Some methods detect faces in the given images based on the appearance of the face such as Eigenfaces [34] or Neural Network face detection [35]. These methods rely on the facial characteristics extracted from large number of face and non-face images using machine learning techniques. However, they are sensitive to even small displacement of facial clues and thus mostly applicable to frontal faces only. Face detection based on skin color has also been studied [36, 37]. Although color information could be an efficient representation for facial area, it may vary with illumination condition, background color and ethnicity. In a famous study, Viola and Jones [1] proposed a rapid face detection algorithm using the sliding window concept and Adaboost classifier [38]. Their method is fast and efficient for frontal and near frontal faces. However, its performance is poor for non-frontal faces. In recent years, face detection from different views has been popular among researchers [2, 3, 39-41]. Most of them used a single detector in different combinations together with the sliding window concept. A few efforts used a statistical method instead of sliding window to solve the problem of face detection from multi-view. However, almost all the proposed approaches are not efficient for non-frontal faces. In the following subsections, some of the popular state-of-the-art face detection approaches from multi-view are explained and their advantages and shortcomings discussed.

2.5.1 View Invariant Face Detection Using Sliding Window

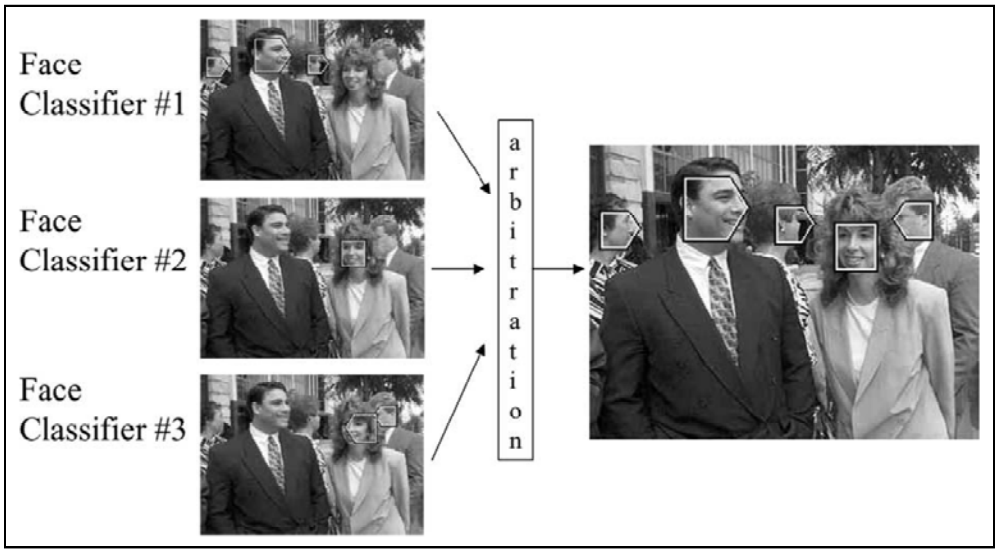


Figure 2.6. Multiple classifiers in different orientations used to detect objects from different viewpoints. (Extracted from **[2]**)

There are several studies on the sliding window concept for face detection and recognition from multi poses. An often cited work to detect objects and faces of different scales and poses is Schneiderman et al. [2]. In this method, a sliding window in different sizes scans the entire image for the presence of faces or cars. To cater to different poses and scales, an ensemble of classifiers is used as illustrated in Figure 2.6. Each classifier covers a different range of object orientation. A discrete set of wavelet coefficients is calculated for all sub regions of size $n \times m$ in each window. The calculation includes combinations of space, frequency and orientation information extracted from the sub region. The occurrence of each part over all the images in the training set is then counted to shape the tables for $P(part_r|object)$ and $P\left(part_r\middle|\overline{object}\right)$. Classifiers are then trained using the statistic of both object parts and non-object parts. Adaboost [38] is used to minimize the classification error on the training set by re-weighting the occurrence of each part. In the detection stage, the occurrence of the parts are computed for each of the sliding window iteratively until the entire test image is evaluated and compared to the values obtained during the training stage. Finally the likelihood ratio test is used to determine whether the image window contains the object of interest or not.

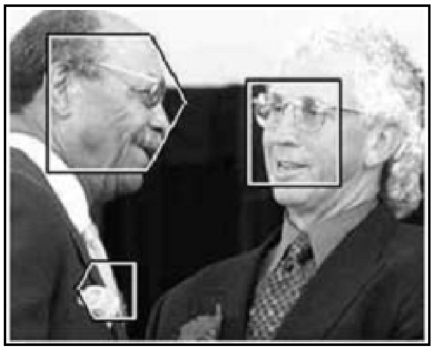


Figure 2.7. Demonstrates the result of window based approaches. The detected face region is coarse as they only specify a window region where the face is located but not necessarily specify the entire face or a consistent reference base. (Extracted from **[2]**)

Although such an approach produced good performance, there are several limitations. Training an ensemble of classifiers for many poses and scales is very difficult and time consuming. The size of windows varies discretely; therefore considering a small number of window sizes might not cover all possible object scales while increasing the number of windows increases the computation time significantly. Moreover, this method only yields coarse face detection as illustrated in Figure 2.7. It means that a window region is determined which might include a part of a face but does not accurately locate the entire face region as it might only covers part of the face or includes also the background. It also does not provide a reference base for the detected face. Another limitation of the method is that manual intervention is required in the training stage such as manually locating the face or to align and label the object.

The other famous studies that use the sliding window concept are the methods introduced by Chen et al. [41] and Huang et al. [3]. In the method by Chen et al. [41], the Eigenfaces were chosen as the face features and Gaussian mixture model was used to select the model features. Huang et al. [3] used Haar-like features and Adaboost classifier similar to Viola and Jones [1] but with different window sizes in various poses. However, all these methods require the training data to be cropped, aligned and labeled manually. For instance, Huang et al. [3] reported manually labeling 30,000 frontal, 25,000 half-profile and 20,000 full-profile faces. Viola and Jones [42] also reported manually collecting and cropping 8356 face images and over 100 million

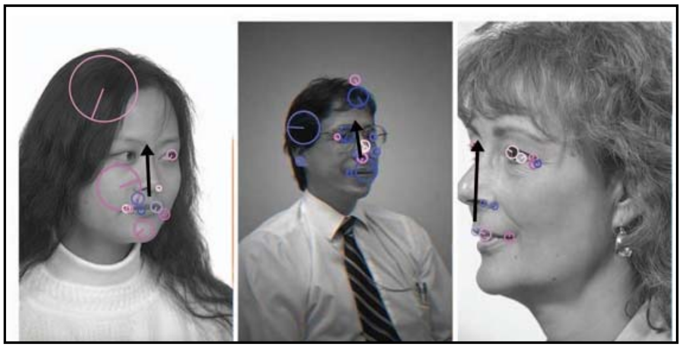


Figure 2.8. Manually labeled faces from different views. This label is a line from the base of the nose to the forehead. (Extracted from **[40]**)

background and non-face samples to train their classifiers. Such effort is costly and time consuming.

2.5.2 View Invariant Face Detection Using Statistical Classifiers

Toews and Arbel [40] proposed a statistical approach to detect face images in multi-view using a Bayesian classifier. To train their method, a reference vector is manually labeled in all the training images. This vector is a line from the nose base to the forehead (see Figure 2.8) and is called OCI.

Scale invariant features are extracted from the training images. The geometry of the features are then normalized based on the geometry of the reference vector in each image. Clusters of features, which agree in both the normalized geometry and appearance, are obtained using the method similar to mean-shift [43]. For the clustering, each extracted feature is considered as the potential cluster center and its geometry is compared to the other features with respect to position, orientation and scale thresholds to find the cluster members for that cluster center. The appearance of each cluster center is also compared to the appearance of the other features. The ratio between the number of features that both agree in geometry and appearance with the number of features that only agree in appearance, is calculated for each cluster center. This ratio gives the distinctiveness of each cluster center and helps the method to discard the redundant features. This value also tells the probability that a feature comes

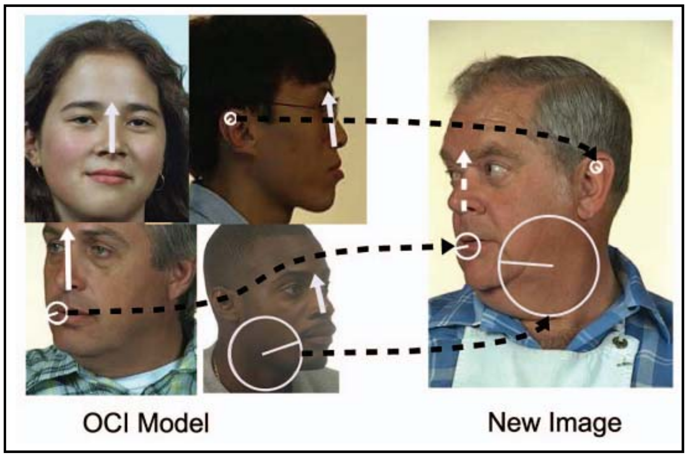


Figure 2.9. Finding the match features between the model and the test image. Each match proposes the existence of an OCI geometry in the new image. (Extracted from **[40]**)

from a part of a face or it is just a background or noise sample. A feature model is then constructed based on the discriminative features found. In the test stage, all the scale invariant features from the test image are extracted and compared to the model features based on the appearance term as shown in Figure 2.9. Each match to the model implies that the geometry of an OCI, as well as the cluster of similar OCIs, in the test image proposes the existence of a true OCI. The hypothesis that the proposed OCI comes from a true face is examined using the Bayesian decision ratio in (2.21).

$$\gamma(o^g) = \frac{P(o^g, o^{b=1}|\{m_i\})}{P(o^g, o^{b=0}|\{m_i\})} \tag{2.21}$$

Although the performance of this method for the frontal face is not as well as the window based methods, the result for the other views is very promising. However, there are several shortcomings. First, manual labeling needed for the training is significant. Secondly, the performance of this method can be improved for both the frontal and non-frontal views. Thirdly, the method fails for in-plane rotated faces and finally its processing time is high and could not support real time applications.

2.6 Face Recognition and Identification for Web Images

Recently there are several reported researches on face recognition for images in the web, particularly in entertainment sites and social networks [4-10]. Nevertheless, the approaches follow the same framework of cascaded stages. All the approaches use some existing face detection methods such as [1-3, 41] to detect the face area and apply the alignment procedure such as [44, 45] to limit the cropped images to the facial area and ensure consistent correspondence in all the images. Facial features are then extracted using global approaches such as Eigen-faces [46], Fisher-faces [47], sparse-representation [48] or local features such as local binary patterns [49] and its variants [50, 51]. This is then followed by some kind of classification stage to label or recognize the faces. However, cascading face detection, localization, alignment and face recognition algorithms causes the entire cascaded system to be limited by the limitations or performance of the individual stages. For example, even though the face recognition algorithm can support multi-view faces, the system could not recognize the non-frontal face if the face detection stage could not handle non-frontal faces. The following subsections further elaborate the reported methods.

2.6.1 Face Annotation with Interactive Labeling

A face naming and annotation method has been introduced by Tian et al. [4]. The face regions in each photo was determined using Viola-Jones face detector [1] and aligned using the eyes detected [45]. They used a similarity matrix and measured the pair wise similarities between images computed using local binary pattern features [49] on the face area and contextual Color Corregram [6] on the clothes area. A partial clustering approach based on Gaussian Mixture Model was employed to cluster the detected faces. The primary clustered faces were then provided to the user to label the identities manually. Assuming that each person could only appear once in an image, the gathered information was then used to update the clusters. Their framework is illustrated in Figure 2.10. There are two shortcomings regarding this framework. Firstly, manual labeling is required which is substantive considering the vast volume of images.

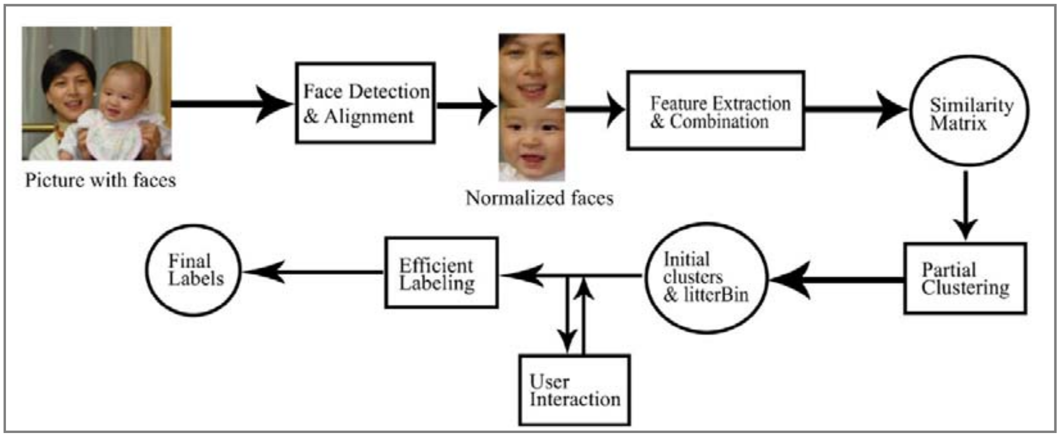


Figure 2.10. Face annotation framework introduced by Tian et al. [4]. (Extracted from [4])

Secondly, the performance of the method is limited by the performance of the face detector [1] used and thirdly, alignment using the eyes are valid for only frontal and near frontal images where both eyes are present

2.6.2 Face Recognition from Multi-poses with Random Projection Matching

Wright and Hua [48] proposed a face recognition method based on matching of spatial and feature information in semi-constrained environments. They used Viola-Jones face detector [1] and eye detection procedure to detect and align the faces. Illumination variation effect was reduced using photometric rectification. They extracted high-dimensional near-invariant features calculated at dense locations in the image space and at different scale. A set of randomized decision trees quantize the final representation of the face as a sparse histogram. Their algorithm is illustrated in Figure 2.11. There are some shortcomings. Firstly, face alignment based on eye detection is only applicable to the frontal and near frontal faces where both eyes are present. Secondly, the training images are collected from the web, hence, it is difficult to put a semi-constrained environment in the images, such as requiring frontal or near frontal faces, resulting in improper training if non suitable images are not removed. Thirdly, the performance of their algorithm is limited also limited by the performance of the face detector. Although they claimed that their framework is able to recognize faces of

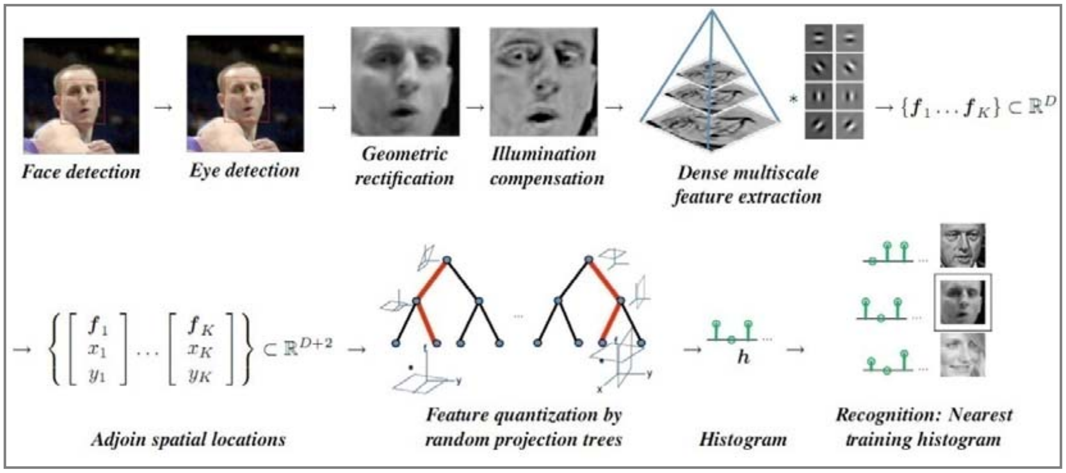

Face detection
Eye detection
Geometric rectification
Illumination compensation
Dense multiscale feature extraction
$\rightarrow \{f_1 \dots f_K\} \subset \mathbb{R}^D$
Adjoin spatial locations
$\rightarrow \left\{ \begin{bmatrix} f_1 \\ x_1 \\ y_1 \end{bmatrix} \dots \begin{bmatrix} f_K \\ x_K \\ y_K \end{bmatrix} \right\} \subset \mathbb{R}^{D+2}$
Feature quantization by random projection trees
Histogram
h
Recognition: Nearest training histogram

Figure 2.11. Face recognition framework proposed by Wright and Hua [48]. (Extracted from [48])

different poses, however, the face detectors such as viola-Jones [1] could not detect the faces from multi-poses efficiently. Thus, their algorithm would fail if the faces are not even detected.

2.6.3 Face Recognition from Caption-Based Supervision

Guillaumin et al. [7] introduced a face recognition and naming method for news images. Their method returns the faces belonging to a specific person in the given images and labels the faces accordingly as depicted in Figure 2.12. They used Viola-Jones method [1] to detect the faces and Huang et al. funneling method [44] for face alignment. Then they detected the facial features at nine points in the face area including eyes and mouth corners and centers and extracted the SIFT descriptors for these points in three different scales in the same way as that of [52]. The vector obtained from the feature descriptors of these points is then used as a representative model for the face image and compared to the representative vectors obtained from the other faces using Mahalanobis distance. In the training stage, they computed the representative vectors for a set of images from the person of interest. A metric was then learned from the representative vectors using two different learning approaches - Turk and Pentland [46] (PCA) and the logistic discriminant based metric learning (LDML) introduced by Guillaumin et al. [53]. Their system then searched the news

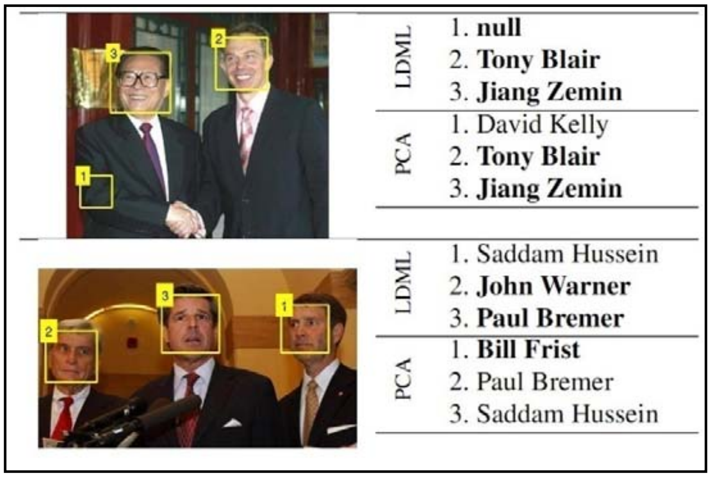


Figure 2.12. Shows the face identification and labeling results performed by Guillaumin et al. [7] in the news images from yahoo news. (Extracted from [7])

database to find the captions that contain the name of the queried person. Faces detected in the news image were ranked based on visual similarity to the face of the obtained model. A graph-based approach is then used to associate the retrieved names to the detected faces.

The main shortcoming of this method is the huge manual labeling needed for the training stage. They reported manually annotating 28,204 documents and associated the correctly detected faces and detected names.

2.7 Concluding Remarks

In this chapter the important image processing tools and state–of-the-art approaches related to our study have been studied. This includes the scale invariant feature extraction, probabilistic analysis of image features, and the matching techniques. Although the matching methods work well for images of fairly similar scene where the number of real correspondences is larger than the false correspondences, they fail when the number of real correspondences is smaller than all the potential correspondences found between the image pairs.

We also investigated several methods for multi-pose face detection. The window based algorithms are popular for face detection and localization, but the fixed size and the shape of the window is cannot determine the face area accurately and would require an ensemble of classifiers trained for difference discreet pose. Statistical frameworks are better in localizing the face area but their performance is not as high as the window based method. Moreover, both approaches require extensive manual intervention to label and crop the training data.

In the last section, we studied the state of the art face labeling and recognition frameworks. The main and common shortcoming is the large amount of manual intervention needed to prepare the training data. The other important drawback is the dependency to the limitation of each cascaded stage used as they typically only limit their research to a single stage in the cascade. Most of the works use the existing face detector and are limited to frontal or near frontal faces.

Chapter 3

Finding Correspondence Points among Image Pairs of Different Views

3.1 Introduction

Finding real correspondence points between images of different views has been in demand in many computer vision applications. Many matching techniques have been introduced by researchers during the past decade [14, 32, 54-58]. Among them, the method proposed by Mikolajczyk and Schmid [32] and by Lowe [14] are very famous. Mikolajczyk and Schmid [32] proposed to find the correspondence points between two images of similar scene using scale and affine invariant interest point detector based on Harris detector [27]. They used cross correlation to discard the low score matches. Lowe [14] proposed another matching method based on scale invariant features. In his method, after finding keypoints in two comparing images, the appearance descriptors from features of the first image are compared to the appearance descriptors of the second image. Finally, the nearest neighbor of each feature descriptor is considered as the best match if its distance was significantly smaller compared to the distance of the second closest match.

These matching algorithms are not perfect and they return many incorrect correspondence points. Thus, a refinement method should be used to reject outliers and select the real correspondence points between two images. Typically, the optical flow between two comparing images is considered for matching refinement and a method such as RANSAC method [30] or Hough transform [59] is performed to find the parameters of the optical flow. Hough transform is useful only when we have a model of object and we desire to refine the matches between the model and the query image. RANSAC is more general than Hough transform and it has been used widely in many computer vision applications. However, it still has some limitations [60-62]. It fails when the number of real correspondence points is small compared with all the matches found between the two images [32]. Thus, any matching technique based on RANSAC is not able to find real correspondence points between the pair of images which are mostly dissimilar. The main issue of these matching and refinement techniques is that they all use only appearance descriptor to find the correspondence points. Dorko and Schmid [25] have shown that the appearance of a descriptor from a patch of object could have many similarities to the one from noise and background area if the descriptor was not discriminative. Therefore, other constraints should be considered to find reliable matching points between image pairs. In this section we present a refinement to the existing correspondence point matching approach. It uses appearance and geometric information of the matched points in addition to considering the discriminative features to find the real correspondence points between comparing images in the presence of large false matches. Such matching algorithm is needed in our multi-pose face detection and recognition framework which we will explain in more detail in subsequent chapters.

3.2 A Robust Matching Framework

In the proposed approach, firstly, interested points and their descriptors are extracted from two comparing images using the scale invariant features (SIFT) [14]. Then, common appearance features are discarded by computing the likelihood ratio for each feature. Finding the real correspondence points proceeds with applying the Lowe’s matching algorithm [14] to the pruned features in order to find the potential correspondence points between these two images. Finally, the geometric registration is applied to find the real matching points. These steps are described in more details in the following subsections.

3.2.1 Distinctive Features and Common Features

In order to find the distinctive correspondence points, we first consider a pair of images, I_i and I_j , and extract invariant feature points for them using the scale invariant feature transform (SIFT) [14]. Let $F_i = \{f_{i,k}\}$, $k = 1,2,, \dots, N_i$ and $F_j = \{f_{j,l}\}$, $l = 1,2,, \dots, N_j$ be the collection of feature points obtained from images I_i and I_j , respectively. Here, N_i and N_j are the number of feature points extracted from images I_i and I_j , respectively. Each feature point includes both appearance (f^A) and geometric (f^G) descriptors. For instance, the notation $f_{i,k}^A$ $(f_{i,k}^G)$ represents the appearance (geometric) descriptor of the k^{th} feature in image I_i. The appearance descriptor (f^A) is a histogram of gradients around the keypoint area with 128 bins in double precision. The geometric descriptor can be represented as $f_{i,k}^G = (x_{i,k}, y_{i,k}, \sigma_{i,k}, \phi_{i,k})$, where (x, y) is the position of the feature point k in image I_i , σ is the scale, and ϕ is the gradient orientation.

All the scale invariant features (SIFT) extracted are not distinctive and may be derived from noise or background clutter. When there is background clutter or when the pose variation between the faces in the two images are significant, the extracted SIFT features would have a substantial number of common or not distinctive features in addition to the required distinctive features. Thus, only a few features can represent the object very well. Unfortunately, those non-distinctive features impress the effect of the distinctive features so much and may change the result. In this part, we aim to select the most distinctive features, which are expected to be the most discriminative parts of an object. It is possible that a feature from the object area and a feature of the background have similar appearance [25]. It indicates that the appearance of this feature is not sufficiently discriminative to represent the object. We are only interested in those features that could represent object of interest with high probability. Suppose we have extracted a feature from an image that includes the object of interest. We can consider two different hypotheses: either the feature $f_{i,k}$ is a distinctive part of a face $(H_{i,k} = 1)$ or it is a non-distinctive feature $(H_{i,k} = 0)$. In order to determine this hypothesis and to find the discriminative features, a likelihood ratio test $\gamma(f_{i,k})$ in (3.1) is performed on all the features of two comparing images. Those features with likelihood ratio greater than one are considered as distinctive features.

$$\gamma(f_{i,k}) = \frac{p(H_{i,k} = 1|\Omega)}{p(H_{i,k} = 0|\Omega)} < 1 \tag{3.1}$$

where Ω is a sequence of observation over feature $f_{i,k}$ in the training images. We have used maximum likelihood estimation (MLE) technique to evaluate the above ratio as follows:

Let feature $f_{i,k}$ be a binary random variable that might be originated from a part of object ($H_{i,k} = 1$) or non-face ($H_{i,k} = 0$) area. We can assign a Bernoulli distribution with parameter θ to it that determines its uncertain value as shown in (3.2)

$$p(f_{i,k}|\theta) = \theta^{f_{i,k}}(1-\theta)^{1-f_{i,k}} \tag{3.2}$$

Given a sequence of M observation data Ω over feature $f_{i,k}$ in the training data, the likelihood for the given data Ω is obtained by (3.3) assuming that the observations are independent.

$$p(\Omega|\theta) = \prod_{t=1}^{M} \theta^{f_{i,k}^{t}}(1-\theta)^{f_{i,k}^{t}} = \theta^{M_k}(1-\theta)^{M-M_k} \tag{3.3}$$

where M_k is the number of times that feature $f_{i,k}$ has represented a face area in the observation data Ω. The term $f_{i,k}^{t}$ represents the t^{th} observation sample over feature $f_{i,k}$ in the training data. When we have sufficient number of observation data or M is a large number, we can estimate the distribution parameter θ using MLE estimation in (3.4).

$$\hat{\theta}^{MLE} = \underset{\theta}{argmax}\, p(\Omega|\theta) \tag{3.4}$$

Thus we compute MLE for the feature $f_{i,k}$ as follows: Let feature $f_{1,k}$ be a feature from the first image I_1 which is compared to all N_2 features ($f_{2,l}$) in the second image I_2. If this feature is a distinctive feature of the face, it should be unique and the occurrence of it in the other part of the face would be very limited. Ideally, there would only be one such feature and for symmetrical parts, only two features can be found. Thus, we can estimate its probability as $p\left(H_{1,k}=1\middle|\Omega\right)=(2/N)$. But if this feature originated from the non-distinctive or common appearance part of the image, the number of similarities found for it could be any number, α. Thus, we can estimate its probability as $p\left(H_{1,k}=0\middle|\Omega\right)=(\alpha/N)$. Substituting these estimations into (3.1) indicates that all features with similarity number (α) greater than 2 can be regarded as common appearance features. To compute the similarity number of feature f_k in the first image, its appearance descriptor ($f_{1,k}^A$) is compared to the appearance descriptors ($f_{2,l}^A$) of all features in the second image within a global threshold ($\left(f_{1,k}^A-f_{2,l}^A\right)^2<T^a$) ($f_{1,k}^A$ is the appearance vector of *k*th SIFT feature in the first image with 128 bins and $f_{2,l}^A$ is the appearance vector of *l*th SIFT feature in the second image). This procedure is then repeated for each feature found in the second image, which is then compared to all the features in the first image. Finally, all common features found are removed. To find an optimum value for T^a several image pairs from different databases (FERET, CMU, and FDDB) with various poses (Left, Right and Frontal views) have been investigated. The results of finding correspondence points between some image pairs with different thresholds are demonstrated in Figure 3.1. As it is seen, a value between 0.2 to 0.3 is suitable value for T^a. A greater value causes high elimination of the distinctive features whereas a lower threshold allows more noise and common features to be accepted as the correspondence points. Since this threshold corresponds to feature appearance descriptors, its value is fixed for the features extracted by SIFT method whose descriptor has 128 bins (standard SIFT feature). Applying the explained procedure might reduce about 10 percent of the features, but these small numbers of features if not removed, could distort the final results significantly. Figure 3.2

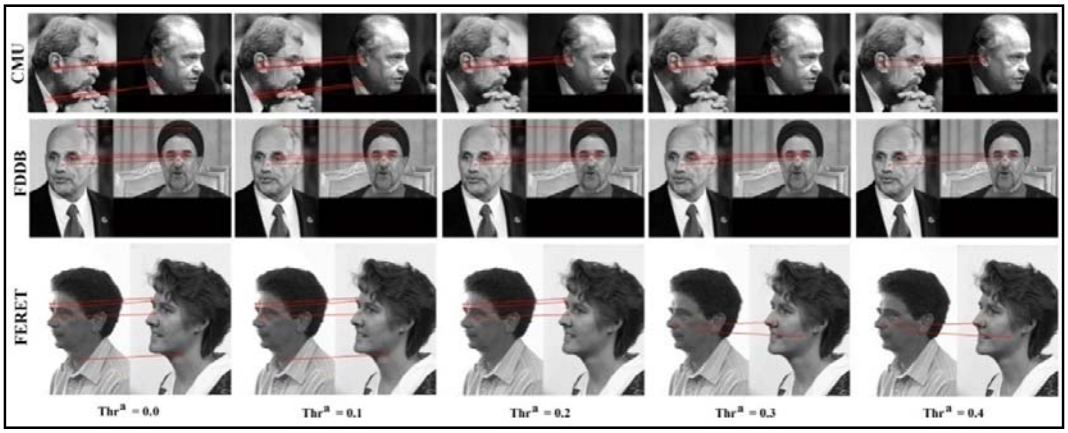


Figure 3.1. Shows the results of applying the different feature appearance thresholds $\boldsymbol{T^a}$ to find common features and distinctive features. Several image pairs from different databases (CMU, FDDB and FERET) and faces from various poses (Left, Frontal and Right views) have been used in this evaluation to find an optimum threshold, which yields the elimination of maximum number of common features and least number of distinctive ones. As it is seen in the results, getting a value in between 0.2 to 0.3 gives the best performance.

compares the results of our proposed matching algorithm with and without applying the preprocessing stage to the features to remove the common features. In Figure 3.2(a) the result of applying matching algorithm based on Lowe's method without discarding the common features is shown. If the common appearance features are discarded prior to applying the Lowe's matching method, the result is shown in Figure 3.2(b). If the common appearance features are not removed from two comparing images, the possibility that non-object correspondence points appear in the result increases as it is demonstrated in Figure 3.2(c). Hence, the best results could be obtained after removing common features as it is depicted in Figure 3.2(d). Graphical representation in Figure 3.3 and Algorithm 3.1 illustrates the matching procedure.

3.2.2 Finding Real Correspondence Points

Mikolajczyk and Schmid [32] suggested to use RANSAC method [30] to reject outliers and select real correspondence points between two images. RANSAC method performs well when data contains a gross error. In the situation that error is dominant and the number of real correspondence points among all the found matches is very

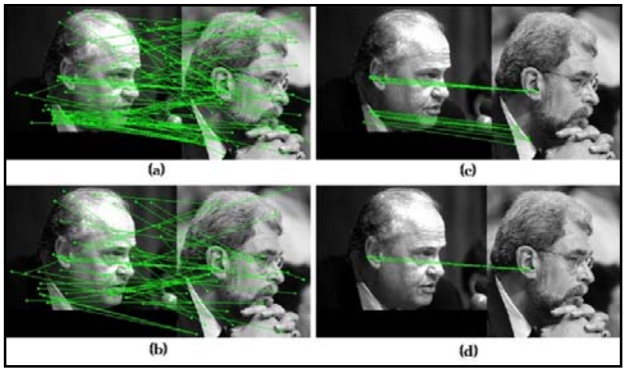


Figure 3.2. Illustrate the difference between not discarding (first row) and discarding (second row) common appearance features in finding the correct correspondence between two objects of the same class

small, RANSAC fails [32]. As in our case study where the number of real correspondence is a small portion of all found matches, it is not suitable to use RANSAC. We proposed to use a registration technique similar to the method described by Goshtasby [31] with amendment of the normalized map to make it suitable for our method. We also used both the geometric information and appearance data to increase the accuracy of our proposed matching method.

After discarding the common features, all potential correspondence points were found between the two comparing images by applying Lowe's matching algorithm [14]. Let $f^G_{i,k(m)} \leftrightarrow f^G_{j,l(m)}$ denote the m^{th} potential correspondence, where $k(m)$ is the index of the m^{th} corresponding feature point in F_i, $l(m)$ is the index of the m^{th} corresponding feature point in F_j, and $m = 1,2, \dots, M$.

These potential correspondences are further filtered using geometric constraint to produce only the salient correspondences. Initially, each matching pair $f^G_{i,k(m)} \leftrightarrow f^G_{j,l(m)}$ is considered a reference correspondence. The geometry of all other matched feature points in each image, are normalized with respect to the scale, position, and orientation of the reference corresponding point in that image using (3.5) and (3.6).

$$\begin{bmatrix} x^N_{i,k(n)} \\ y^N_{i,k(n)} \end{bmatrix} = \frac{1}{\sigma_{i,k(m)}} \begin{bmatrix} cos\,\phi_{i,k(m)} & sin\,\phi_{i,k(m)} \\ -\,sin\,\phi_{i,k(m)} & cos\,\phi_{i,k(m)} \end{bmatrix} \begin{bmatrix} x_{i,k(n)} \\ y_{i,k(n)} \end{bmatrix} - \frac{1}{\sigma_{i,k(m)}} \begin{bmatrix} x_{i,k(m)} \\ y_{i,k(m)} \end{bmatrix}$$

$$\sigma^N_{i,k(n)} = \frac{\sigma_{i,k(n)}}{\sigma_{i,k(m)}}, \tag{3.5}$$

$$\phi^N_{i,k(n)} = \phi_{i,k(n)} - \phi_{i,k(m)}.$$

$$\begin{bmatrix} x^N_{j,l(n)} \\ y^N_{j,l(n)} \end{bmatrix} = \frac{1}{\sigma_{j,l(m)}} \begin{bmatrix} cos\,\phi_{j,l(m)} & sin\,\phi_{j,l(m)} \\ -\,sin\,\phi_{j,l(m)} & cos\,\phi_{j,l(m)} \end{bmatrix} \begin{bmatrix} x_{j,l(n)} \\ y_{j,l(n)} \end{bmatrix} - \frac{1}{\sigma_{j,l(m)}} \begin{bmatrix} x_{j,l(m)} \\ y_{j,l(m)} \end{bmatrix}$$

$$\sigma^N_{j,l(n)} = \frac{\sigma_{j,l(n)}}{\sigma_{j,l(m)}}, \tag{3.6}$$

$$\phi^N_{j,l(n)} = \phi_{j,l(n)} - \phi_{j,l(m)}.$$

where $(x^N_{i,k(n)}, y^N_{i,k(n)}, \sigma^N_{i,k(n)}, \phi^N_{i,k(n)})$ and $\left(x^N_{j,l(n)}, y^N_{j,l(n)}, \sigma^N_{j,l(n)}, \phi^N_{j,l(n)}\right)$, $(n = 1, \ldots, M; n \neq m)$ are the normalized geometric descriptors of the other matched feature points with respect to the potential reference point. The matched pairs that are geometrically consistent with respect to the reference correspondence $f^G_{i,k(m)} \leftrightarrow f^G_{j,l(m)}$ are found by applying the threshold individually to each geometric attribute using (3.7).

$$\begin{aligned} C^x_m &= \left\{n: \left|x^N_{i,k(n)} - x^N_{j,l(n)}\right| < T^X_H\right\}, \\ C^y_m &= \left\{n: \left|y^N_{i,k(n)} - y^N_{j,l(n)}\right| < T^Y_H\right\}, \\ C^\sigma_m &= \left\{n: \left|log\,\sigma^N_{i,k(n)} - log\,\sigma^N_{j,l(n)}\right| < T^\sigma_H\right\}, \\ C^\phi_m &= \left\{n: \left|\phi^N_{i,k(n)} - \phi^N_{j,l(n)}\right| < T^\phi_H\right\}. \end{aligned} \tag{3.7}$$

where C_m^x , C_m^y , C_m^σ , $and\ C_m^\phi$ are the four groups of indexes to those potential correspondences which satisfy the geometric constraints in the normalized space created by the reference correspondence. The thresholds $(T_H^X, T_H^Y, T_H^\sigma, T_H^\phi)$ control the amount of tolerance to variations between the two images. The lower the tolerance, the higher is the degree of match required between the two feature points. This reduces the amount of false matches but also reduces the amount of allowed variations, consequently affecting the ability to support larger pose changes. Normalizing the features to unit length of the reference feature m, we allow the position of the match points to vary by about half. Thus we used 0.5 for T_H^X and T_H^Y. Similarly, to provide the same tolerance level, we set T_H^ϕ as 25 degrees and T_H^σ as $log(1.5)$.

The intersection of the four sets of indices in equation (3.7), $C_m = \{C_m^X \cap C_m^Y \cap\ C_m^\sigma\ \cap\ C_m^\phi\}$ forms the match cluster for the reference correspondence $f_{i,k(m)}^G \leftrightarrow f_{j,l(m)}^G$. Let the size of a match cluster, i.e., the number of elements in C_m, be denoted as s_m. The procedure is then repeated for all the matched pairs, $f_{i,k(m)}^G \leftrightarrow f_{j,l(m)}^G$ and $m = 1,2,\dots,M$, to obtain the cluster which has the maximum value of s_m. Let $m^* = argmax_m\ (s_m)$ denote the set of values of m that maximize s_m. If m^* has more than one element, i.e., more than one reference correspondence give the same maximum match cluster size, the reference correspondence which results in the lowest Euclidean distance among its underlying matched pairs in the normalized space is taken as the real correspondence as indicated in equation (3.8).

$$C_R = \begin{cases} C_{m^*}\,, & if\ |m^*| = 1 \\ C_{m'}, m' = \underset{m \in m^*}{argmax}\left(\sum_{n\,\in\,C_m} R_{i\leftrightarrow j,n}^N\right), & if\ |m^*| > 1 \end{cases} \qquad (3.8)$$

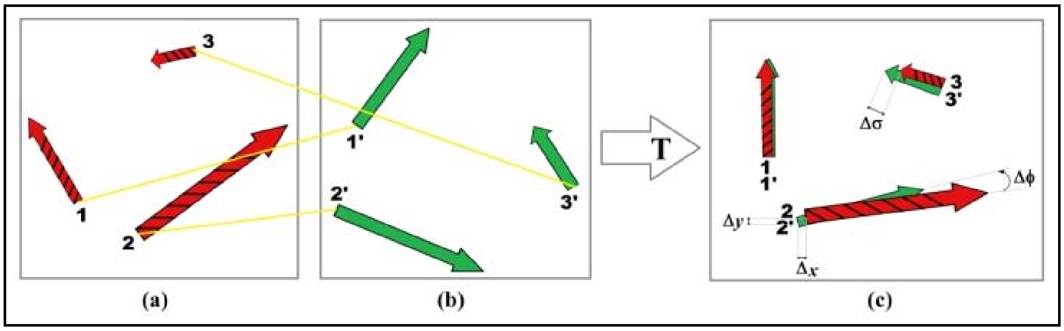


Figure 3.3. Depicts a graphical representation for finding consensus points after applying the affine transformation to two sets of correspondence points.

where C_R is the cluster of indexes to the real correspondence points between the two images and $R^N_{i\leftrightarrow j,n} = \left(x^N_{i,k(n)} - x^N_{j,l(n)}\right)^2 + \left(y^N_{i,k(n)} - y^N_{j,l(n)}\right)^2$ is the square of the Euclidean distance between the normalized pairs in the underlying cluster C_m. For a valid match, we require three points from one plane to be matched to three points from another plane. Since each pair contains information about location, scale, and orientation, a minimum of two pairs of the matching feature points are sufficient to produce a valid match (i.e., the size of the cluster C_R should be greater than or equal to 2). However, higher number of matched pairs increases the confidence level of the matching operation. The procedure is summarized in Algorithm 3.1.

The graphical representation of finding consensus points is demonstrated in Figure 3.3. In this figure, (a) and (b) are two sets of potential correspondence points between two images. Let feature "1" (Figure 3.3(a)) be a real match for feature "$1'$" (Figure 3.3(b)); thus, after applying the affine transformation based on this feature to the two sets of correspondence points, the geometric relationship should still remain unchanged. To measure the consistency of the transferred points, we let their geometries to vary by a small amount $\Delta x, \Delta y, \Delta\sigma$ and $\Delta\phi$ in Figure 3.3(c).

Algorithm 3.1. Finding distinctive correspondence points and Reference points.

```
Input: Two face images $i$ and $j$
Output: Real correspondence points between two faces

1. Extract SIFT features from two images.
2. Remove common features from them using appearance descriptor of SIFT.
3. Consider the $K$ potential correspondence points $f_{i,m}^{G} \leftrightarrow f_{j,m}^{G}$ , $(m = 1,2, \dots, K)$ between
   two images ($i$ and $j$) obtained from the Lowe's matching algorithm, where $f_{i,m}^{G} =$
   $(x_{i,m}, y_{i,m}, \sigma_{i,m}, \phi_{i,m})$ contains the locations, scale and orientation of the feature $f_{i,m}^{G}$
   calculated by the SIFT algorithm.
4. For $m = 1,2, \dots, K$ Do
   (a). Assume the feature $f_{i,m}^{G}$ is the reference point.
   (b). Normalize all the geometric descriptors of potential points $f_{i,k}^{G} = (x_{i,k}, y_{i,k}, \sigma_{i,k}, \phi_{i,k})$,
        $(k = 1, \dots, K \;\&\; k \neq m)$ in the first image $i$ to the geometric descriptor of the
        reference point $f_{i,m}^{G} = (x_{i,m}, y_{i,m}, \sigma_{i,m}, \phi_{i,m})$.
   (c). Name the normalized geometric descriptors as $\{f_{i,k}^{N}\} = \{(x_{i,k}^{N}, y_{i,k}^{N}, \sigma_{i,k}^{N}, \phi_{i,k}^{N})\}$,
        $(k = 1, \dots, K \;\&\; k \neq m)$.
   (d). Repeat (a), (b) and (c) for the second image $j$.
   (e). Find the cluster of consensus points $(C_m)$ with respect to the reference point $m$,
        which are the points that their normalized geometric descriptors are in a vicinity
        $\left|\{f_{i,k}^{N}\} - \{f_{j,k}^{N}\}\right| < T_{H}^{x,y,\sigma,\phi}$.
End
5. Consider the pair with the maximum number of inliers $f_{i,R}^{G} \leftrightarrow f_{j,R}^{G}$ as the reference pair,
   where $R = argmax(C_m)$, $(m = 1,2, \dots, K)$).
   5.1. If more than one pair was found as the reference pair, select the one that its inliers in
        the two images have closer distance to each other $(C_R = argmin_{C_m}\left(\sum_{n \in C_m} R_{i\leftrightarrow j,n}^{N}\right))$.
6. Return all the pairs $f_{i,r}^{G} \leftrightarrow f_{j,r}^{G}$ , $(r \epsilon\, C_R)$ as the real correspondence points and the pair
   $f_{i,R}^{G} \leftrightarrow f_{j,R}^{G}$ as the reference points between two images.
```

3.3 Experimental Result and Discussion

In this section, experimental results of the proposed framework on different input datasets are evaluated. In order to extract the SIFT features in comparing images, we used the SIFT implementation provided by Vedaldi [63] based on the method proposed by Lowe [14]. The SIFT parameters such as number of octaves of the Gaussian scale space and number of scale levels within each octave are considered such that at least 1000 features are extracted from each image. Using a 2.66GHz C2Q PC with 3 GB RAM running Windows Vista 32-bit OS, the entire process to locate the correspondence points takes less than a second for an image pair of size 256x384 pixels, based on a Matlab implementation.

3.3.1 Experimental Setup

The first image dataset have been used to evaluate the matching algorithm is the dataset that Mikolajczyk and Schmid [15] used to determine the performance of different feature extraction methods in different scenarios. This dataset is available at [64] and contains eight sets of different scenes, which may be blurred, zoomed, rotated or with changed illumination condition and viewpoints in six different levels. JPEG compression effect is also included. The images have different resolution from 765x512 to 1000x700 pixels. To make sure that the proposed matching algorithm is reliable, the worst-case images of each category are selected, which means the pair with maximum variation is selected in each category. The second data sources used to evaluate our matching algorithm, are multi-view evaluation images provided by Strecha et al. [65]. These images have wide view variation. The third dataset, which have been used to find reliable correspondence points among faces of multi-views is the standard color FERET images [66]. This dataset contains over 1200 various people with minimum 7 face images of different views from left profile to oblique, frontal and right profile. The Fourth dataset is CMU profile face images [67] that we have used to investigate the optimum threshold for T^a. In this dataset, faces are taken in high clutter background and the resolution of the face images is poor (i.e. the size is small). The fifth dataset was the FDDB dataset that contains 2845 images taken from the faces in the Wild data [68]. Some images of objects such as airplane have been gathered from web.

3.3.2 Mutual Information Evaluation Criterion

To measure the accuracy, we computed the precision and recall value. However, using comparison factors such as precision, recall, accuracy or F-score to rank binary classifiers can yield counterintuitive results [69]. A more suitable measurement is the use of “informativeness” measurement described in [70, 71]. This measurement shows the Mutual information between the predicted label and the ground truth label using (3.9).

$$I(y,t) = \sum_{y=0}^{1}\sum_{t=0}^{1} p(y,t)\, log\, \frac{p(y,t)}{p(y)p(t)} \tag{3.9}$$

where $p(y = 1, t = 1)$ represents the true positive normalized with the number of test points, $p(y = 1, t = 0)$ represents the normalized false positive and so on. The two individual terms can be calculated by applying summation in one variable for instance $p(y) = \sum_{t=0}^{1} p(y,t)$.

3.3.3 Matching Objects and Scenes from Multi-views

To show that the proposed method is generally applicable to many matching applications, we have evaluated our methods on images of different scenes and objects from multi-views. However, it is not necessary the best and suitable approach for other case studies as we optimized it to perform well for face images from multi-poses.

The robustness of the algorithm over zoomed-rotated images, 180 rotation and different viewpoints is examined in Figure 3.4. A total number of 86 correspondence points are found between the comparing images in Figure 3.4(a) in image resolution of 425×340 and 1004 correspondence points found in Figure 3.4(b) in image resolution of 378×256 without any error. If two comparing images follow an affine transformation, the number of real correspondence points is usually large. However, the algorithm is also able to find correspondence points between images that undergo nonlinear transform such as those with wide view differences as shown in Figure 3.4(c & d). The main issue of the comparing image pair in Figure 3.4(d) is the repeated textures such as the window pattern that usually cause confusion to the matching techniques to detect the correct correspondence points. As the proposed method overlays two comparing images in the feature level and considers the relative geometry between features, this problem is solved. The number of correspondence points in the comparing images at Figure 3.4(c) is as small as 10 points for the resolution of 308×205 due to wide tilt angle between them. Images at Figure 3.4(d) have a milder variation in tilt angle which yielded 118 correspondence points among them for the resolution of 308×205 pixels.

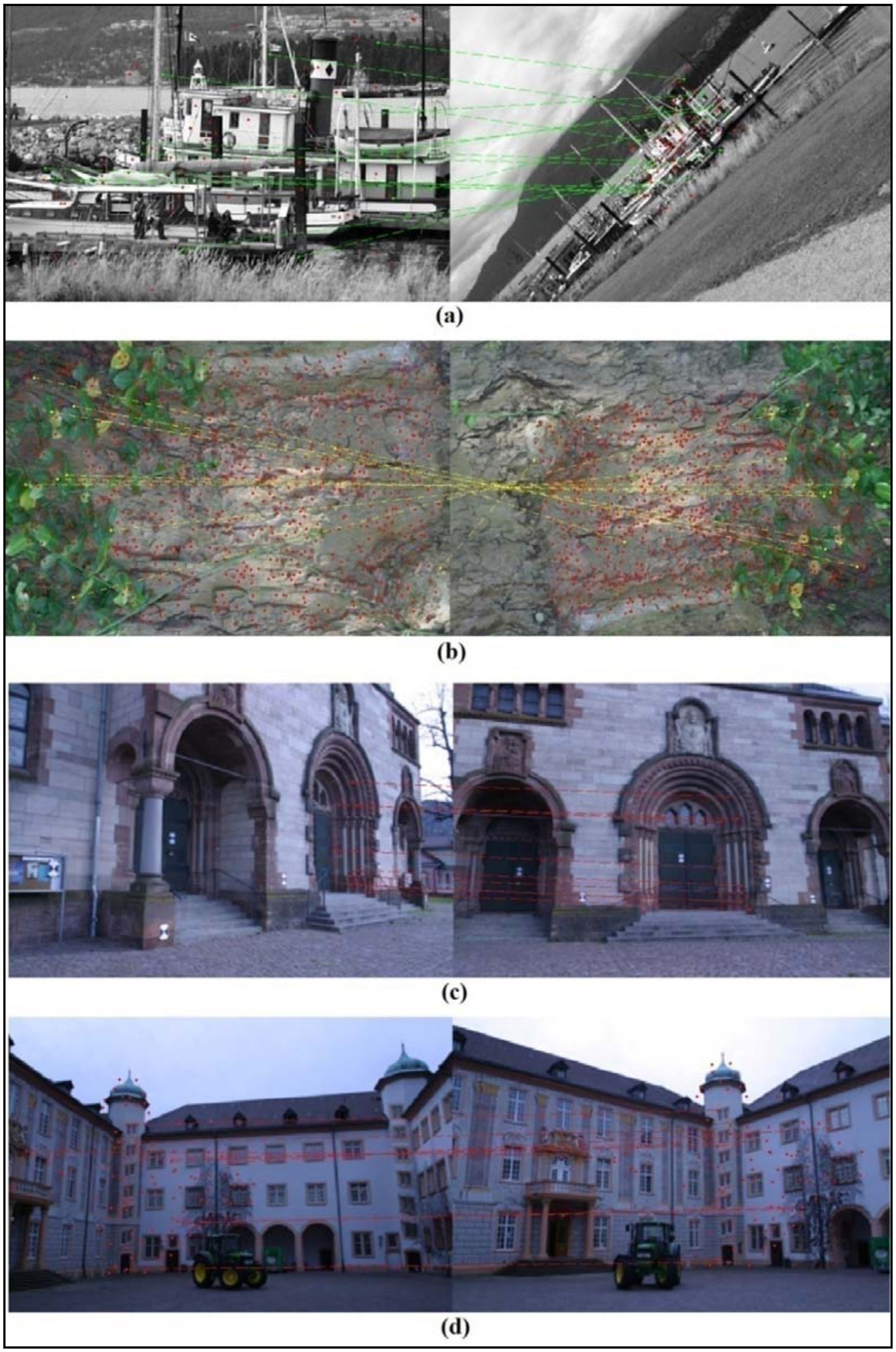


Figure 3.4. Demonstrate the result of finding correspondence points in the different views of objects or scenes. Since lines, which connect the correspondence points, may block the image, only some lines are drawn.

We have also evaluated our method on images of similar objects but with in-class variation. Figure 3.5 shows two different views of the same object (airplane).

The result of applying Lowe's matching algorithm is shown in Figure 3.5(a). Among the 51 correspondence points found, many of them are outliers and should be discarded. Figure 3.5(b) shows the refinement achieved by RANSAC 8-points fundamental matrix. This method reduces 51 potential matched points to 20 refined one but ground truth data shows that only 9 correspondence points are obtained correctly and the rest (11 points) are wrongly indicated as correspondence points. The result of finding correspondence points using our method is depicted in Figure 3.5(c). Comparing to the ground truth data, our method calculated 18 true correspondence points out of 19 matched points found by it.

In the other comparison presented in Figure 3.6(c), it is seen that our method found 3 correct matches and 1 error over 108 potential correspondence points (Figure 3.6(a)); however, the RANSAC method totally failed, with 25 errors and only 3 correct matches (Figure 3.6(b)). Thus, its results are not reliable with such huge amount of errors.

We have measured the performance of the two methods using the mutual information criterion as is illustrated in Table 3.1. It is seen that our method totally outperforms RANSAC method based on this measurement.

Table 3.1. Comparison between our method and RANSAC 8-points fundamental matrix method on images of airplanes from different views.

Image NO.	Method	# Potential correspondence points	TP	FP	FN	TN	Mutual Information Index
Figure 3.5(b)	RANSAC	51	9	11	10	21	0.008
Figure 3.5(c)	Our method	51	18	1	1	31	0.496
Figure 3.6(b)	RANSAC	108	3	25	0	80	0.038
Figure 3.6(c)	Our method	108	3	1	0	104	0.106

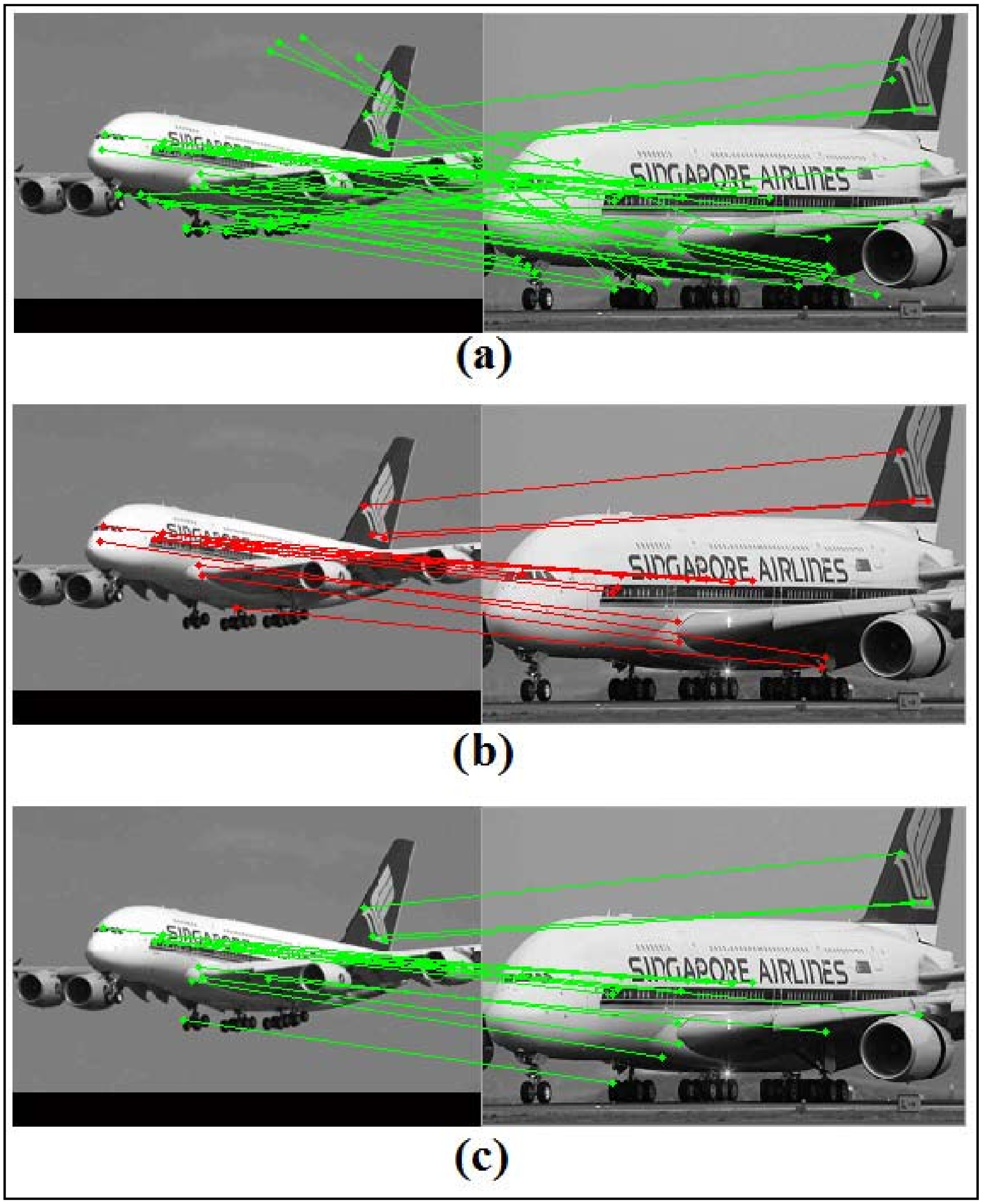
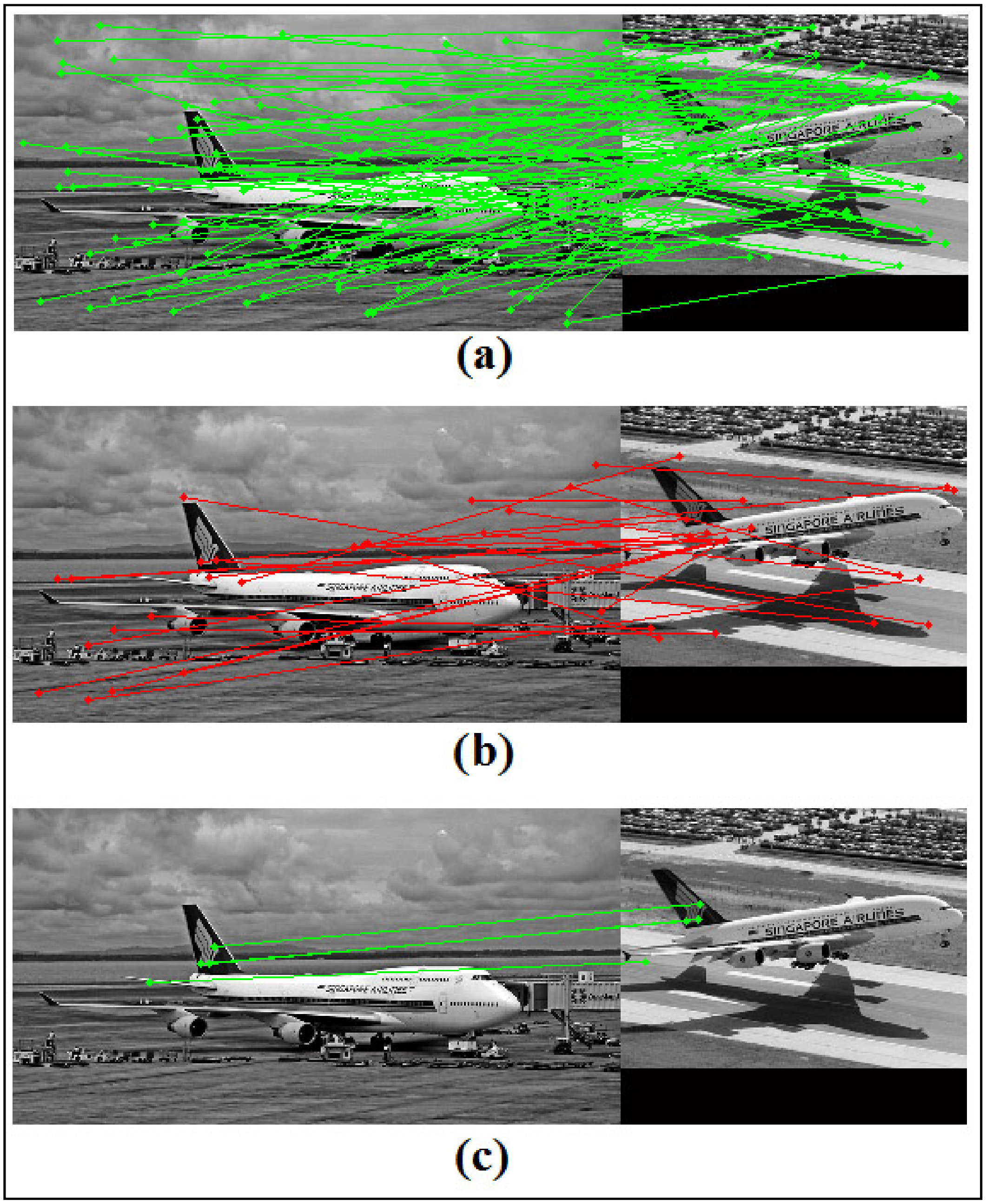


Figure 3.5. Shows the result of finding correspondence points of the same object in different views and compares RANSAC 8-point fundamental matrix method (b) to our method (c).



(a)

(b)

(c)

Figure 3.6. **S**hows the result of finding correspondence points in different views of the same object which have in-class variations using Lowe's matching algorithm (a), RANSAC 8-point fundamental matrix method (b) and our method (c).

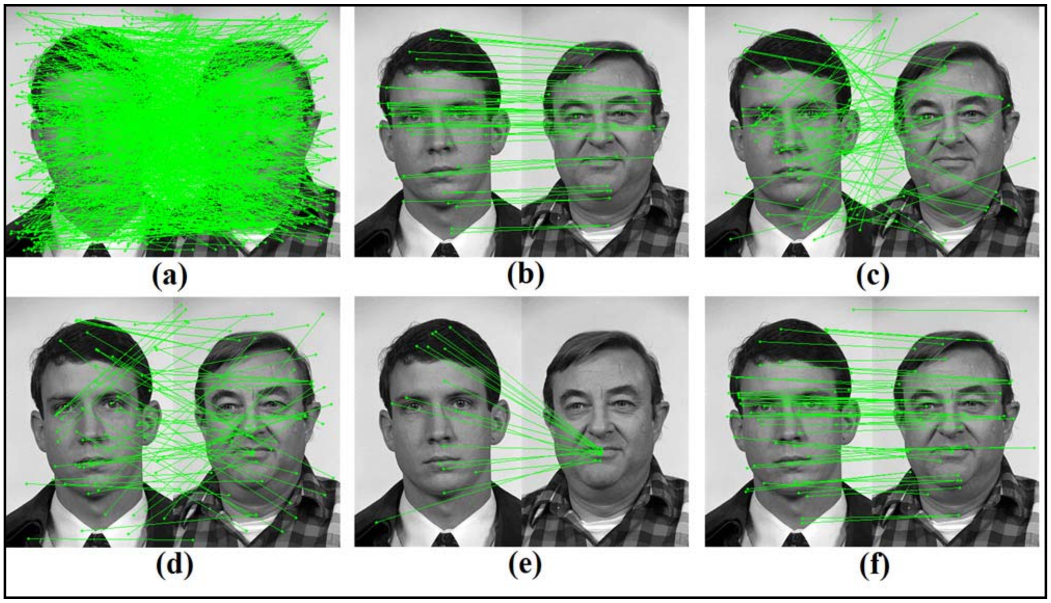


Figure 3.7. Compares the result of applying different methods to find real correspondence points between face images of different persons from frontal view. (a) shows the potential correspondence points between two images found by the Lowe's method [14]. (b) is the ground truth correspondence points found manually. (c), (d) and (e) present the results of applying RANSAC method with 8-pints fundamental matrix, affine fundamental matrix and Euclidean matrix, respectively. (f) demonstrates results of our proposed method.

3.3.4 Matching Different Faces from Multi-views

The proposed method is mainly designed to work with face images. We have applied the proposed method to some face samples from different views and compared its results with the results obtained from RANSAC method with different models. In Figure 3.7, different methods have been applied to find the real correspondence points between two face images of the same view (frontal view). Figure 3.7(a) demonstrates the potential correspondence points between image pair found by the matching algorithm proposed by Lowe [14]. Since Lowe's method only considers the feature appearance to find the correspondence, many found correspondence points are not correct correspondences. Hence, we have manually selected the real correspondences between these two images among all the potential matches as shown in Figure 3.7(b). We have tried to find the real correspondence points among the potential correspondence points by applying RANSAC method with 8 points fundamental matrix model and affine fundamental matrix model in Figure 3.7(c) and Figure 3.7(d), respectively. But both methods failed due to the low number of real correspondence points. In Figure 3.7(e) RANSAC method with Euclidean matrix proposed in [72] has been applied to the potential correspondence points in Figure 3.7(a). This model is very close to the affine transformation matrix we used in our method but the method we have used for determining the model parameters is different from RANSAC method; therefore its outcome is not as reliable and accurate as our method. The result of our proposed method is shown in Figure 3.7(f).

Given the same duration to each method, the number of correct correspondence points and wrong correspondence points found by them with respect to the ground truth data and precision and recall curves are presented in Figure 3.8. It is seen that after a short period of time, our method reaches a stable result but other methods cannot estimate a stable model for the real correspondence points. We have also compared the performance of different methods using mutual information index. The results in images of the same view (Figure 3.8) shows that based on this measurement our method totally outperforms other method with a very high index value compared with others. RANSAC method works very well when the number of real correspondence points between two comparing images is large with respect to the all potential points. In fact it is good for removing the gross error. In the case that the number of real correspondence points is a small portion of the overall potential points, RANSAC method fails [32].The other problem with RANSAC method is the time consumption. Since it selects data randomly, sometimes it takes a long time to estimate a model for the data. Recently Chum and Matas [60] tried to optimize the sample set selection to reduce the running time. However, random selection increases the time. The correct setting of parameters such as the minimum number of samples in the selected clusters and maximum number of iteration, are some other issues which affect time and efficiency of RANSAC method significantly.

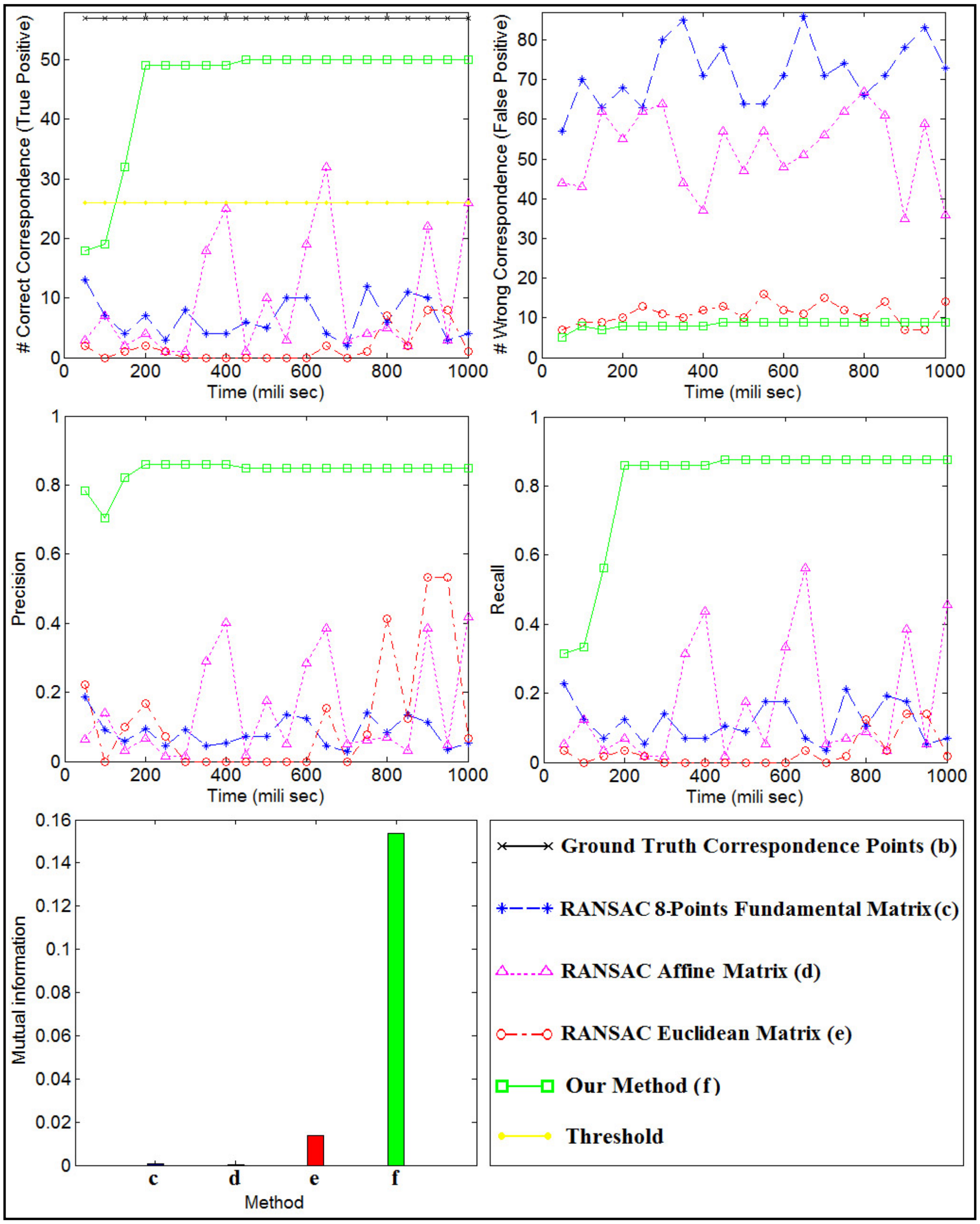


Figure 3.8. Compares the stability and performance of different methods to find real correspondence points between face images of different persons from frontal view.

We examined our method with face images from different views to make sure that the proposed method can be used for multi-view face detection framework. Figure 3.9 shows the results of applying different methods to find the real correspondence points

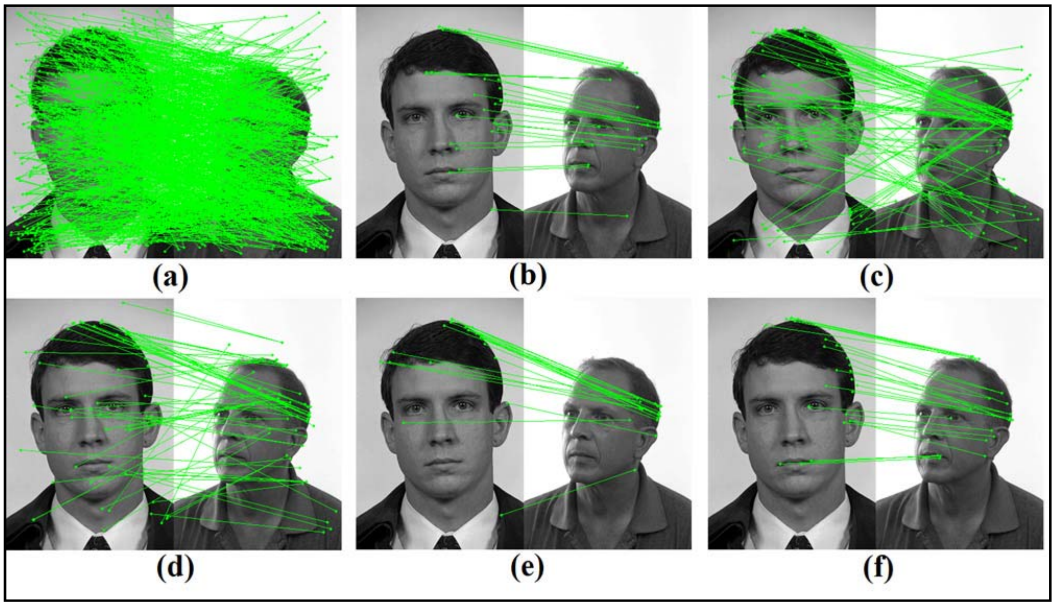


Figure 3.9. Compares the result of applying different methods to find real correspondence points between face images of different persons from different views. (a) shows the potential correspondence points between two images found by the Lowe's method [14]. (b) is the ground truth correspondence points found manually. (c), (d) and (e) present the results of applying RANSAC method with 8-pints fundamental matrix, affine fundamental matrix and Euclidean matrix, respectively. (f) demonstrates results of our proposed method.

between one image from frontal view and one randomly selected face image from oblique view. It is seen that only our method (Figure 3.9(f)) is able to return some real correspondence points between comparing images. The true positive, false positive, precision and recall graphs versus time are plotted in Figure 3.10 for the correspondence points found using different methods between two face images presented in Figure 3.9. As before, the same results are obtained for our method compared to the previous case studies (frontal faces) but with lower Precision and Recall values.

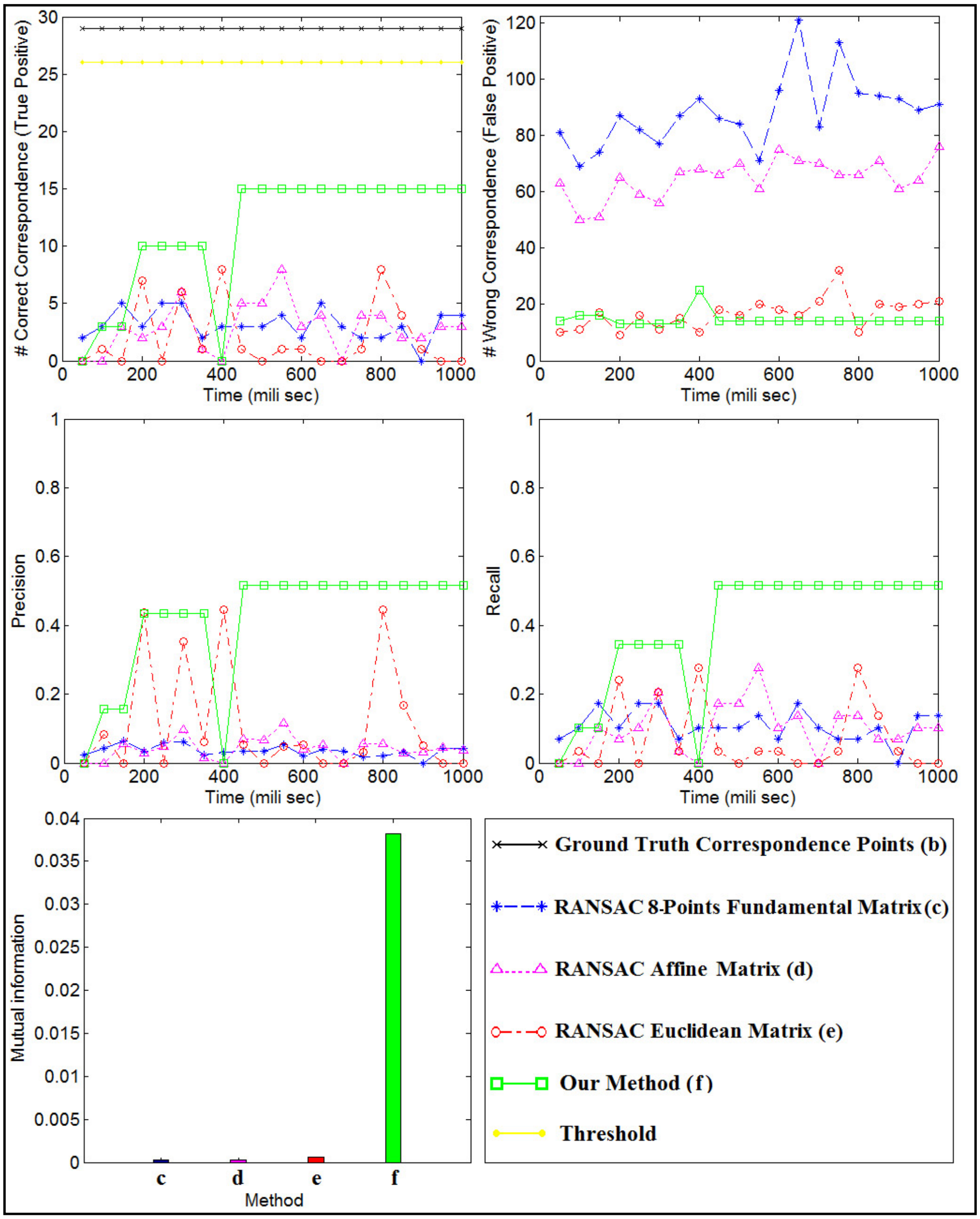


Figure 3.10. Compares the stability and performance of different methods to find real correspondence points between face images of different persons from different views. Given the same time to each method, the number of correct, wrong correspondence points precision and recall with respect to the ground truth data are obtained and sketched.

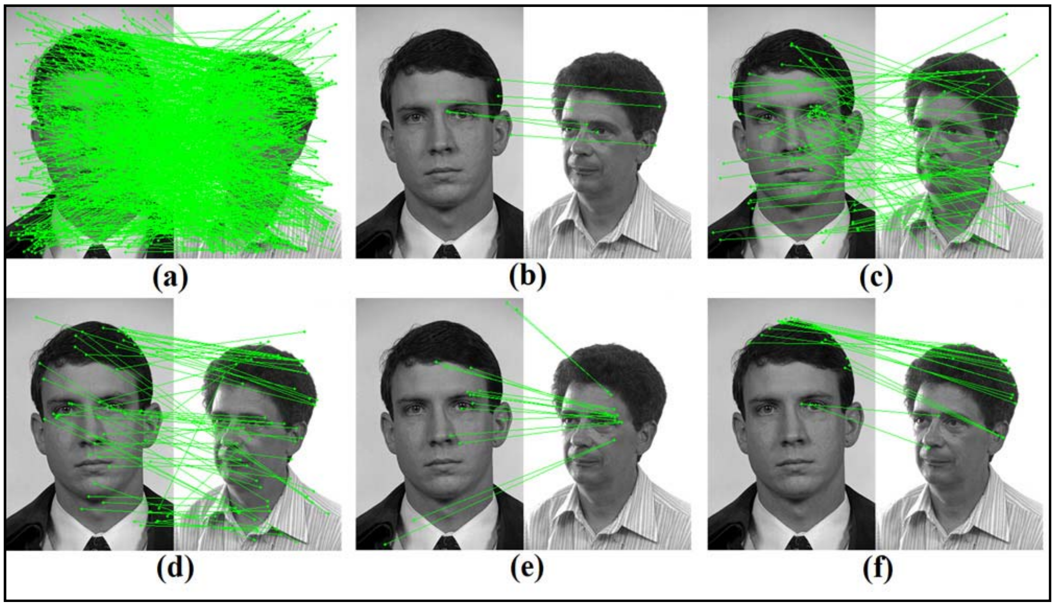


Figure 3.11. Compares the result of applying different methods to find real correspondence points between face images of different persons from different views. (a) shows the potential correspondence points between two images found by the Lowe's method [14]. (b) is the ground truth correspondence points found manually. (c), (d) and (e) present the results of applying RANSAC method with 8-pints fundamental matrix, affine fundamental matrix and Euclidean matrix, respectively. (f) demonstrates results of our proposed method.

The test was repeated for another randomly selected face image from oblique view as demonstrated in Figure 3.11. In this case study, the appearance of two comparing images are very different. Therefore, the number of real correspondence points found from all the potential points are quite low. Hence, our method could not find sufficient real correspondence points. The true positive, false positive, precision and recall curves for this case are presented in Figure 3.12.

Although all methods failed when the number of true correspondence points was very low, our method has still an advantage compared to the others as it can return NULL results in this scenario whereas others only return some false positive results.

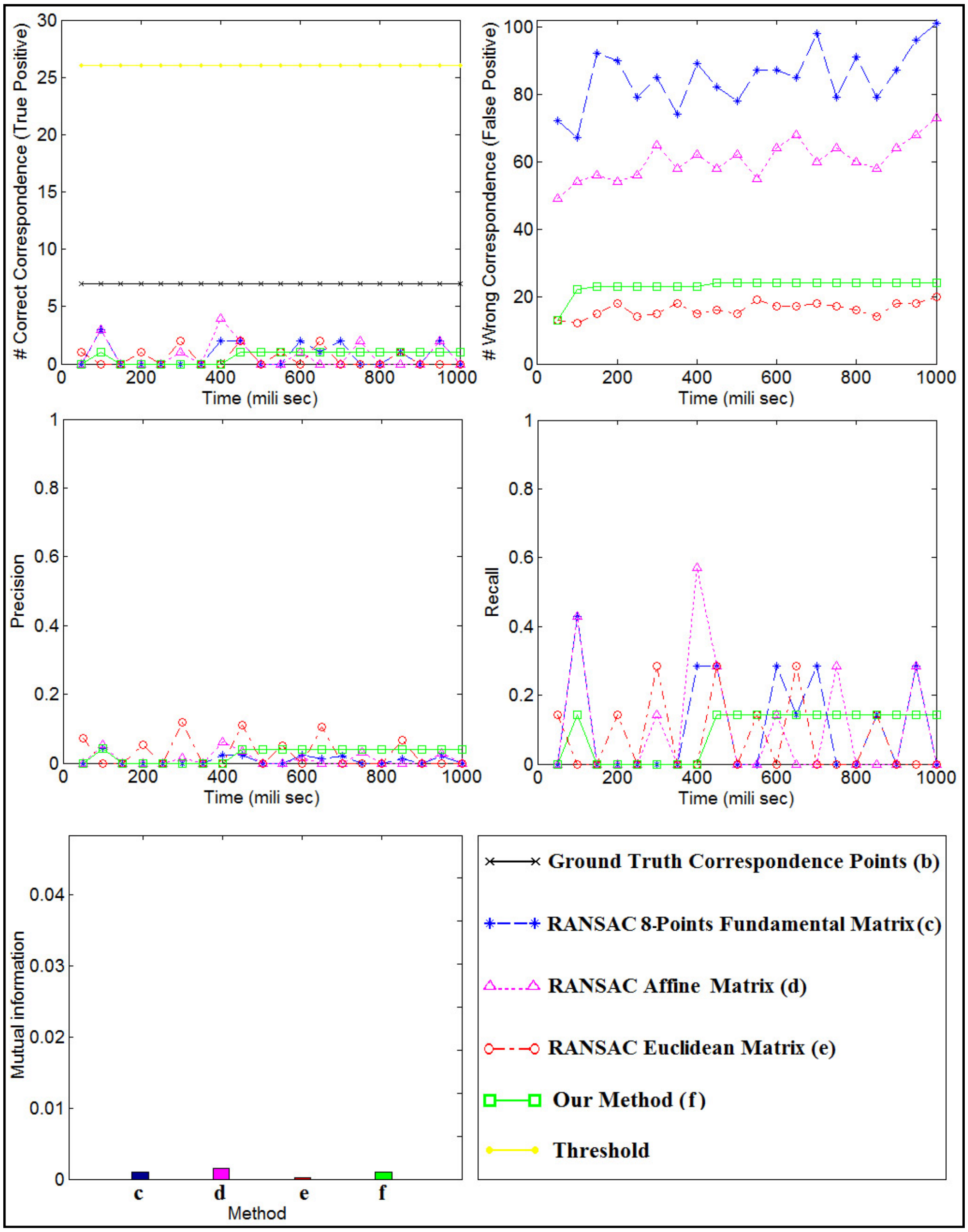


Figure 3.12. Compares the stability and performance of different methods to find real correspondence points between face images of different persons from different views. Given the same time to each method, the number of correct and wrong correspondence points with respect to the ground truth data are obtained and sketched.

This could be happen if we set a threshold on the total number of true positives and false positives correspondence points found by our method and if it is less than this threshold, our method returns a NULL. This threshold is shown with a yellow dotted line in true positive graphs on Figure 3.8, Figure 3.10 and Figure 3.12. It is seen that in all successful cases (Figure 3.8 and Figure 3.10) the summation of the green squared lines in both true positives and false positives are greater than the yellow dotted line. However, for Figure 3.12 it is less than this line.

To check the applicability of the proposed matching algorithm to other scenario, we have repeated our experiment for 500 different face images from multi-views over 500×499 different permutations and image combinations. It confirmed our findings that as long as the number of real correspondence points is above the threshold, our method is able to detect them without significant number of false matches. Thus, our method is suitable for finding real correspondence points between face images of multi-poses so long as the changes in the pose are not extreme.

3.3.5 Computation Time

Although the procedure presented in Algorithm 3.1 seems rather complicated, the time consumption for our proposed method is not significantly higher than the existing approaches, especially since it converges quickly instead of falling into continuous loop which is exhibited by RANSAC. For instance, we have extracted about 1000 SIFT features for each of the image pairs in Figure 3.7. After applying the Lowe's matching method, which took about 114 ms, we applied our method to refine the 987 potential correspondence points shown in Figure 3.7(a). The computation time for our method to search among 987 potential correspondence points and return 39 real correspondence points (see Figure 3.7(f)) was 62 ms based on C++ implementation on an ordinary PC (3.1GHz Ci5 processor with 4 GB RAM and Windows OS, using only a single core). However, when we applied the RANSAC method with any model, it ran into infinite loop and we have to explicitly terminate the process instead or to indicate a hard threshold on the number of iterations allowed..

Please note that the time indicated in the graphs in Figure 3.8, Figure 3.10 and Figure 3.12 are measured based on Matlab code implementation, which is higher than C++ implementation.

3.4 Concluding Remarks

A matching refinement algorithm which is an enhanced version of the existing approaches has been presented in this section. The approach uses both appearance and geometric information besides the discriminative features to find the real correspondence points between two images. The presented method is able to find the correct matches even if the number of true correspondence points between images is very small compared to all the found correspondence points.

It is seen that when the number of real correspondence points between two images are greater than a threshold, the proposed method is able to find many of these points with high Precision and Recall. However, when the number of real correspondence points is significantly lower, the Precision and Recall values obtained by the proposed method will be quite low and the results obtained are not reliable. However, it has the advantage that it can converge and returns a NULL instead of being trapped in an infinite loop if it could not find any real correspondence points.

Chapter 4

Face Registration and Face Model Construction from Multi-Poses

4.1 Introduction

The characteristic appearance of a normal or global face in all people is the same. All faces contain two eyes, a nose, lip, and mouth and the relative position of each feature is quite constant with respect to the other feature (the golden ratio for face). This property is key for most face detection methods from frontal view. Global features such as eigenfaces [46] are not suitable for face detection as they are sensitive to view variations and face rotation. Local features are more robust than the global features with respect to view rotation and variation. However, selecting large part of the features is also an issue as the features vary from one person to another. For instance, the shape of the eyes is not consistent in all people, especially people of different ethnicity. To overcome such problem, we proposed using image registration and *reference map*. Although a specific region of a face may not be similar in all faces, we can find some similar patches in a cluster of face images and some other similar patches in the other clusters. If a large variety of face images are registered to a *reference map*, we obtain a model that includes most patches of faces in different appearance and possible variations.

Image registration is the task of overlaying different images of the same scene, which may differ in scale, illumination condition and point of view. It is widely used in many computer vision related applications such as remote sensing and medical image processing.

In this chapter, we aim to register the face images of different people captured from arbitrary views. We assume that the images used for registration contain only one face. It is a crucial step because face images can differ in scale, illumination condition and head position. A *reference map* is shaped based on a *reference image* and all the images of different views and scales are then registered to it via their links to the *reference image*. The registration process would be more accurate if the number of images increases. This technique results in a coarse registration. Fine tuning the registration is achievable by applying some fine tuning methods such as that described in [73]. Fine tuning methods require the input image to be registered to some degree and a few correspondence points to be known. Face fine tuning registration is out of scope of this research and for the face model construction which will be discussed in the next chapter, only coarse registration is required.

4.2 Multi-View Face Registration with Intra-Class Variation

Image registration process usually consists of four essential steps. First salient features are detected in all comparing images. Then, the correspondences between these features are found. Based on these correspondence points, a transformation model can be estimated to best describe the relationship between the images. In a case of a linear transformation, estimating this model can be reduced to calculating only the translation, rotation and a few other parameters depending on the type of transformation. Finally, images are re-sampled and transformed to the *reference map* using the transformation matrix obtained. Although the procedure seems clear, proper implementation of each stage has its own problems and difficulties [74].

In order to register face images to a *reference map*, firstly we need to find the distinctive correspondence points between all the images which could participate in the registration process. This procedure is described in full detail in the previous chapter.

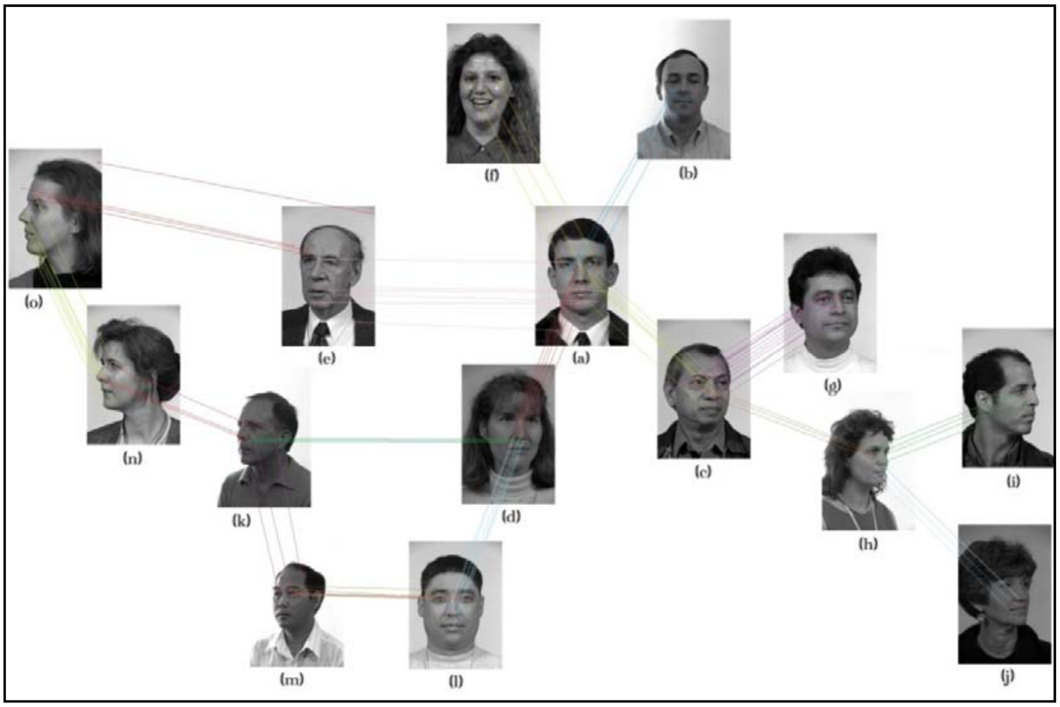


Figure 4.1. Illustrate the constellation connection between random selected images of different views and scales.

The next task is to define the *reference map*, initialize it in order to map it to a reference face and align the various face images differing in position, scale, and pose to the reference face. This is needed to allow computation of a similarity measure in the overlap region so as to establish whether the detected region is indeed a valid face. Given any two images, the correspondence points obtained serve as links between these images. Each image has links to its neighborhood images and each of the neighborhood images will link to its neighborhood images as well. With sufficient training images, one or more images will eventually link to the *reference image* registered to the *reference map*. If the number of images is large enough, it is possible that all the images have connection to one another through a path among the links. The connection among these face images form a constellation connection as illustrated in Figure 4.1. The constellation allows relationship to be established among the training face images and the *reference image*.

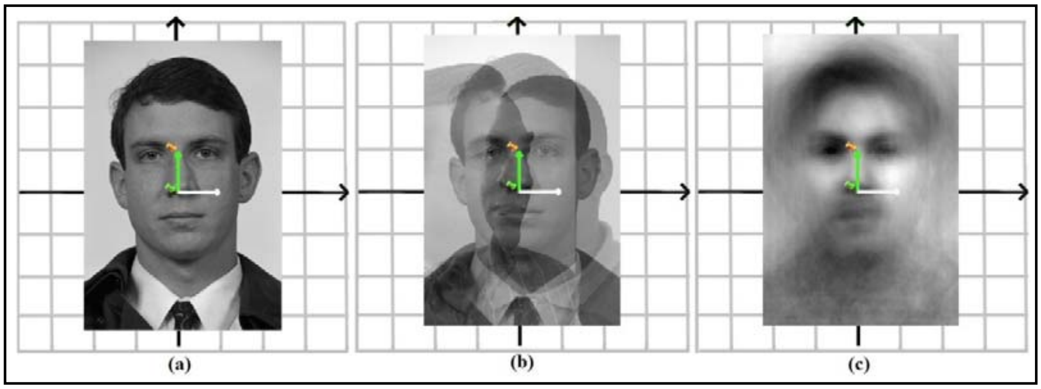


Figure 4.2. *Reference image and reference map* (green & white vectors) are illustrated in (a). In (b) images of left oblique and right profile are registered to the *reference map*. After registering all the images, which have a link to the *reference image*, to the *reference map*, the final model (c) is constructed.

The *reference map* is constructed from one *reference image* using two control points (also called *reference points*) whose positions are known as shown in Figure 4.2(a). A *reference map* encodes the information about the position, scale, and orientation of the reference face in the form of unit vectors (represented by the green and white arrows in Figure 4.2(a)). It is recommended that an un-occluded frontal face image with sufficient resolution and minimal background clutter be used as the *reference image*. The two control points are selected manually in the *reference image*. They should be representative and able to withstand pose variations. In all our experiments, the nose base and upper part of nose in the middle of the two eyes are selected as the control points (see Figure 4.2(a)). The *reference image* is pinned to the *reference map* using these two control points.

Then the next image that contains some correspondence points with the *reference image* containing the two control points is registered to the *reference map* using the geometric transformation given by the correspondence points. With two images pinned to the *reference map*, a third image that has correspondence points to either the *reference image* or the second image is then registered. The process is then iterated until all the images having correspondence to at least one image being registered is processed. As more images are being registered to the *reference map*, the possibility that another image can be registered increases. Thus, face images of different poses

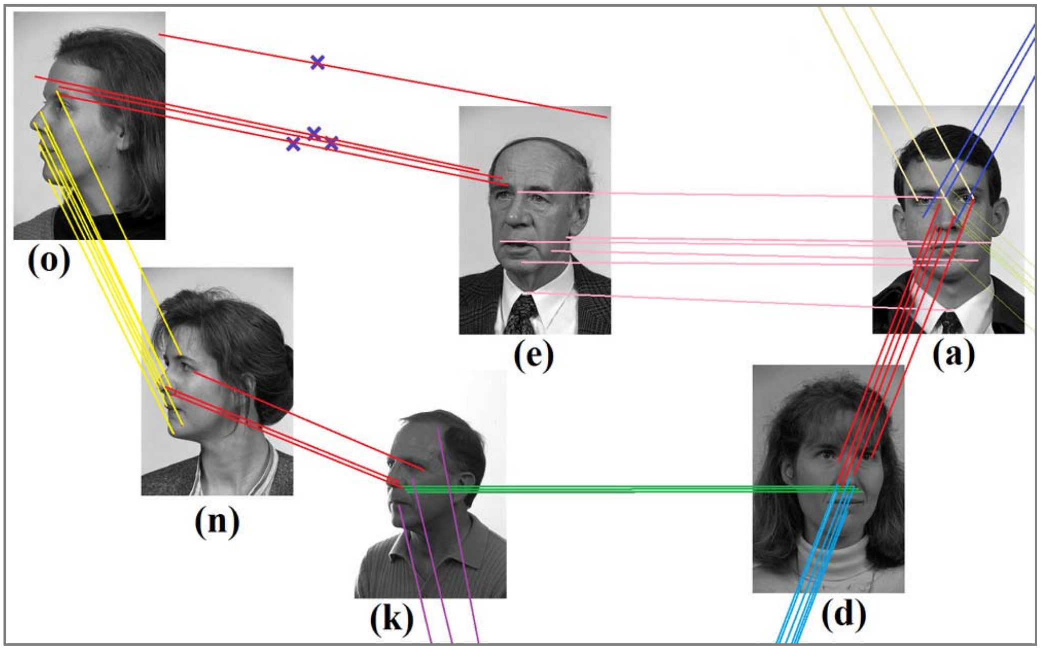


Figure 4.3. Magnifies the weak links between two images of different views ((o) and (e)) and shows the alternative route for (o) to connect to the *Reference image* through (o)(n)(k)(d)(a) links .
.

can eventually be processed, with each major pose forming a node in the constellation graph as shown in Figure 4.1. As such, the ability of the face detector to handle pose, expression and illumination variation depends only the availability of training face images with pose, expression, and illumination variations in the database. Therefore, there is no limit on the type of facial variations that the system can cope with, as long as the features used to establish the correspondence are robust to some degree of local variations encountered in both images to be matched.

4.2.1 Error Propagation Problem

As a face image is registered to the *reference map* based on the correspondence points found between the face with another face image already registered to the *reference map*, any error in the correspondence will result in the error being propagated as more faces are being registered to this link. Consider the constellation connection illustrated in Figure 4.3 (which is a magnified part of constellation connection illustrated in Figure 4.1) and the *reference image* shown in Figure 4.3(a). The face image in Figure

4.3(e) is connected to the *reference image* through its linkages with that image. If face image in Figure 4.3(o) is connected to the *reference image* through its linkages with the face image in Figure 4.3(e), it would be registered wrongly because the links between Figure 4.3(e) and Figure 4.3(o) indicated by crosses do not establish a correct correspondence points. This error would be propagated through any other images that use the link to Figure 4.3(o) for registration to the *reference map* such as image Figure 4.3(n). Thus we need to consider a mechanism to avoid such weak connections such as those between the face images in Figure 4.3(e) and (o). When the weak connections are removed, images can find other links to be registered to the *reference map*. For instance, after removing the connection between the face images in Figure 4.3(e) and (o), the face image in Figure 4.3(o) can be registered to the *reference map* through the chain {(o),(n),(k),(d),(a)} which results in correct registration.

To avoid such error propagation, we proposed a mechanism to remove the weak links from the constellation since such links have higher chances of error. We noticed that weak links usually arise from correspondences found on the non-face region. Let $F = \{f_k\}$, $k = 1,2, \dots, K$ be the collection of feature points in a given image β, which have participated in establishing the connection between the image and other images in the constellation. K is the maximum number of distinctive features detected. To detect the weak links, the likelihood ratio $\vartheta(f_k)$ is computed for all features (f_k) in F as shown in equation (4.1). Let H_k denote a binary random variable indicating whether the feature f_k originates from a part of face $(H_k = 1)$ or not $(H_k = 0)$. This likelihood ratio should be greater than one for valid face features as shown in equation (4.1).

$$\vartheta(f_k) = \frac{p(H_k = 1|\psi)}{p(H_k = 0|\psi)} > 1 \tag{4.1}$$

This likelihood ratio is computed from a sequence of L independent observations of data ψ when forming the constellation. Let feature f_k be a binary random variable that might originate from a part of face $(H_k = 1)$ or non-face $(H_k = 0)$ area. We can assign a Bernoulli distribution [69] with parameter θ to it that determines its uncertain value as (4.2).

$$P(H_k|\theta) = \theta^{H_k}(1-\theta)^{1-H_k}, \tag{4.2}$$

Assuming a sequence of L independent observation of data ψ over feature f_k in the training data, the likelihood of obtaining ψ is given by (4.3).

$$P(\psi|\theta) = \prod_{j=1}^{L} \theta^{H_k^j}(1-\theta)^{H_k^j} = \theta^{L_k}(1-\theta)^{L-L_k}\,, \tag{4.3}$$

where L_k is the number of time that feature f_k has represented a face area in the observation data ψ. When L is large, we can estimate the distribution parameter θ using MLE estimation as (4.4).

$$\hat{\theta}^{MLE} = \underset{\theta}{argmax}\, P(\psi|\theta), \tag{4.4}$$

However, if L is small, the MLE will produce a bias outcome and there is also the sparse data problem. We use the Bayesian estimation as even though the number of times a feature establishes a correspondence between two images in the constellation is low (typically < 50 in our experiment), the number of valid correspondences is higher than the noise after the wrong correspondences have been removed as described earlier. Therefore, we consider uncertainty on the distribution parameter θ and use a *Beta* probability distribution as a prior with parameters α_1 and α_1 as indicated in equation (4.5).

$$P(\theta|\alpha_1,\alpha_0) = \frac{\Gamma(\alpha_1+\alpha_0)}{\Gamma(\alpha_1)\Gamma(\alpha_0)}\theta^{\alpha_1-1}(1-\theta)^{\alpha_0-1}, \tag{4.5}$$

where $\Gamma(x)$ is the gamma function as denoted in (4.6).

$$\Gamma(x) = \int_0^\infty u^{x-1}e^{-u}du, \tag{4.6}$$

Considering the Bayesian rule, the posterior probability of the distribution parameter θ can be calculated in the form of prior probabilities as (4.7).

$$P(\theta|\psi) = \frac{P(\psi|\theta)P(\theta|\alpha_1, \alpha_0)}{P(\psi)}, \tag{4.7}$$

Substituting (4.3) and (4.5) into (4.7) yields another *Beta* distribution with new hyper-parameters in (4.8).

$$\begin{aligned} P(\theta|\psi) &= \frac{1}{P(\psi)}\theta^{L_k}(1-\theta)^{L-L_k}\frac{\Gamma(\alpha_1+\alpha_0)}{\Gamma(\alpha_1)\Gamma(\alpha_0)}\theta^{\alpha_1-1}(1-\theta)^{\alpha_0-1} \\ &= Beta(\theta|L_k+\alpha_1\,, L-L_k+\alpha_0), \end{aligned} \tag{4.8}$$

Thus, the posterior prediction for random variable H_k, which indicates if the feature f_k comes from a part of face, is given by (4.9).

$$\begin{aligned} P(H_k=1|\psi) &= \int_0^1 \theta\, P(\theta|\psi)\, d\theta \\ &= \int_0^1 \theta Beta(\theta|L_k+\alpha_1, L-L_k+\alpha_0)\, d\theta \\ &= \frac{L_k+\alpha_1}{L+\alpha_1+\alpha_0} \end{aligned} \tag{4.9}$$

where L is the total number of images used to form the constellation (considered as the training stage) and L_k is the number of images in the constellation where feature f_k can establish connection to the image β. For cases where feature f_k comes from the non-face region, we consider it as a white noise with average value $\overline{w}$. This implies that the average number of times it can make connection with different images is $\overline{w}$. This value cannot be large since noise features have random appearance and do not have strong connections between many images given the appearance similarity and geometric constraints. Thus, its likelihood probability can be computed in the same way as (4.8) which is shown in (4.10)

$$P(H_k = 0|\psi) = \frac{\overline{w} + \alpha_0}{L + \alpha_1 + \alpha_0}, \tag{4.10}$$

Substituting (4.9) and (4.10) into (4.1) indicates that the number of times the face feature f_k has participated in establishing connection with the images in the training set should be greater than a threshold $(L_k > (\overline{w} + \alpha_0 - \alpha_1) = Thr_w)$ for it to be considered as a valid face feature. Otherwise, it is a weak link and should be discarded. In setting the threshold Thr_w, a small value will increase the noise while a large value will unnecessarily remove valid features. We empirically observe that a value of 5 is appropriate for Thr_w. Alternatively, if the likelihood ratio in (4.1) is ranked, then it is not necessary to set the threshold Thr_w . After finding the distinctive correspondence points among all the images in the training set and computing the likelihood ratio in equation (4.1); for each of the correspondence points, the links between image pairs can be ranked in descending order. We can then start to establish the connection between images that have higher degree of match. Once the connection is successfully established between one image and the *reference image*, the other connections can be discarded. Thus, the number of weak links will be negligible.

4.3 Feature Selection and Face Model Construction

Once all the possible images have been registered to the reference map, the model construction starts with assigning a new normalized geometric descriptor $(f_i^G)^N = (x_i^N, y_i^N, \sigma_i^N, \phi_i^N)$ to each feature which is participated to make connection between images in the constellation connection (N represents the normalized space which implies that the normalized data and not the raw one is used). Whenever all the features are normalized, they shape a primitive model with thousands of features, depending on the number of training images and the number of features extracted per image. Not all these features are distinctive. Therefore, a mechanism should be considered to remove the redundant features. We used likelihood ratio test in (4.1) and evaluated it for every feature in the primitive model to find the most distinctive features. An intermediate model is formed by applying a threshold to the ratio in equation (4.1) and pruning the features with low probability. This procedure can reduce the number of features by approximately a factor of two to three. However, these amounts of features in the intermediate model still contain many redundant features. Therefore, a clustering algorithm is applied to reduce the model features further. Clustering begins with shaping a binary matrix χ from similar-appearance features in the neighborhood area of each feature in the intermediate model. The neighborhood area of each feature is considered as a circle region with a radius equal to 0.5 since the geometry of all the features in the model is normalized to the reference map before. Cell $\chi(i,j)$ denotes the similarity value between feature f_i and f_j as illustrated in (4.11).

$$\chi(i,j) = \begin{cases} 0 & f_i \ \ is\ not\ similar\ to\ \ f_j \\ 1 & f_i\ \ is\ similar\ to\ \ f_j \end{cases}, \tag{4.11}$$

A feature f_i is considered to be similar in appearance to the feature f_j if the Euclidean distance between their appearance descriptors f_i^A and f_j^A is less than a threshold T_i^∂. This threshold is not a global value and should be determined individually for every feature in the constellation. In the registration stage, if a feature $f_{i,k}$ from image i is matched to the feature $f_{j,l}$ from image j, the appearance threshold $T_{i,k}^\partial$ for feature $f_{i,k}$ is the Euclidean distance between $f_{i,k}^A$ and $f_{j,l}^A$. If $f_{i,k}$ from image i is involved in many matched pairs, the appearance threshold $T_{i,k}^\partial$ is given by the maximum Euclidean distance between $f_{i,k}^A$ and the appearance descriptors of the other features that match with $f_{i,k}$.

The number of similar features to the feature f_i are computed by summation of the rows in the matrix χ. These values indicate the similarity value $\pi(f_i)$ for feature f_i as illustrated in (4.12).

$$\pi(f_i) = \sum_{j=1}^{N_f} \chi(i,j) \tag{4.12}$$

where N_f is the number of features in the model. The similarity members values $\pi(f_i)$ are sorted in descending order and from the top of the list downward $(i = (1,2,\ldots,N_f))$. The median for each group is taken to represent the descriptor of that group. Each feature in the intermediate model can only be a member of one group. Therefore, if a feature was a member in multiple groups, that feature is considered as a member in the group with the maximum number of members and its membership to the other groups are discarded.

Applying the clustering algorithm to the intermediate model reduces the number of features by approximately another factor of ten to twenty, resulting in about less than a thousand features in the final model. This model is used to detect faces from arbitrary view to be discussed in the next chapter. This stage is very crucial because removing the redundant features not only speeds up the processing time but also helps improve the detection accuracy.

4.4 Experimental Result and Discussion

In this Section, the proposed framework for face registration has been evaluated using the standard color FERET dataset [66, 75, 76]. FERET dataset contains images of 1,209 persons of various age, gender and ethnicity. Each person has an average of 10 images of arbitrary view and in different illumination conditions. The number of images in each viewpoint is not uniform. For instance, a person in the dataset may contain more than three frontal faces but only one full right profile. We resize all the images to 256x384 pixels and convert the color images into gray scale.

We randomly choose 500 images of different people from the FERET dataset, with one image per person of random view, to form the initial training set. Then, the SIFT features [14] extracted from the training images and the proposed method outlined in the previous chapter is applied on all the image-pairs to find the distinctive correspondence points between them. The SIFT parameters such as number of octaves of the Gaussian scale space and number of scale levels within each octave are determined such that at least 1,000 features are extracted from each image. After finding some reliable correspondence points between the image-pairs, face registration algorithm is applied to register face images to the *reference map*. We took the first image with frontal view as the *reference image* and manually selected the nose base and upper part of nose in the middle of two eyes as the reference points for registration as depicted in Figure 4.2(a). Marking these two points is the only manual intervention needed. Starting with 50 images as training images to form the initial constellation of links, we evaluated the percentage of images which were successfully registered, wrongly registered, and cannot be registered over the total number of images available in the training dataset. We then repeated this by increasing the number of training images until all the available images are used. Once all the possible images registered, a final model is shaped which is the projection of 3D model to the 2D space as illustrated in Figure 4.2(c).

The registration result is shown in Figure 4.4. We manually checked all the images to ensure correct registration and found that only 6% of the images are wrongly registered after all 500 images are added to the initial training set. Those images without any link to the *reference image* are classified as non-registered faces. The results confirm that as the number of training images is increased, the number of correctly registered images also increases because the diversity (number of images with different viewpoints) of face images in the constellation increases. Thus, the probability of finding a path to the *reference image* increases, resulting in successful registration. Figure 4.5 shows the cumulative plot of the accuracy of estimating the reference points in the 500 training images. The horizontal axis shows the error in estimating the position of the reference points, which is obtained by computing the Euclidean distance between the estimated reference points and the ground truth data (marked manually in all images). The vertical axis shows the cumulative percentage of images having estimation error less than the given number of pixels in the horizontal axis. The number of images is normalized by the total number of registered images (440 images out of 500 as shown by the solid lines in Figure 4.4). One can observe that our algorithm is able to estimate the reference points to within half of the average nose length in more than 90% of the images. About 97% of the images can be registered to

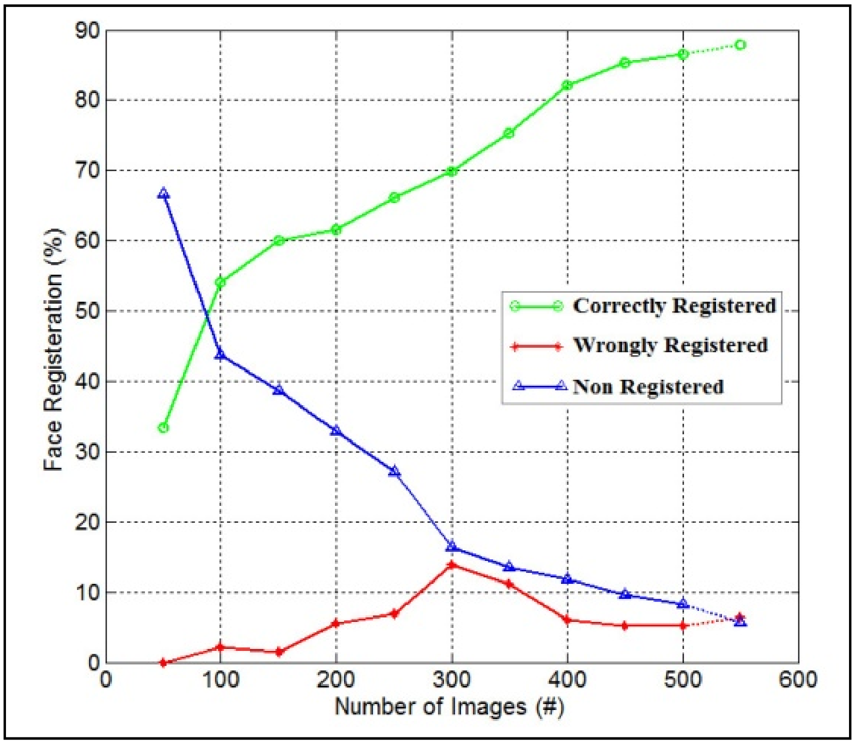


Figure 4.4. Shows the Percentage of face registration versus the number of images used. The percentage of correctly registered images increases when the number of images increases while the percentage of non-registered images decreases. The rate of wrongly registered images is kept low by the propagation error prevention algorithm. It is rather constant at about 6% after 400 images are used for training. This low error percentage does not affect the model construction significantly.

within the average nose length. Using half the average nose length as bound for successful face registration, the performance of our proposed approach is good considering that only one image is manually registered and the fact that there are large variations due to pose, ethnicity etc. In our implementation, the nose length is used to normalize the images.

4.5 Concluding Remarks

We have introduced a method to register faces of multi-poses into a predefined reference map. The correspondence points between the image pairs help us to determine the transformation between the images. Since all the correspondence points might not be reliable, an error propagation mechanism has been proposed to discard

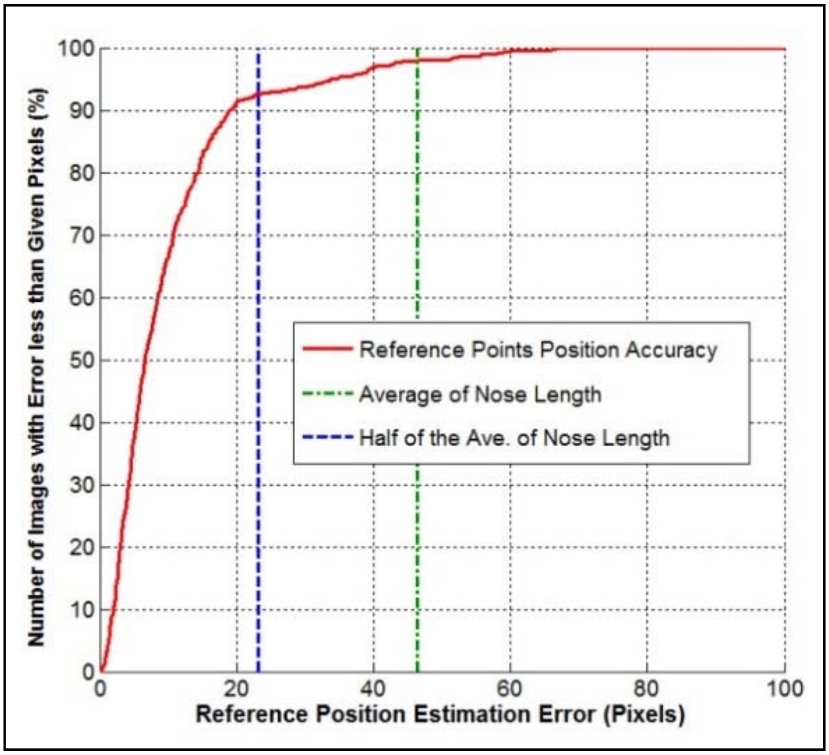


Figure 4.5. Cumulative plot of the reference point's location estimation accuracy. Analysis is done for all the images that are registered to the *reference map*, but excludes those that cannot be registered by our method.

the weak links between image pairs using the probability functions which determine the strength of the links. Once the weak links are removed, registration procedure would be performed that yields the minimum error in the registered faces. The features from the registered faces are then used to build a view invariant face model. Since faces from different poses have participated in the registration, the obtained model has the property of a 3D model projected onto the 2D space.

Chapter 5

Rotation Invariant Face Detection and Localization from Multi-Poses

5.1 Introduction

Face detection and localization are the basic part of face recognition that should be performed before any recognition can take place. Several studies have been conducted to solve the problem of detecting objects and faces from multi-view [2, 3, 40, 41]. Most of them [2, 3, 41] are based on single detectors in different combinations. A sliding window in different scales heuristically searches every pixel of given image to determine the existence of a face image. This procedure is repeated for every view of the query object using an ensemble of trained classifiers. Each single classifier trains with some images of a face, which are manually cropped to the facial area and labeled as face or non-face. However, training several classifiers is very time consuming and requires the training data to be labeled manually which is very difficult. In this chapter, a multi stage framework is proposed which requires no prior assumption about the training images. It uses the view invariant face model obtained as described in the previous chapters and utilizes a probabilistic classifier to detect and localize faces in more challenging scenarios such as high cluttered background or an image that includes many faces in different views. The proposed framework is also capable of estimating the in-plane rotation angle for the detected faces.

5.2 Generalized Face Detection and Localization Framework

In the previous chapters, a method for finding the reliable matches between different views of face images has been discussed. Then, arbitrary views of face images from different people were registered to a predefined reference map. A primitive face model is then constructed from face features which participated to make linkages between image pairs in the constellation. Using a clustering method and feature pruning, the final model is obtained from face images of multi-poses. Thus the previous chapter can be considered as the training part, and in this chapter we aim to find a method to detect faces and generalize the face detection algorithm to handle multiple faces of arbitrary views. In our proposed approach, once a given image is registered, it is possible to approximately locate the features belonging to specific facial traits. Thus, not only it detects the face, it also provides localization of the detected face which can be used to normalize the face. Therefore, to detect multiple faces, we need to cluster all the features that approximately form a face as a cluster.

The posterior probability function $P(\rho|F)$ gives the probability that the set of features $F = \{f_1, f_2, \dots, f_K\}$ completely register to a reference map ρ. Applying Bayes theorem gives the prior probability $P(\rho)$ and likelihood $P(\{f_1 f_2 \dots f_K\}|\rho)$ as shown in (5.1).

$$P(\rho|F) = \frac{P(\rho)P(F|\rho)}{P(F)}, \tag{5.1}$$

where $P(\rho)$ is the prior probability and $P(F|\rho)$ is the likelihood of observing the features $F = \{f_1, f_2, \dots, f_K\}$ given a reference map ρ. To simplify the evaluation of the likelihood term $P(F|\rho)$, we assume that the discriminative features of face images are independent. Hence, equation (5.1) can be re-written as (5.2).

$$P(\rho|F) = P(\rho) \times \prod_{k=1}^{K} \frac{P(f_k|\rho)}{P(f_k)}, \tag{5.2}$$

where $P(f_k|\rho)$ is the likelihood term which indicates the probability that the feature f_k is a part of a face (i.e., $P(H_k = 1)$). This term was previously calculated for all the features in the model using the equation in (4.9) as described in the previous chapter.

$$P(H_k = 1|\rho) = \frac{L_k + \alpha_1}{L + \alpha_1 + \alpha_0} \tag{5.3}$$

The term $P(\rho)$ in equation (5.2) is the prior probability of the reference map and is a constant scaling factor that can be factored in the formulation, and hence, not evaluated. The term $P(f_k)$ is the prior probability of face features. This term represents the spatial distribution of the face features around the reference map and can be modeled using a Gaussian 2D distribution given by (5.4).

$$P(f_k) = \frac{1}{2\pi\delta_x\delta_y} e^{-\left(\frac{(x_k - x_0)^2}{2\delta_x^2} + \frac{(y_k - y_0)^2}{2\delta_y^2}\right)} \tag{5.4}$$

where (x_k, y_k) is the position of feature f_k and (x_0, y_0) is the position of the center point between the two reference points. Here, δ_x and δ_y indicate the spread of the face features and they depend on the size of the face in the given image. In all our experiments, we approximate $\delta_x = \delta_y =$(1.5 × *reference map* scale).

During face detection, all features of the test image β are extracted using SIFT method. Each feature in the test image is compared against the features of all images in the face constellation. An image feature $f_{\beta,m}$ is considered to be similar in appearance to the feature $f_{i,k}$ from the i^{th} image in the constellation if the Euclidean distance between their appearance descriptors $f_{\beta,m}^A$ and $f_{i,k}^A$ is less than a threshold $T_{i,k}^\partial$. This threshold is not a global value and should be determined individually for every feature in the constellation. In the training stage, if a feature $f_{i,k}$ from image i is matched to the feature $f_{j,l}$ from image j, the appearance threshold $T_{i,k}^\partial$ for feature $f_{i,k}$ is the Euclidean distance between $f_{i,k}^A$ and $f_{j,l}^A$. If $f_{i,k}$ from image i is involved in many matched pairs, the appearance threshold $T_{i,k}^\partial$ is given by the maximum Euclidean distance between $f_{i,k}^A$ and the appearance descriptors of the other features that match with $f_{i,k}$.

Let F_m be the set of features in the constellation that have high appearance similarity to the image feature $f_{\beta,m}$. This process is repeated for all the features in test image β and the collection of all matching features in the constellation $F = \bigcup_{m=1}^{M} F_m = \{f_1, f_2, \dots, f_K\}$ is obtained. Each matching feature f_k is used to estimate a reference map in the test image. The estimated reference maps are clustered using a mean shift algorithm [43]. Other clustering methods that do not require apriori knowledge about the number of clusters can also be used. In addition, clustering reduces the processing time and comparison is only required against one feature per cluster instead of all features present. Let cluster χ have K_χ features. Each cluster center can be considered as a potential reference map (ρ_χ). Among these potential reference maps, the valid reference maps can be obtained as follows:

$$P(\rho_\chi | F) > Thr_F \tag{5.5}$$

which is reformulated using (5.2) as it is illustrated in (5.6).

$$\prod_{k \in \chi} \frac{P(f_k | \rho_\chi)}{P(f_k)} > \frac{Thr_F}{P(\rho_\chi)}. \tag{5.6}$$

The right hand side of equation (5.6) can be replaced by another constant threshold. While evaluating $P(f_k|\rho_\chi)$, we consider a uniform prior for *Beta* distribution ($\alpha_0 = \alpha_1 = 1$) in equation (4.9) and substituted L_k value for features in the constellation that have at least one match in cluster χ. For those features in the constellation that do not have any match in cluster χ, L_k is set to zero. The computed value for each cluster is then compared to a threshold to classify it as a face or non-face, thereby yielding multiple detected faces. This threshold is determined experimentally such that it minimizes the classification error.

The probabilistic framework also gives the ability to handle partially occluded faces and faces with high background clutter because it does not require finding features from all parts of a face. Occluded faces can still be detected if distinctive features from some parts of the face satisfy equation (5.6). Since the distribution of the features from the background clutter is usually more dispersed compared to the face, the likelihood that sufficient feature similar to a face existing in a cluster is rather low. Therefore, it is possible to detect faces even in the presence of cluttered background.

5.3 Estimating In-Plane Rotation Angle for Detected Faces

Window-based methods such as Viola and Jones [1], Li and Zhenqiu [39], Huang et al.[3] and Schneiderman and Kanade [2], are not able to handle rotated faces. Even the recently proposed method by Toews and Arbel [40] cannot detect them. To solve this problem, Rowley et al. [35] proposed to rotate the entire image repeatedly by small increments and apply the face detector after each rotation. The number of times required to rotate the test image depends on the degree of variation take the face detector can handle. For instance the famous face detector proposed by Viola and Jones [42] is able to handle face rotation with ±15° of in-plane rotation. Thus, they suggested using 12 different detectors in 12 different rotation classes to cover the full 360 degrees of rotation as it demonstrated in Figure 5.1. This procedure is very time consuming since we need to rotate the input image and apply the face detector several times to find the rotated faces. The chance of false detection also increases and the estimated angle from this method is coarse.

Our proposed face detection and localization framework has the capability to be extended to detect in-plane rotated face images with an amendment to the detection stage without any significant computation Once the in-plane rotated face is detected successfully, the system will provide an estimate of the rotation angle as well.

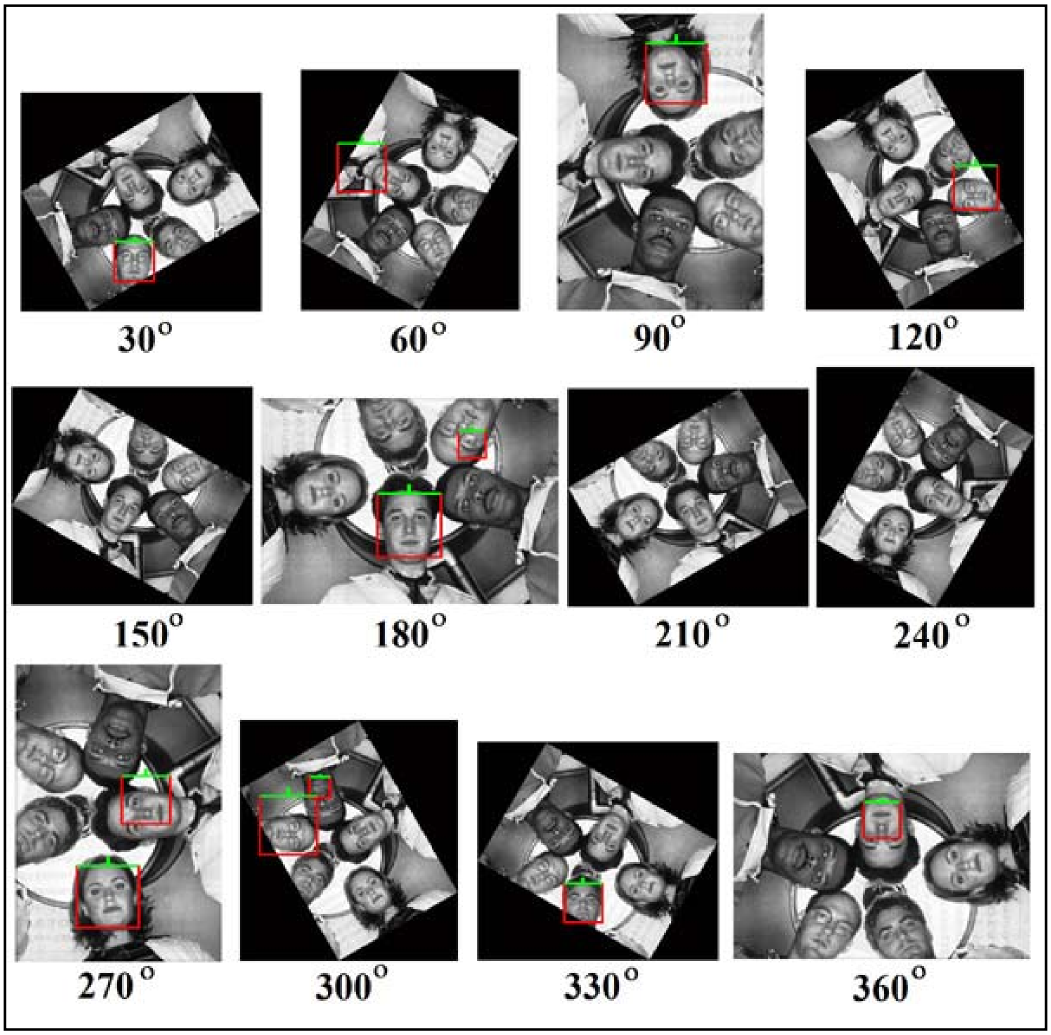


Figure 5.1. Sample of detecting rotated faces for an image in CMU database using Viola & Jones method [42]. Rotated faces are determined by rotating the input images in small increments and applying the face detector repeatedly.

In the test stage, all the extracted features from the test image are compared to the model to find the possible matches between model and the test image. Let a model feature f_m to be similar in appearance to the image feature f_i and matches it. This model feature f_m proposes two reference points (or a reference map) in the test image. These two points (as a vector) are rotated in the image plane with respect to the center of model feature f_m such that the orientation of the image feature f_i fits to the orientation of the model feature f_m .

Let vector $\vartheta(f_m) = \{X_m^M, Y_m^M\ , \sigma_m^M\ , \phi_m^M, X_m^{R1M},\ Y_m^{R1M}, X_m^{R2M}, Y_m^{R2M}\ \}$ be the x, y position, scale, orientation, first reference x, y position and second reference x, y position of the model feature f_m, respectively. If the geometric vector of the image feature f_i has a form of $f_i^G = (X_i^I, Y_i^I\ , \sigma_i^I\ , \phi_i^I)$, the suggested reference points for the image feature f_i is given by the transformation equation in (5.7).

$$\begin{bmatrix} X_i^{RkI} \\ Y_i^{RkI} \end{bmatrix} = \frac{\sigma_m^M}{\sigma_i^I} \cdot \begin{bmatrix} cos(\phi) & sin(\phi) \\ -sin(\phi) & cos(\phi) \end{bmatrix} \begin{bmatrix} X_m^{RkM} - X_m^M \\ Y_m^{RkM} - Y_m^M \end{bmatrix} + \begin{bmatrix} X_i^I \\ Y_i^I \end{bmatrix}, \qquad (5.7)$$

where $\phi = \phi_m^M - \phi_i^I$ is the orientation difference between the model feature f_m and the image feature f_i. Vector $\{X_i^{RkI}\ ;\ Y_i^{RkI}\}$ for $k\ \epsilon\ \{1,2\}$ is the reference pair x, y position for the image feature f_i.

Once all the matched features to the model are found and transferred to the new geometry using the transformation (5.7) , many potential pairs of reference points are obtained at different locations in the test image with various orientations and scales. These pairs are then clustered with respect to their similarities in location, scale and orientation and cluster centers besides their underlying members are participated in the Bayesian classifier in (5.1) to determine the real reference maps among all the potential reference points found in the test image as discussed in the previous section. For every real reference map found in the image, an in-plane rotation angle is assigned using the orientation angle of the cluster center for that reference map.

5.4 Experimental Result and Discussion

In this Section, the proposed framework is evaluated using different image datasets. Its performance is compared to the state-of-the-art methods for face detection and localization. For evaluating its robustness, we inserted occlusion disk over the face image before performing face detection. Then, we measured the accuracy of the estimated angle for in-plane rotated face images from various databases. Finally we presented the computation time comparison between the proposed method and other state-of-the-art methods.

5.4.1 Experimental Setup

We evaluate our proposed algorithm using publicly available datasets comprising of the standard color FERET dataset [75, 76], the Face Detection Dataset and Benchmark (FDDB) [77], the CMU rotated face images [35] and the CMU profile face image dataset [67]. FERET dataset contains images of 1,209 persons of various age, gender and ethnicity. Each person has an average of 10 images of arbitrary view and in different illumination conditions. The number of images in each viewpoint is not uniform. For instance, a person in the dataset may contain more than three frontal faces but only one full right profile. We resize all the images to 256x384 pixels. In the FDDB dataset, 2,845 images are taken from the faces in the wild dataset [68] and separated into 10 folds randomly. These 2,845 images captured under natural settings contain 5,171 faces (each image may contain more than one face) annotated manually as a benchmark. Faces in this dataset have a wide range of difficulties such as occlusion, difficult poses, low resolution, and out of focus problems. The CMU profile faces dataset contains images acquired under natural settings. The image size is not uniform and the resolution of the faces varies. Some face images have poor resolution (i.e. the size of the face is very small). Faces in this dataset have high background clutter and each image may contain more than one face from different viewpoints. The CMU rotated faces dataset contains 50 face images with in-plane rotation and each image may contain multiple faces, some with very low resolution. The very poor quality images are manually removed, resulting in 40 images and 65 faces. The last data source is the Google[1] image search engine used by the author to download some face images. For all the above datasets, we convert the color images into gray scale.

5.4.2 Face Detection and Localization from Multi-Views

[1] http://www.google.com/

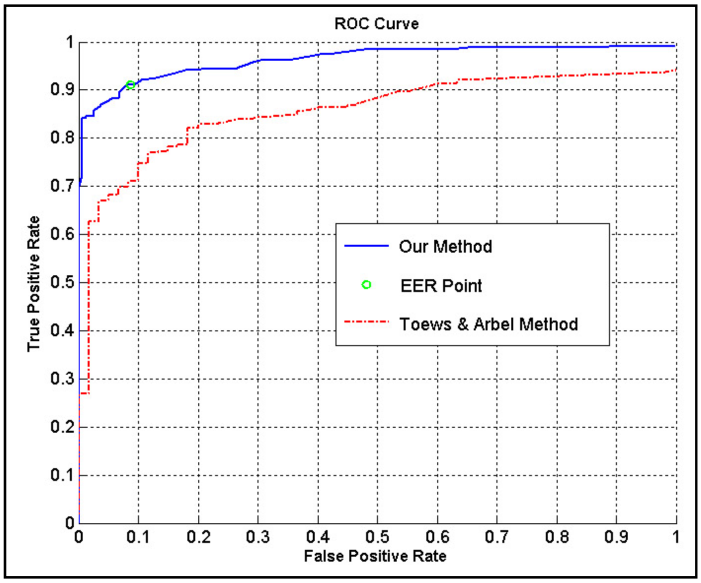


Figure 5.2. ROC curve is sketched based on the results of our method and Toews and Arbel method in two different test set conditions. First, the test set includes all 500 images of random view from left profile to oblique, frontal and right profile. Second, the full profile images (+/- 180 degrees) are discarded from the test set and only 367 images from left oblique to frontal and right oblique are considered

In this experiment, we compared our approach with the method by Toews and Arbel [40, 78]. Similar to [40, 78], 500 images of different people (one image per person with random viewpoint) are randomly selected from the FERET database as the training set. The significant difference between our method and [40, 78] is that their method requires manual selection of all the control points in all the 500 training images. In our case, we only manually selected two control points on only one image (the reference image). It is independent on the number of training images used. For testing, we selected 500 images of different persons not in the training set. Again, one image per person from random viewpoint is used. The Receiver Operating Characteristic (ROC) curve obtained is shown in Figure 5.2. The ROC curve is obtained by changing the threshold value of the probabilistic classifiers in equation (5.6) from zero to infinity. Those reference vectors, which fall within a circle centered at the reference point (base of nose, which is one of the two control points) with a radius of half the distance of the two control points and having scale and orientation within 20% of the reference vector are considered as correct detection. The results

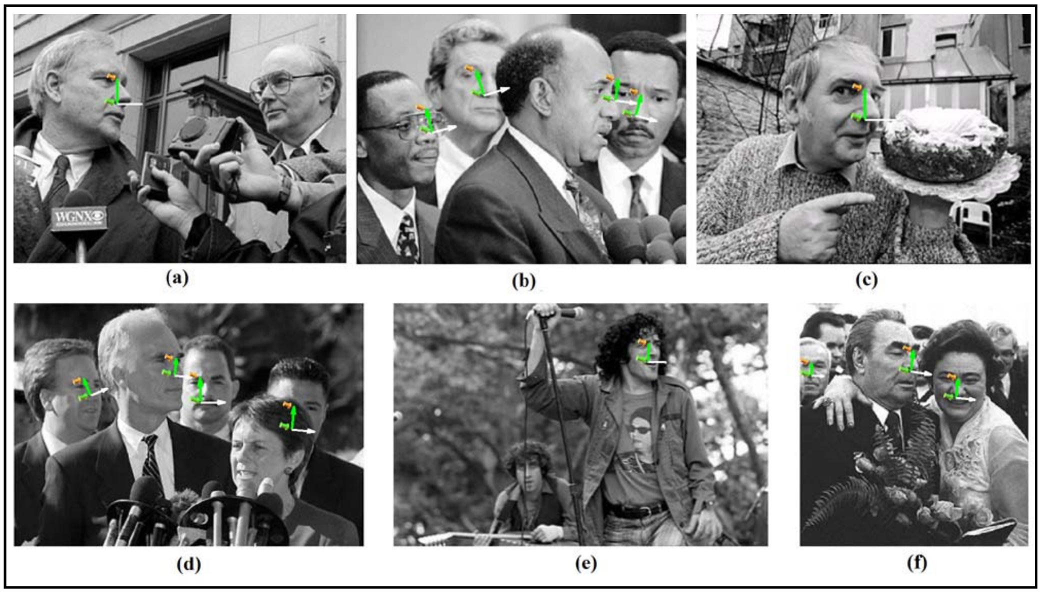


Figure 5.3. Sample results of our proposed method applied to the CMU database. The results show that this framework is invariant to viewpoint, occlusion and scale changes. Some faces in the images are not detected. The small faces are not detected due to poor resolution of these faces, causing SIFT feature extraction to fail. For those left profile faces not detected, the number of training images available in the dataset is not sufficient to cover these views.

obtained show that our method outperforms Toews and Arbel [40, 78] by an average of 12%. Based on this curve, we compute the threshold corresponding to the equal error rate (EER) point and use this threshold in equation (5.6) for all subsequent experiments.

We further evaluate our method trained using the FERET dataset, on the CMU face dataset [67]. Sample results are shown in Figure 5.3. Most of the undetected faces are due to low resolution (or small faces). For such faces, the size is close to or below the lower bound of the scale of the SIFT operators used. Therefore, the features extracted are not distinctive and this affected the accuracy of our face detector. Another reason for the failure is due to insufficient training images, causing failure to find the appropriate link or sufficient correspondence matches to the reference face. We found that in the FERET dataset used for training, there is insufficient number of left profile or left oblique faces, causing the model built to be incomplete. As a result, such faces cannot be detected as seen in Figure 5.3(a) and Figure 5.3(d). However, after adding randomly selected 50 images of left profile and left oblique faces from the FERET

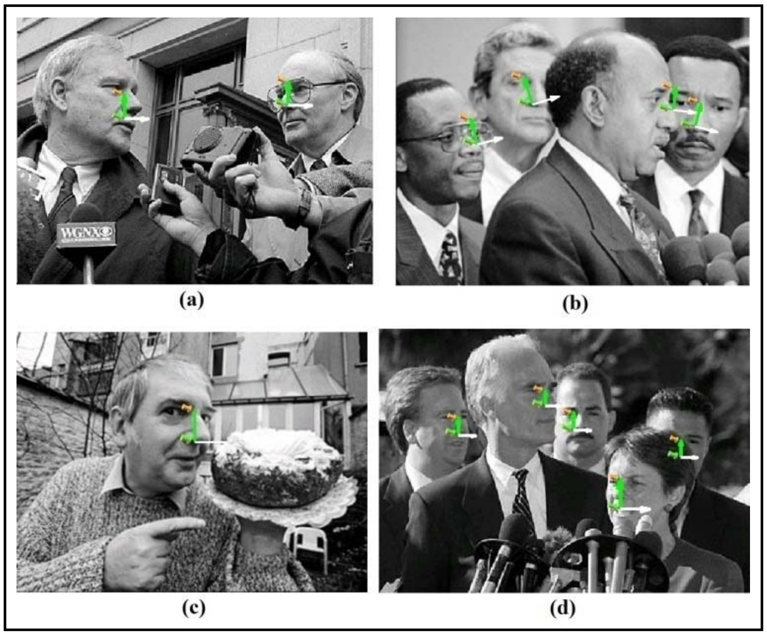


Figure 5.4. Sample results of our proposed method after adding 50 random images of left profile and left oblique faces in the FERET dataset to the training stage. It shows that almost all images can be successfully detected.

dataset to the training stage, such images can be successfully detected as seen in Figure 5.4.

The third dataset used to evaluate our method is FDDB [77]. We used the same data as Jain and Learned-Miller [79]. We compared the faces detected with the ground truth data and the degree of match is obtained by calculating the ratio of intersecting area using (5.8).

$$S(D,G)=\frac{Area(D)\cap Area(G)}{Area(D)\cup Area(G)}, \tag{5.8}$$

where D is the detected face and G is the ground truth face. Here, $Area(\,)$ implies the area of face which typically is defined by an ellipse or a square window. Those detected faces with match value, $S(D,G)$, greater than 0.5 are considered as a correctly detected face in the discrete score. In continues score, the direct value of $S(D,G)$ is used. For further detail on the evaluation protocol, please refer to [77]. Figure 5.5

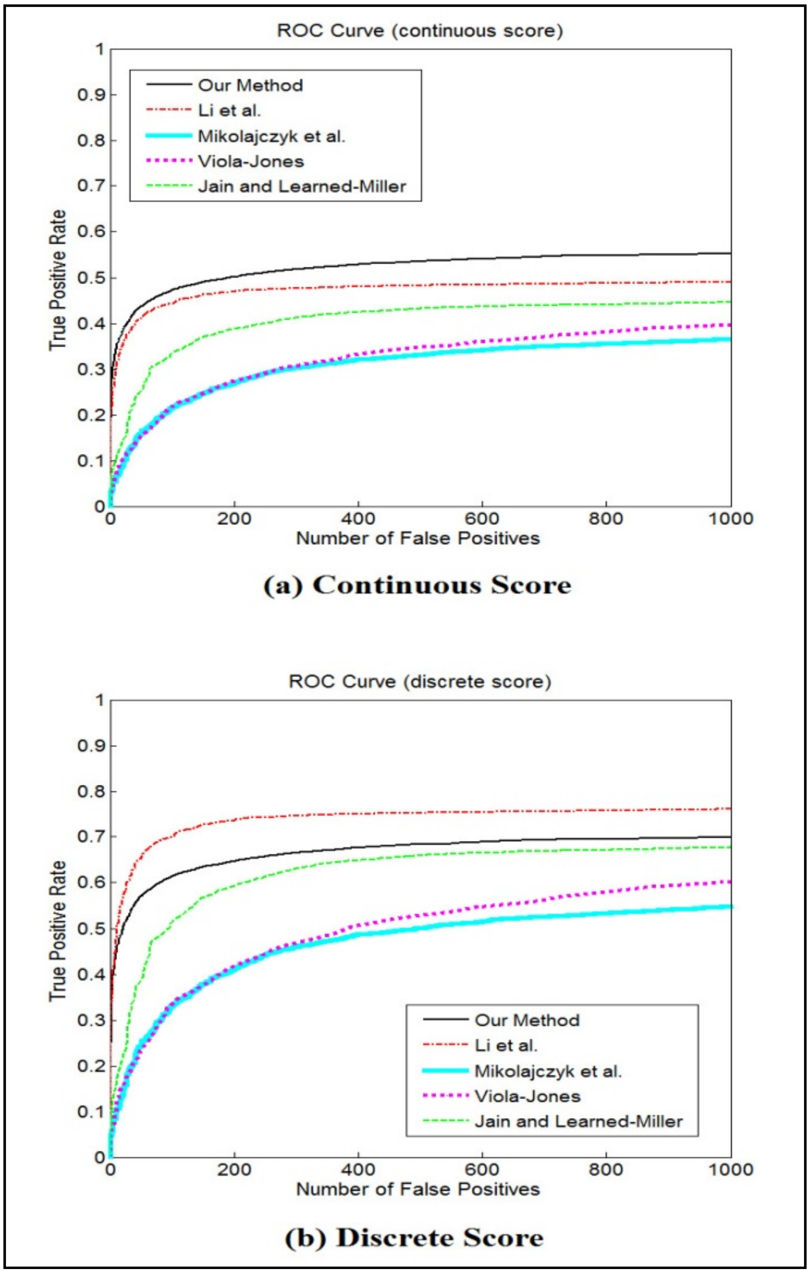


Figure 5.5. ROC curves obtained using the evaluation method proposed in [77] on the FDDB dataset, showing both (a) Continuous score and (b) Discrete score. The proposed method is compared against Li. et al. [80], Jain and Learned-Miller[79], Viola and Jones [81] and Mikolajczyk et al. [82].

shows that our method has results comparable to the other methods evaluated in the FDDB database.

Our approach performs better than others in the continuous score category because it is able to locate the faces more accurately that the others. Our approach automatically determines the reference points in the faces, while the other methods only return a window that included a part of the detected face. However, in discrete score, our method is not as good as Li et al. [80].

This is partly because we did not perform any post-processing to validate the face region, but merely draw an ellipse with parameters similar to the reference image and centered at the midpoint of the two reference points to meet the requirement of the evaluation. Thus, for non-frontal faces, the estimated ellipse will cover some background regions. In the extreme case of a profile face, the ellipse drawn will only cover about half the face. Thus achieving greater than 50% overlap is not possible, lowering our performance. Nevertheless, the performance is still comparable because typically other methods could not find faces of non-frontal views or occluded faces. It should be emphasized that our method is trained with only 500 images. Even then, we only manually label one image with two points whereas the other methods require much more positive and negative samples to train, thus requiring large number of manual labeling.

5.4.3 Occlusion Analysis

In order to evaluate the robustness of our proposed approach against occlusion, we used a disk to occlude the face and then grow it gradually. A random face in frontal view different from the training set (Figure 5.6(a)) is selected from the FERET dataset. The disk is then placed at four locations: top, bottom and left or right mid-points and center of the image to minimize any ambiguity due to features chosen and exact position of the face. Figure 5.6(b) shows the result without any occlusion. The maximum level of occlusion which our method is able to tolerate for the occlusion growing from bottom, top, center and left is shown in Figure 5.6(c), (d), (e), and (f) respectively. Then the ratio of the area of the disk relative to the total image size is computed as percentage. Using this protocol, we conduct the experiment on 500 randomly selected FERET dataset images. We plot the percentage for all the four locations just before the detection fails (see Figure 5.7). This shows that the proposed approach can tolerate about 40% to 50% occlusion, as long as sufficient distinctive

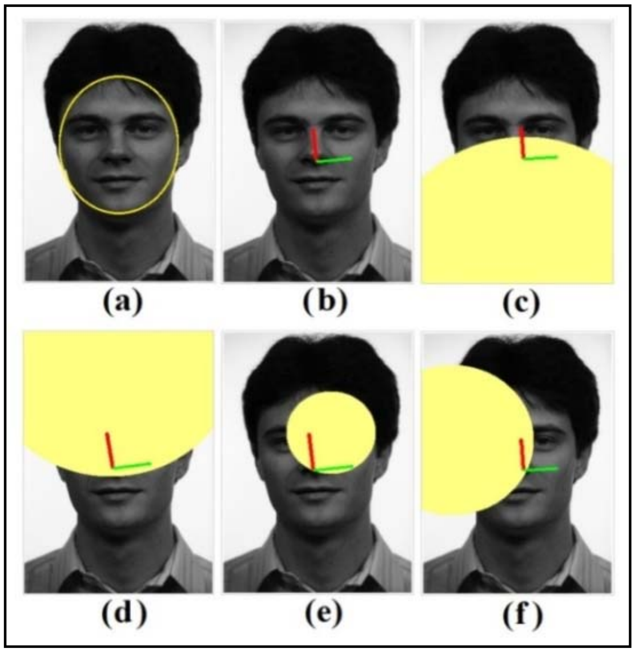


Figure 5.6. Shows the capability of the proposed method to detect faces under occlusion. (a) Original face image.. (b) Detection and localization result without any occlusion. (c) – (f) The largest occlusion before the detection fails when the size of the occluding disk is increased from bottom, top, center and left/right (symmetrical for frontal face) respectively.

features (typically eyes and mouth regions that have high similarity for most faces) can be detected to establish the correspondence.

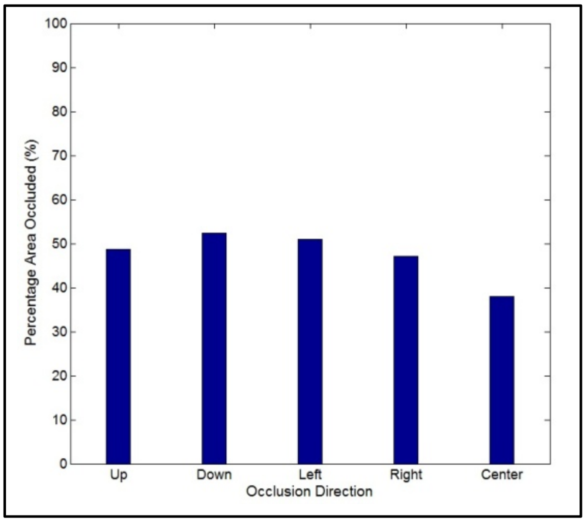


Figure 5.7. Percentage of occlusion in the 500 images taken from FERET dataset just before the face detection fails.

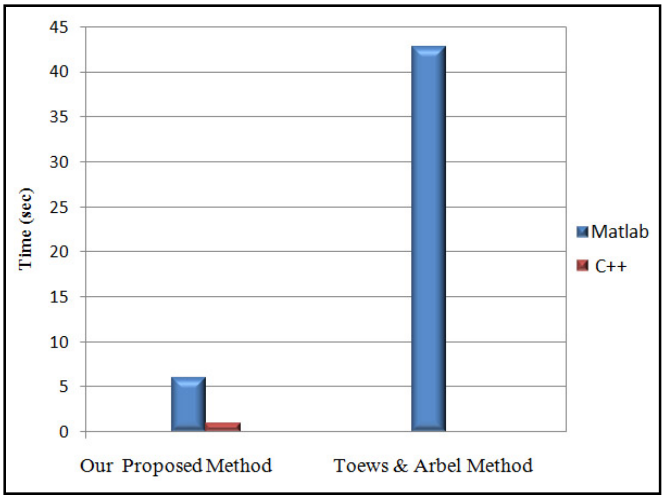


Figure 5.8. Compares the processing time between our proposed method and the method proposed by Toews and Arbel [40]. The processing time indicated is based on the average time taken for each method to process a test image of resolution 320×240 pixels in a total of 500 images.

5.4.3 Computation Time

We compared the computation time between our method and Toews and Arbel's method [40, 78]. It is shown (see Figure 5.8) that our method is up to 7 times faster than their method on the same machine. This is because our method is able to select more distinctive features, resulting in a more compact and efficient model construction. We implemented the face detection and localization portion using C++. The average time for detecting and localizing a face in an image of 320×240 pixels is about 540ms using an ordinary PC (2.8GHz C2D PC with 3 GB RAM and Linux OS using a single core). As a comparison (shown in Figure 5.9), the execution time for the window based method of Huang et al. [3] and Li and Zhenqiu [39] is 250~330ms (using Pentium 4-3.0GHz) and 200ms (using Pentium 3-700MHz) respectively. However, these methods yield only coarse detection. It will still require face alignment to localize the face. On the contrary, our approach performs both detection and localization, providing the position, scale, and amount of in-plane rotation of a detected face simultaneously. Thus, the computational time of our method is reasonable and not significantly slower.

Majority of the time is spent on SIFT feature extraction (480ms) [83]. The time needed for comparing the model and the extracted features is not significant. Based on our

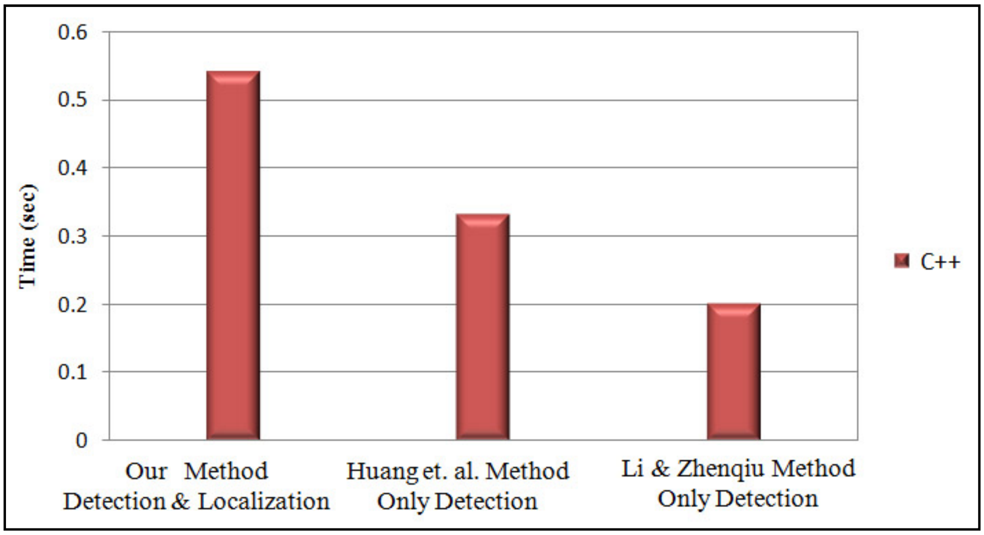


Figure 5.9. Compares the processing time between our method and the methods proposed by Huang et al. [3] and Li and Zhenqiu [39].

experiment, with 500 training images, we have about 8,000 features in the constellation. After mean shift clustering, the number is reduced to about 800 features. The comparison time between the 800 features in the constellation and the extracted features is only about 60ms. In addition, our method has a huge time advantage in the training or model construction stage. The window-based methods and Toews and Arbel [40, 78] method require manually labeling all the training images which is a very time consuming task. For example, Huang et al. [3] reported labeling 30,000 frontal, 25,000 half-profile and 20,000 full-profile faces manually. However, our approach requires only manually labeling two control points in only one image, the reference image. There is no further manual intervention needed. Thus, the effort is negligible compared to the other state-of-the art approaches proposed thus far.

5.4.4 In-Plane Angle Estimation Evaluation

We used the same face model obtained in the previous sections (trained with the 500 images of different people randomly selected from the FERET database). Only the detection stage has been changed and some amendment has been added to acquire an in-plane rotation invariant face detector. In order to evaluate the rotation invariant

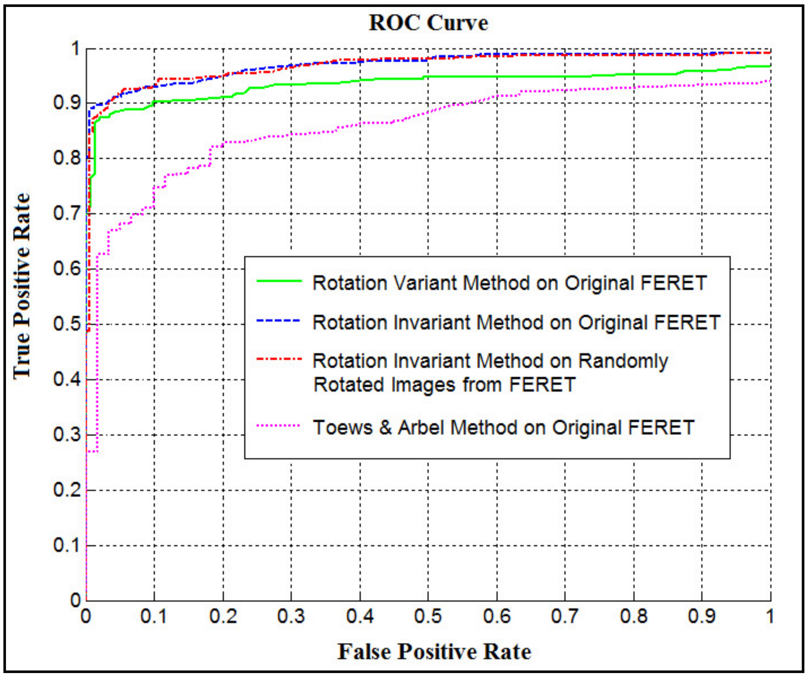


Figure 5.10. Compares the performance of the proposed method in three different scenarios: Rotation Variant method on the original images, Rotation Invariant method on the original images and Rotation Invariant method on the randomly rotated images from FERET dataset. The ROC curve for the method proposed by Toews and Arbel [40] is also sketched for better comparison.

method and its performance in estimating the in-plane rotation angle, FERET dataset [75, 76] and the rotated faces from CMU dataset [35] were used.

We plotted the Receiver Operating Characteristic (ROC) curves for the proposed method with and without amendment added to detect the in-plane rotated faces, which we call them Rotation Invariant method and Rotation Variant method, respectively. The results are shown in Figure 5.10. For evaluation, we used 500 randomly selected images from FERET database. From this database, we also created another database where the image was rotated around the center of the image with a random rotation angle. We considered three different scenarios: Rotation Variant method applied to the original images from FERET dataset, Rotation Invariant method applied to the original images from FERET dataset and Rotation Invariant method applied to the randomly rotated images from FERET dataset. It is seen that Rotation Invariant method performs a bit better than the Rotation Variant method on the same original images since some images in the dataset have small rotation and thus the Rotation Invariant method would

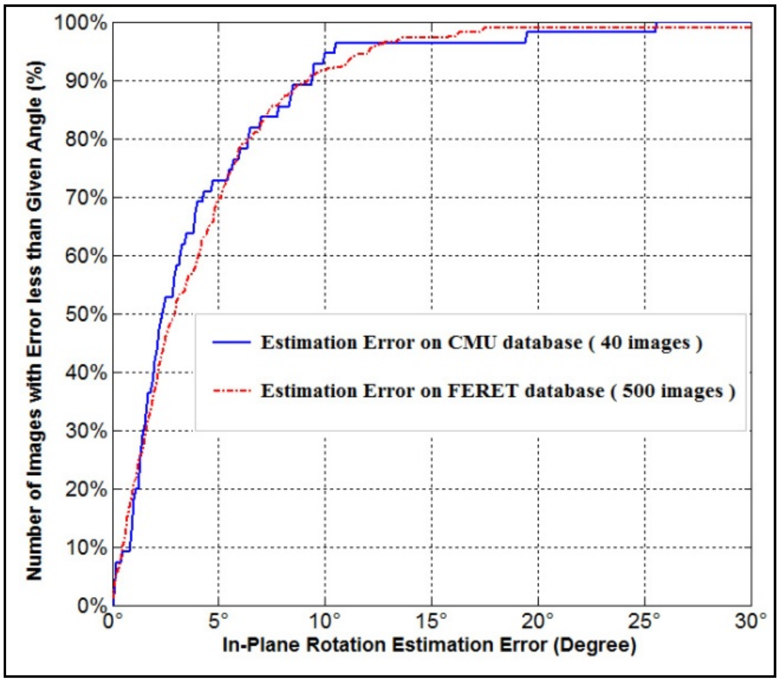


Figure 5.11. Accuracy of the estimated in-plane rotation angle on both the CMU and FERET databases.

be able to detect them more accurately. The performance of Rotation Invariant method in the original and randomly rotated images is the same, showing the proposed method is able to handle the rotated faces without any significant loss. For further comparison we plot the ROC curve obtained from the method proposed by Toews and Arbel [40] on the same test data. We should mention that their method is only able to detect up-right face images in from multi-poses. They cannot handle non up-right face images.

To measure the accuracy of the estimated angle by the proposed method, we compared the estimated angle for each face image with the ground truth data. The result is plotted in Figure 5.11. The horizontal axis represents the in-plane angle estimation error and the vertical axis represents the number of face images which have angle estimation error less than the value indicated in the horizontal axis at each point. For instance, point $(70\%,5^o)$ tells that 70% of face images have angle estimation error less than 5°. Overall, about 93% of all the angles estimated have error within 10^o, which shows the accuracy of the proposed method.

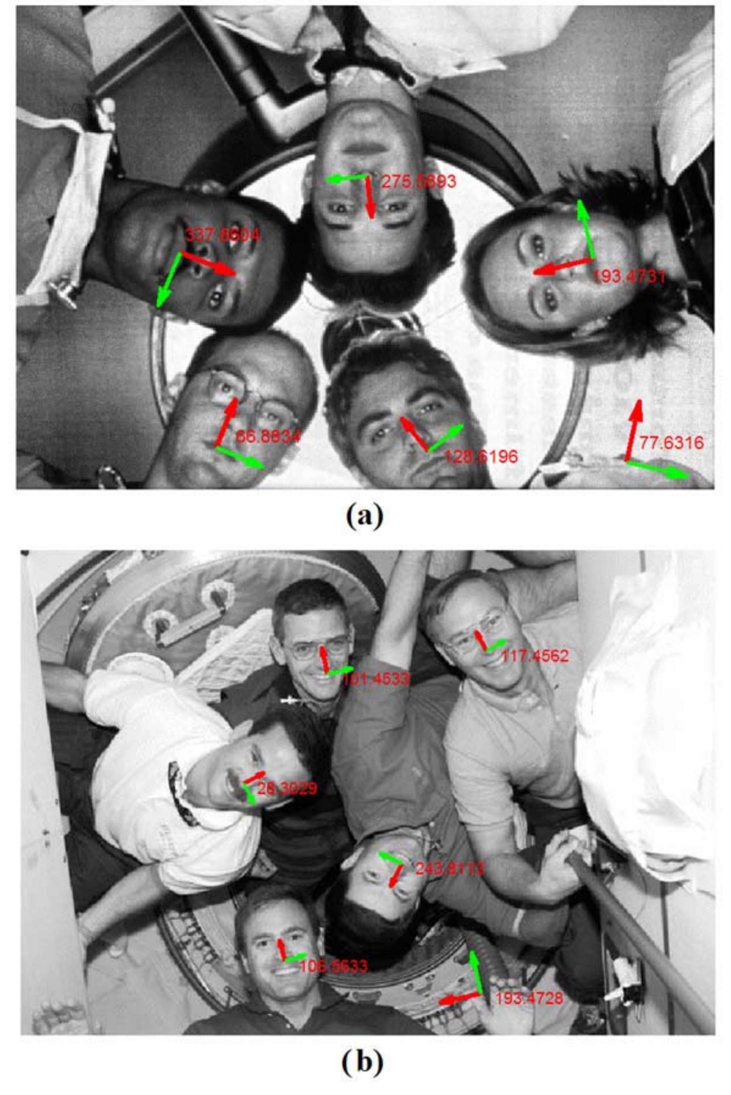


Figure 5.12. Samples of detected faces and the estimated in-plane rotation angles on some images of multiple faces in CMU database.

Some results of the proposed method on CMU face dataset are shown in Figure 5.12 and Figure 5.13. The detected reference vectors for each face are indicated by the red and green arrows and the in-plane rotation angle estimated for the face is shown near the reference vectors in the image. The orientation angle is measured relative to the horizontal line. The results show that the proposed method is able to detect most rotated faces and once the face image is detected, the estimated angle is quite accurate.

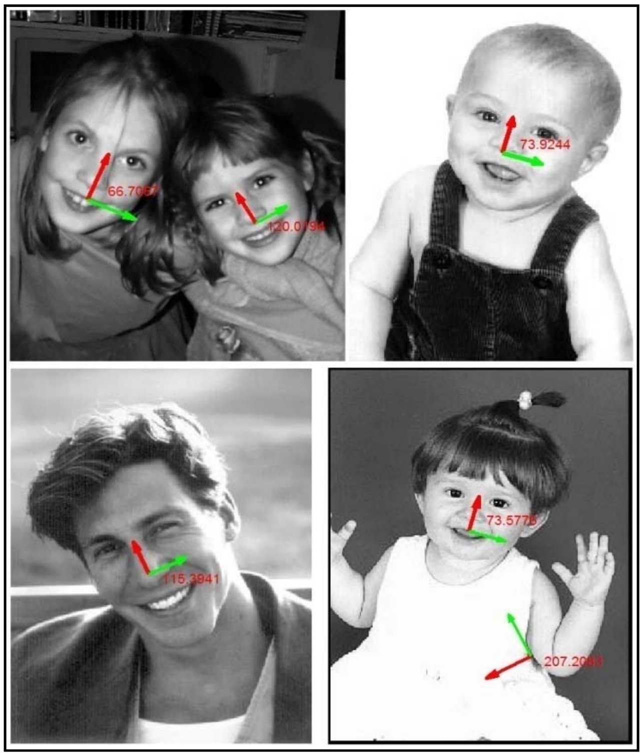


Figure 5.13. Samples of detected faces and the estimated in-plane rotation angles on some images of individuals and twin faces in CMU database.

Finally we have compared our proposed method with Viola & Jones [42] method on the rotated face images obtained from the CMU dataset (see Figure 5.14). The result is tabulated in Table 5.1. It shows that although the number of detected faces is slightly lower than Viola & Jones by about 3%, our proposed method has 50% lower number of falsely detected faces. The accuracy of the estimated angle is also about 3 times better. The processing time per image is also about 3 times faster. Although the Viola & Jones face detection computation time for an image size of 240×320 pixels is about 120 ms, they have to repeat the detection process at least 12 times to find rotated faces and calculate the face in-plane rotation angle as depicted in Figure 5.1. In addition, our method has also another advantage where the training is very simple. We only used 500 face images and require manually labeling only one image. In contrast, Viola & Jones's method used 8356 face images and over 100 million background and non-face

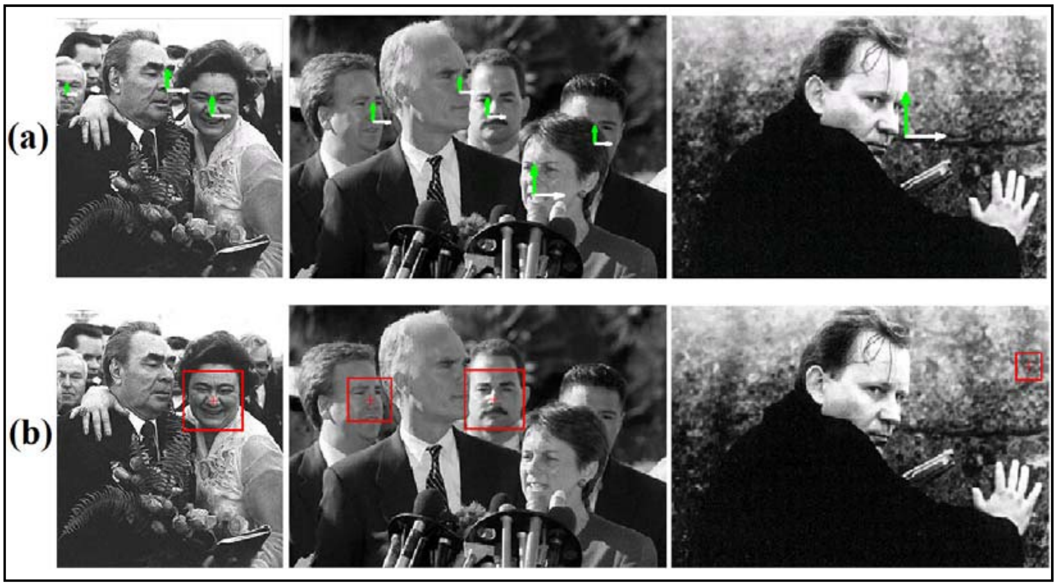


Figure 5.14. Compares the results of detecting face images between the proposed method (a) and Viola & Jones [42] method (b) on some samples from CMU dataset. It is seen that the proposed method is able to handle multi-view face images and faces which are occluded partially

Table 5.1. Accuracy and processing time comparison between our proposed method and Viola & Jones's method on the CMU dataset

Method	**Number of Images**	**Number of Faces**	**Detection Rate**	**Number of Falsely Detected Faces**	**In-Plane Rotation Estimation Error**	**Detection Time for a 240×320 image**
Our Method	40	65	90.9 %	52	±10°	550 ms
Viola & Jones [42]	40	65	93.8 %	96	±30°	12×120 ms

samples to train their classifiers. If more images are used to train our model, we believe the accuracy of the face detection result will be higher.

5.5 Concluding Remarks

In this chapter, we have presented an approach to simultaneously detect and localize multiple faces from arbitrary views and in different scales. It is also able to detect rotated face images and estimate the in-plane rotation angle. This is achieved by establishing a reference image on a reference map using two points selected on the reference image and then registering other faces to this reference image either directly or indirectly through the already registered faces. This will produce a constellation of linked faces as the basis for face detection. We showed that by providing more diversity, the proposed method is able to handle arbitrary views, varying illumination conditions and complex backgrounds. The probabilistic framework to evaluate whether the detected features is likely a face allows the approach to detect multiple faces and occluded faces. Unlike other approaches, it does not require many manually labeled faces apart from two points in the reference face image for training. Despite its simplicity, experimental results show that the performance is better than the state-of-the-art approaches for multi-view face detection and the processing time required is also reasonable. Note that the proposed method does not have any prior assumption about the training images. It starts with a seed image of face and enriches the classifier with some training data that it can link to and gradually enrich the links through more varying samples. Thus, the capability of the classifier built will only grow with more images that are diverse, so long as the intra-class variation is not too large such that sufficient correspondence points cannot be found.

Chapter 6

Automatic Face Model Learning from Ordinary Images in the Web

6.1 Introduction

Emerging new media technologies such as interactive television, electronic photo albums and social network photo sharing services have a thing in common, that is they have many images or videos with faces. This opens up new demand to query or manage the images and video based on face recognition. Some of the examples include face recognition module found in Picassa, Facebook and iPhoto that helps you tag the faces of the people whom you have tagged before as shown in Figure 6.1. In interactive video, tagged faces allow the relevant segments of the video that contain the person you are looking for to be retrieved or the statistics of a player to be shown when the player is in view. Consequently, face recognition and identification have attracted many computer vision and pattern recognition research works, some of which are [7-10, 46-48, 84]. In an interesting research Bicego et al.[85] have investigated a method to determine the distinctive patches from face images for face identification. However it is sensitive to view changes. There are also famous contribution conducted around face identification and authentication using SIFT features that showed the strength of SIFT feature for face identification task [86-88]. However, almost all these works assume that the input images are a priori detected, aligned and cropped. There are also works that use special imaging equipment such as the near infra-red camera [51]. Very few considered the general identification problem using the ordinary images from the World Wide Web and most of the state-of-the-art approaches only work well

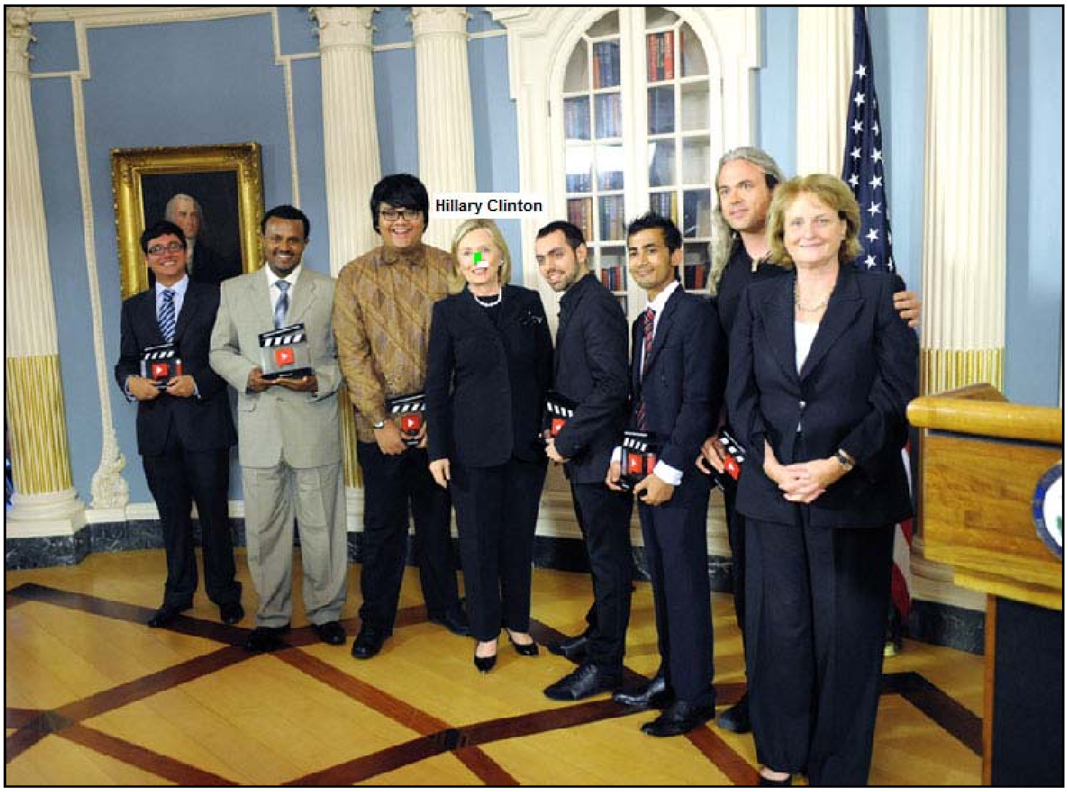


Figure 6.1. Demonstrates the results of finding and tagging people in their digital photos Albums. As it is seen, the automatically learned model is able to detect and localize the person of interest with high quality. (This image with photo number 4978695826 was adopted from the US secretary of state website labeled as "Public domain" under the free copyright license.)

for frontal face images. They cannot handle non-frontal views such as the oblique or profile face images. In addition, these methods require substantial effort to prepare the images, such as under controlled setting or collecting a large number of face images and non-face images for training during the enrolment stage. As such, many standard datasets [66, 89] have incurred high cost in preparing them, yet could only cover specific or limited cases. Without sufficient training images covering all possible scenarios, there will be limit to successful application of general face recognition.

In this chapter, a general face/character identification and face tagging method is introduced which is able to detect, localize and identify faces of arbitrary view in a given image even though the faces only occupy a small part of the overall image. The proposed framework is completely automatic, from the training or enrollment stage to the identification process. No manual intervention is considered apart from identifying the identity of the face image. We also proposed a simple training method that uses only images obtained from the web and does not require additional training dataset or special equipment. In the beginning, our framework requires only the name of the queried person. The name is used to search for suitable images from the web and to validate some of the found images.

6.2 Automatic Approach for Learning Face Model from Multi-Views

Acquiring or developing proper training dataset is crucial for the success of face detection and recognition. The purpose of training is to develop face models for use by the face detector and face recognition module. For cascaded system, each module has its own face model. Nevertheless, the model has to encompass as many variations as possible for the module to work in general settings. The variations could be due to age, gender, ethnicity, facial hair, pose, emotion, lighting condition, occlusion and background settings. Therefore, preparing a proper dataset is expensive but it is needed by the existing system to achieve good performance in general settings. As such, researchers have developed several datasets such as PIE, FERET and FDDB for training and evaluation [66, 77, 90]. Any further improvement will require further addition to the training datasets. In this chapter, we aim to remove the dependency of our algorithm to such special dataset. Instead, we propose to leverage the wide availability of face images found from the web. This implies that we have to work with images that we cannot control, with possibility of wrong tagging and under many possible variations and conditions. In the next part, we describe the process of training a suitable model using such images.

6.2.1 Develop Face Model from the Images Found from the Web

We assume that there is sufficient number of different tagged face images of a person of interest. The larger the number of different faces, the higher the variations of pose, emotion etc that the system can handle. The framework starts with an entry of the name of a candidate person to be queried (please see Figure 6.2(a)), for example “Nicolas Sarkozy”. A web search is then initiated to find the images of that candidate. As the search engine finds the images based on the tag or metadata prepared by the crowd, the metadata is not always correct. Several studies [91, 92] have shown that the correlation between the text query and the images returned by a web search engine achieves only a precision of minimum 2% to maximum 70%. In addition, there are also duplicate images being retrieved by the search engine. To solve these issues, we filter and cluster the images returned by the search engine.

We assume that among the retrieved images, the majority are related to the candidate. This assumption is reasonable as the top 100 images usually have average precision of 50%. If this assumption is violated, the constructed face model will be poor, resulting in higher error of face recognition. In order to handle variations in the face, sufficiently large number of images is required from the retrieved result. In our work, we typically use 400 images. These images will form the training images. Subsequently, we perform matching on the training images by extracting the scale invariant features using the method introduced in [14, 63]. Then the real correspondence points are obtained using the algorithm proposed in chapter 3. The images with high amount of corresponding features are possibly from the same person and are thus clustered together. However, images with very large number of correspondence points are rejected because it is very likely that these are the exact duplicates. This will reduce processing time and avoid scattering of the face model among the duplicates. The result of matching stage is a number of clusters of coherent or similar images with each image in the cluster connected to the other images through their correspondence points as depicted in Figure 6.2(c). Applying the probabilistic approach proposed in chapter 4 discards the weak connections between the images in each cluster. The number of clusters depends on the number of unrelated images retrieved by the search engine. The cluster with the largest number of coherent images is more likely to contain the samples of the candidate. Other clusters could be unrelated or from the poor quality images of the candidate which could not connect to the correct cluster.

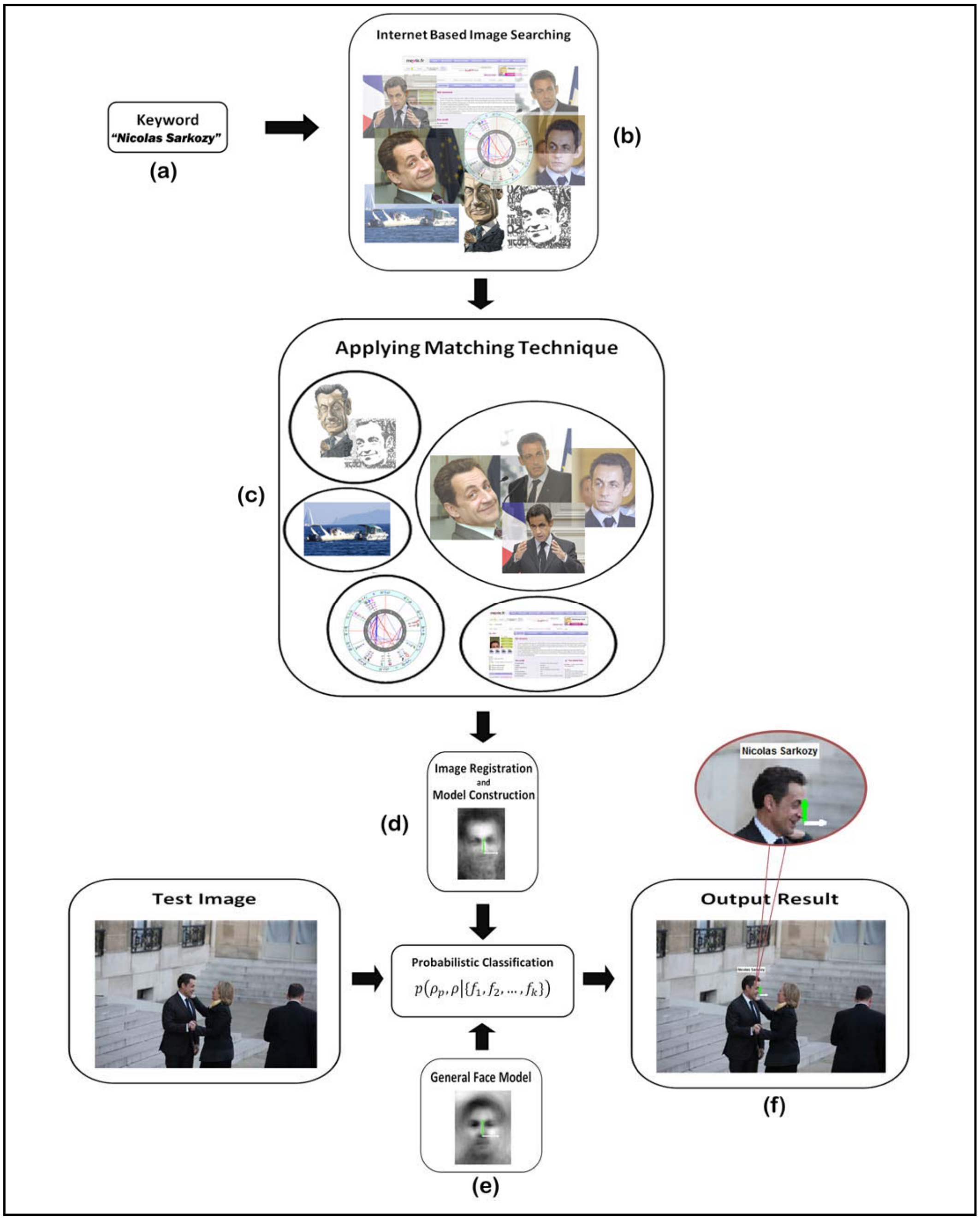

Keyword
"Nicolas Sarkozy"
(a)
Internet Based Image Searching
(b)
Applying Matching Technique
(c)
Image Registration and Model Construction
(d)
Test Image
Probabilistic Classification
$p(\rho_p, \rho|\{f_1, f_2, \ldots, f_k\})$
General Face Model
(e)
Output Result
Nicolas Sarkozy
(f)

Figure 6.2. General framework for automatic model learning and face tagging system. The system starts with a keyword (a) and searches the Web for the images of the queried person (b). Matching process clusters the images retrieved by the search engine into different segments (c). Members of each segment are similar according to certain similarity metrics. The group with the maximum number of members is considered as that of the queried person and a *reference map* and feature model (d) from projection of the 3D into the 2D space is constructed. This model together with the general face model is used as a input to the probabilistic classifier to detect, localize and tag the queried person in the given image (f). As shown in the output result (f), the system is able to handle variations due to rotation, scale and pose. (Test image with Photo number 5537360864 was adopted from the US secretary of state website labeled as "Public domain" under the free copyright license.)

6.2.2 Locate *Reference Point*s and Register Similar Images to a *Reference Map*

Given the face image cluster with the correspondence points within the cluster, we need to associate these points to the face in order to properly detect and localize these faces. We used the approach in chapter 5 which allows for simultaneous face detection and localization and is able to support scale, rotation and pose variation. To begin, the method requires a face image (called reference image) with two control points indicated in the image. The control points, called *reference point*s, are selected such that they do not change when the face is subjected to in-plane rotation or pose variation. As such, all the correspondences the image have with the other images can be mapped as well as the correspondences among the immediately related images but not with the reference image. The links is further grown until all the images in the cluster are connected to form a constellation. However, in chapter 5, the reference image and the *reference point*s have to be manually selected. This limits the approach from being fully automated. Hence, in this section, we further generalize the approach in chapter 5 to allow automatic selection of the reference image and estimate the associated *reference point*s.

Since the face images use their links to the reference image for registration, we select the image with the highest number of links as the reference image. We then estimate the two *reference point*s in two stages; (i) estimate two *reference point*s from all the feature points with valid links and (ii) refine accuracy of the *reference point*s after all the images have been registered to the reference image.

To estimate the two *reference point*s, we consider only those image features that have made connection between the reference image and the other images. We determine the base point by averaging the positions of all the features in the reference image as shown in Figure 6.3(red diamond point). This point is used as the base of the *reference vector*. The scale of the *reference vector* is obtained from the variance of the position of all the features around the base point limited to factor of all the feature points nearest to the base point (in our work, we used $\sqrt{2}/2$). Since majority of the face images are close to up-right, we initially estimate the orientation of the *reference*

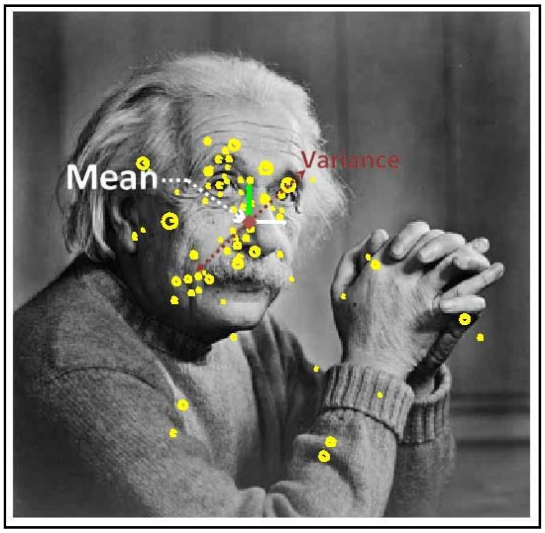


Figure 6.3. Estimating the initial *reference vector*. The average location of the distinctive features is used as the base point for this vector (indicated by the red diamond point) while the scattering (variance) of the distinctive features around the base point is considered as the scale of this vector (indicated by the red dotted line). The orientation assigned to the vector is perpendicular to the horizontal line (indicated by the green solid line)

vector perpendicular to the horizontal line (shown by the green solid line in Figure 6.3).From the *reference vector*, the two *reference point*s can be determined. All the other images are then registered to the *reference map* as described in detail in chapter 4.

Once the initial registration process is completed and initial face model has been shaped based on the features of the registered faces, the *reference point*s are refined as follows: Let the obtained model contains N features from face area of different poses. We would like to determine two *reference point*s such that they represent faces in most of the views. In other word, these two points should appear in the most number of images from different views. Since the features are derived from the face area, we determine the location of the point ($f_{basepoint}^{x,y}$) with the highest probability to represent the faces by maximizing the probability function in (6.1).

$$f_{base\ point}^{x,y} = arg \max_{n=1:N_f} P\left(\left|f_n^{x,y}\right| < Z\right), \qquad (6.1)$$

where$Z$ represents the face region and $f_n^{x,y}$is the location of the feature n in the model. The term $P\left(\left|f_n^{x,y}\right| < Z\right)$ determines the probability that the features are located within the face region. We can change the probability to the form of density function by integrating the features' position inside the face region as(6.2).

$$f_{base\ point}^{x,y} = arg \max_{n=1:N} \int_Z p\left(f_n^{x,y} \middle| \mu, \sigma\right) dz, \tag{6.2}$$

where μ, σ are the mean and variance of the density function $p(\,)$ respectively. Assuming that the features in the model follow a Gaussian distribution function with equal distribution (σ_0) in both axes (xy) and mean values μ_x and μ_y in x and y directions, respectively give in (6.3).

$$\begin{aligned} p\left(f_n^{x,y} \middle| \mu, \sigma\right) &= \frac{1}{2\pi\sigma_0^2} . e^{\frac{-(x-\mu_x)^2-\left(y-\mu_y\right)^2}{2\sigma_0^2}} \\ &= p(f_n^x | \mu_x, \sigma_0) \times p\left(f_n^y \middle| \mu_y, \sigma_0\right) \end{aligned} \tag{6.3}$$

where f_n^x and f_n^y are the x position and the y position of the feature n in the model, respectively. Thus, the equation in (6.2) could be simplified using (6.3)as in (6.4).

$$\begin{aligned} & f_{base\ point}^{x,y} \\ &= arg \max_{n=1:N_f} \left(\int_{-y}^{y} \int_{-x}^{x} p(f_n^x | \mu_x, \sigma_0) p\left(f_n^y \middle| \mu_y, \sigma_0\right) dxdy \right) \\ &= arg \max_{n=1:N_f} (\int_{-y}^{y} p\left(f_n^y \middle| \mu_y, \sigma_0\right) dy \times \int_{-x}^{x} p(f_n^x | \mu_x, \sigma_0)\, dx), \end{aligned} \tag{6.4}$$

Instead of computing (6.4)we find the maximum point by solving the equations in(6.5).

$$\begin{cases} \dfrac{\partial}{\partial \mu_x}\left(\int_{-x}^{x} p(f_n^x|\mu_x, \sigma_0)\, dx\right) = 0 \\ \dfrac{\partial}{\partial \mu_y}\left(\int_{-y}^{y} p\left(f_n^y|\mu_y, \sigma_0\right) dy\right) = 0 \end{cases} \tag{6.5}$$

Considering an initial value $\sigma_0 = 1$ at the beginning, the position of the base point (μ_x, μ_y) could be obtained from solving the equation pair in (6.5). The variance of the normalized features around this point (σ_0) is recalculated by considering the fact that their positions' are scattered according to a Gaussian distribution as in (6.3). The scale of the refined *reference vector* is then taken from σ_0 but its orientation remains unchanged. This is iterated to refine the location of the *reference point*s (i.e. base point location and scale) and the iteration stops when the scale of the refined *reference vector* converges to the final approximation such that the new updated value for it does not vary more that 1% from its updated value in the previous iteration. In our experiments, we noticed that usually about 10 iterations suffice.

Once the *reference point*s have been obtained, the face model is constructed from the normalized features to the *reference point*s and redundant features are pruned from the face model as explained in chapter 4.

6.3 Face Identification from Multi-views

The previous section provides the face model of the queried person. This section will describe a probabilistic approach to detect and localize the candidate person in a queried image. The face could be rotated or in different scale, pose and expression.

6.3.1 Probabilistic Approach Face Recognition

We define a probability function to estimate the occurrence of the candidate's face in the query image. Let $\{f_1, f_2, \dots, f_k\}$ be some features from a part of an image. The posterior probability function (6.6)gives the probability that these given features $\{f_1, f_2, \dots, f_k\}$ represent a face area (ρ) and originated from the candidate's face(ρ_p). Many similarities can be detected between the face model and the features from different parts of the query image. However, the occurrence of a person's face is affirmed when the evaluated probability function in (6.6) yields a high value, indicating that many similar features to the face model are found.

$$p\left(\rho_p, \rho \middle| \{f_1, f_2, \dots, f_k\}\right), \tag{6.6}$$

In the remaining part of this section, we focus on the evaluation of the probability function in (6.6) based on the training stage. Unlike other region-based and feature-based methods that used only some parts of the face for identification, we proposed that the extracted features from any parts of the face must participate in estimating the identity of a person. This implies that the features from different regions must be analyzed holistically and contributions from all features which shape the face of the queried person have to be considered in affirming or negating the identity of a person. All human faces share some common visual characteristics that distinguish them from other objects, but the detail of an individual's face differs from the others to allow an individual to be recognized. Thus we consider the candidate's face (ρ_p) as a member of a larger group of faces (ρ). Using the Bayesian and chain rule, the probability function in (6.6) can be re-written as in (6.7).

$$\begin{aligned} &p\left(\rho_p, \rho \middle| \{f_1, f_2, \dots, f_k\}\right) \\ &\qquad = p(\rho|\{f_1, f_2, \dots, f_k\}) \times p\left(\rho_p \middle| \{f_1, f_2, \dots, f_k\}, \rho\right), \end{aligned} \tag{6.7}$$

This function states that to identify a person, two probability functions must be considered. The first is the face detection function where given the image features, what is the probability that a face (ρ) occur given a location in the image. The second is the face recognition function where given that a face (ρ) has been found, or occurred in a region of the image, what is the probability that the features $(\{f_1, f_2, \dots, f_k\})$ belongs to a desired candidate face (ρ_p).

To evaluate the posterior form of the function $p(\rho|\{f_1, f_2, \dots, f_k\})$, we used the Bayesian rule to equivalently calculate the prior form $p(\rho)$ and the likelihood $p(\{f_1, f_2, \dots, f_k\}|\rho)$ as given in(6.8).

$$p(\rho|\{f_1, f_2, \dots, f_k\}) = \frac{p(\rho).\, p(\{f_1, f_2, \dots, f_k\}|\rho)}{p(\{f_1, f_2, \dots, f_k\})}, \tag{6.8}$$

Assuming that the given features $(\{f_1, f_2, \dots, f_k\})$ are independent, the equation in (6.8) can be simplified to the form in(6.9).

$$p(\rho|\{f_1, f_2, \dots, f_k\}) = p(\rho) \times \prod_{i=1}^{k} \frac{p(f_i|\rho)}{p(f_i)}, \tag{6.9}$$

where $p(\rho)$ and $p(f_i)$ are the probability of occurrence of a face and of a feature and can be regarded as two constant prior probabilities from the general face and features respectively. Thus they could simply be considered in the threshold term when the probability of the existence of a face area is being evaluated. As such, we only need to evaluate the likelihood term $p(f_i|\rho)$ as the main term. This term gives the probability that a feature f_i comes from a part of a face $(f_i = 1)$ or from the background or noise area $(f_i = 0)$. We use the general face model from chapter 5 and compare it to the face model obtained for the candidate as explained in the previous section to evaluate this term. Each feature from the candidate's face model, which is similar to the general face model, is considered as shared feature (f_i^{Sh}) while the remaining features from the candidate face model, which have no similarities to the general face model, are considered as dedicated features. The similarity is obtained by computing the Euclidean distance between the descriptors of the candidate face model's features and the general face model's features and comparing the distance to the appearance threshold T_i^{∂}. This threshold is evaluated for every feature in the general face model as described in chapter 5.

In the test stage, SIFT features [14, 63] from the query image are extracted and the Euclidean distance between them and the features in the model of the candidate is computed and compared against the appearance threshold T_i^{∂}. Each match represents a *reference map* in the test image. If N features from the query image match the model, N *reference map*s appear in different locations, orientations and scales in the query image as shown in Figure 6.4(a). In order to check whether the *reference map*s come from a true or valid face, they are clustered using a mean-shift clustering approach [43]. Mean-shift is used because initially the number of clusters is not known. Figure 6.4(b) shows an example of the cluster centers obtained from clustering the *reference map*s found in Figure 6.4(a).

After finding the cluster centers, we compute the probability whether each cluster center represents a valid face area by computing the probability function in (6.9) from the associated features in the cluster. Let M features represent the dedicated features and $(K-M)$ features represent the shared features, the equation in (6.9) would be re-arranged as in (6.10). We select only those cluster centers with probability value greater than a predetermined threshold.

$$\prod_{i=1}^{K-M} p(f_i|\rho) > \frac{\prod_{i=1}^{K-M} p(f_i)}{p(\rho)} \times THR = T^{SH}, \tag{6.10}$$

whereT^{SH} is a threshold due to probabilistic classifier and could be obtained from Receiver Operating Characteristic (ROC) curve.

Given the cluster centers of valid face areas, the next step is to evaluate the probability that any of these cluster centers represents the candidate face using the second probability term in (6.7). The posterior probability function $p\left(\rho_p\middle|\{f_1, f_2, \ldots, f_k\}, \rho\right)$ is simplified to the form in (6.11) since we have only cluster centers of valid face areas and each cluster contains only M dedicated features representing the identity of the candidate. The Bayesian rule changes the posterior form to the form of priors and likelihood functions.

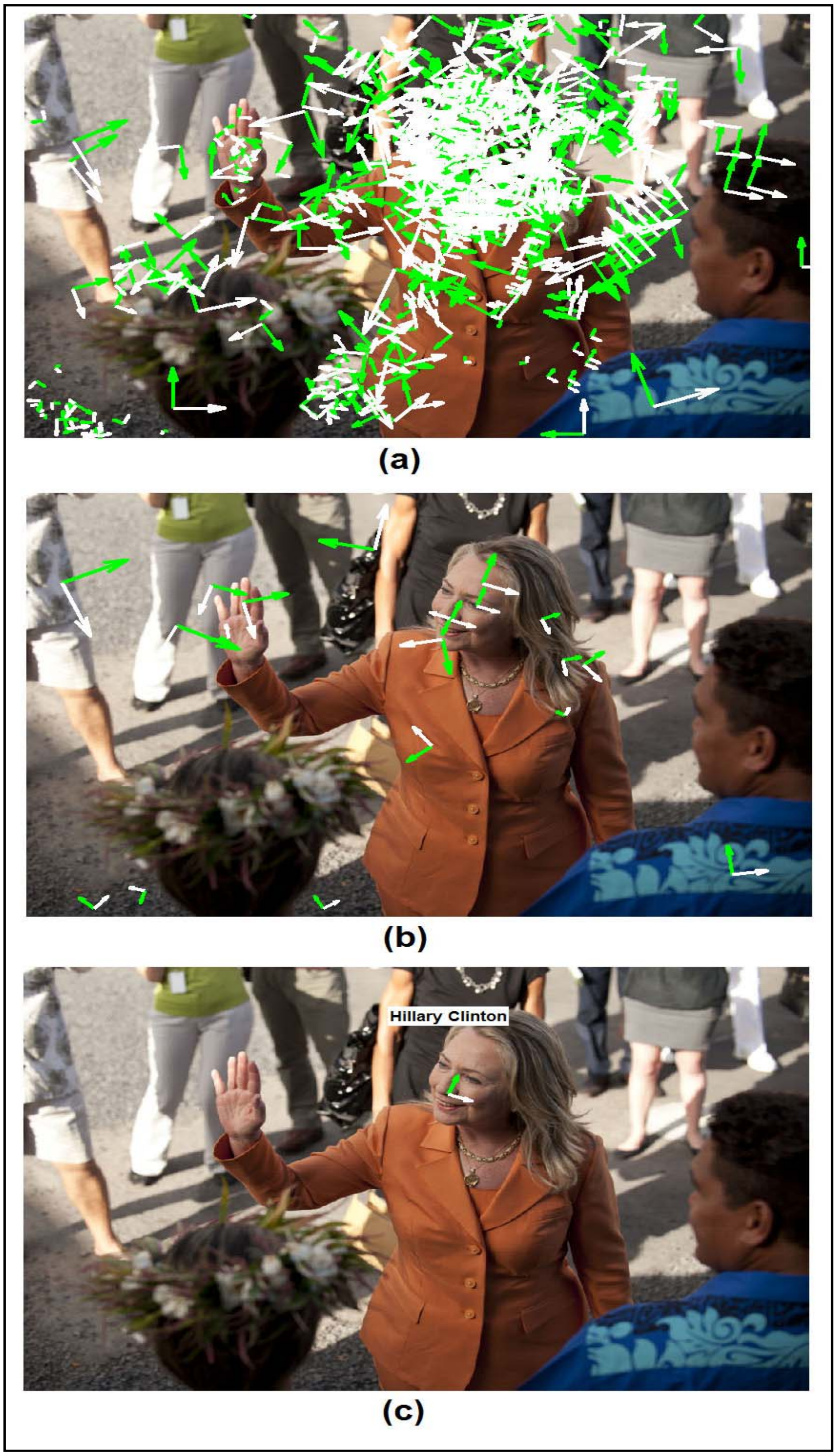


Figure 6.4. SIFT features are extracted in the query image and compared to the model. Each match to the model represents a *reference map* in a location in the test image (a). All the *reference map*s then participate in the clustering stage to find the cluster centers (b). The probability that each cluster center represents the candidate face is evaluated using the probability function in(6.10) and (6.14). The cluster center whose value in (6.10)exceeds a pre-determined threshold and has the maximum probability value in (6.14) is considered as the candidate face and labeled accordingly as seen in the (c). (This image with photo number 7902992110 is taken from the US secretary of state website labeled as "Public domain" under the free copyright license.)

$$p(\rho_p|\{f_1, f_2, \dots, f_M\}) = \frac{p(\rho_p).p(\{f_1, f_2, \dots, f_M\}|\rho_p)}{p(\{f_1, f_2, \dots, f_M\})}, \tag{6.11}$$

The function in (6.11) is further simplified to the form of (6.12) assuming the face features are independent.

$$p(\rho_p|\{f_1, f_2, \dots, f_M\}) = p(\rho_p) \times \prod_{m=1}^{M} \frac{p(f_m|\rho_p)}{p(f_m)}, \tag{6.12}$$

where $p(\rho_p)$ and $p(f_m)$ are two constant terms representing the prior probabilities of the candidate and the model features, respectively. The likelihood term $p(f_m|\rho_p)$ is evaluated using Bayesian approach as follows: Feature f_m could originate from the face of the candidate ($f_m = 1$) or not ($f_m = 0$). Considering a Beta probability distribution as a prior with parameters$\alpha_0$and α_1, the likelihood term could be evaluated as in (6.13).

$$p(f_m = 1|\rho_p) = \frac{L_m + \alpha_1}{L + \alpha_1 + \alpha_0} \tag{6.13}$$

where L_m is the number of times that feature f_m make connection between images when finding the correspondence points in the training set and L is the total number of images related to the candidate.

While the likelihood term is evaluated, we apply the test function in (6.14) by rearranging (6.12), for every cluster centers that passed the test in (6.10).

$$\prod_{m=1}^{M} p(f_m|\rho_p) > \frac{\prod_{m=1}^{M} p(f_m)}{p(\rho_p)} \times THR = T^D, \tag{6.14}$$

whereT^D is a threshold of the probabilistic classifier and could be obtained from the ROC curve at the training stage.

To determine the thresholds T^{SH} and T^D we need to evaluate a large number of test images which is not appropriate in our automatic framework. Instead, we proposed to set T^{SH} to half of the highest probability value in (6.10) which are computed for all the cluster centers. In this manner, the noise and non-face cluster centers would be rejected effectively after applying the equation in (6.10). Then, the best candidate among the remaining cluster centers is obtained by taking the cluster center with the maximum value of probability in (6.10) multiplied by probability in (6.14). The best cluster center candidate is selected as the place of the face of candidate and tagged with its name.

6.4 Experimental Results

In this section, we present the accuracy of our proposed method to automatically learn the face model and then the performance of face recognition of our proposed approach in photo albums after automatically learning from the web the face model of selected people.

6.4.1 Database

We used several sources of images in our experiment. The first is the standard color FERET dataset [75, 76] as used in the earlier experiments. 500 images of different persons at random pose are randomly selected, resized to 256x384 pixels and converted to grayscale. These images are then used to generate a general face model.

The second source is the images published in the web. We used the Google search engine to collect images of candidates to be queried. For each person, his/her name is used as the keyword and the top 400 images returned by the Google search engine are downloaded. These images are then used as the training images to construct the view invariant face model of that candidate. Since there is no supervision, there is possibility

Table 6.1. Face Pose Distribution in the Collected Photo Album/her photo album images.

Person Code	1	2	3	4	5	6	7	8	9
Number of left view images (%)	37%	45%	25%	27%	28%	28%	19%	26%	26%
Number of frontal or near frontal view images (%)	43%	34%	51%	52%	43%	50%	52%	35%	44%
Number of right view images (%)	20%	21%	24%	21%	29%	22%	29%	39%	30%
Total number of images in the test photo album	139	113	120	126	114	125	103	131	110

that there are fake, unrelated or duplicate images among the downloaded images per candidate. All the training images are resized to the resolution of 700×700 pixels or lower, ensuring that the aspect ratio of the original images is maintained. We collected images for 50 people, giving a total of 20,000 images.

The third source is the digital photo galleries issued by some official websites. These galleries are used to test the face recognition accuracy from the learnt face model. As these images appear as part of online news, they are not meta-tagged and thus not available in the returned images from the Google search. We used the photo galleries of the official websites such as the US secretary of State [93] and Council of the European Union [94]. For each candidate, about 100 photos are collected manually, so long as the face of the candidate exists in the photo. However, the faces can be in any scale and pose. Altogether, we have chosen 9 candidates (we limit to 9 as it is not easy to find large free photo albums in the web categorized as “public domain” so that we can publish the photos where necessary). These 9 candidates are the same as the first 9 candidates in the second dataset. The number of collected images and the estimated face pose for each candidate is tabulated in Table 6.1. Altogether, a total of 1081 different digital photos from nine digital photo galleries are collected. 29% of these images are of left view, 26% right view and 45% frontal faces.

6.4.2 Face Detection Performance of Automatic Model Learning Approach

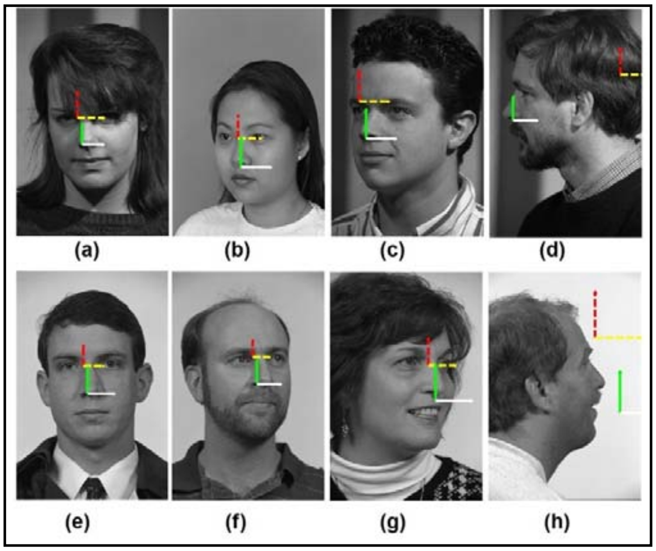


Figure 6.5. *Reference vector*s found in the 500 randomly selected images from FERET dataset. The initial values are shown by the dashed red-yellow perpendicular vectors and the final refined results are shown by the solid green-white perpendicular vectors.

After preparing a set of training images, scale invariant features [14] are extracted from them using the same method as Vedaldi et.al [95]. The SIFT parameters such as number of octaves of the Gaussian scale space and number of scale levels within each octave are determined such that at least 1,000 features are extracted from each image. Subsequently, the correspondence points among the images are found using the point matching algorithm described in chapter 3. If the number of correspondence points found between two comparing images is very high (above 100 matches), one of the images is discarded as it is likely a duplicate copy of the other. After finding the correspondence points, the *reference vector* for each face is determined as described in Sec.6.4. Figure 6.5illustrates the *reference vector*s found for some images of different poses in the training set of 500 randomly selected images from the FERET dataset. The initial *reference vector* in each image is shown using the red-yellow perpendicular dashed lines and the final *reference vector* after the iteration stops is indicated using the green-white perpendicular solid line. As shown in Figure 6.5(d), even though the initial *reference vector* is far from the correct location, the iterative process will still move it to a reasonably correct location. Figure 6.5(h) shows a case where final estimated vector is not accurate due to pose angle of more than 90 degrees, where the

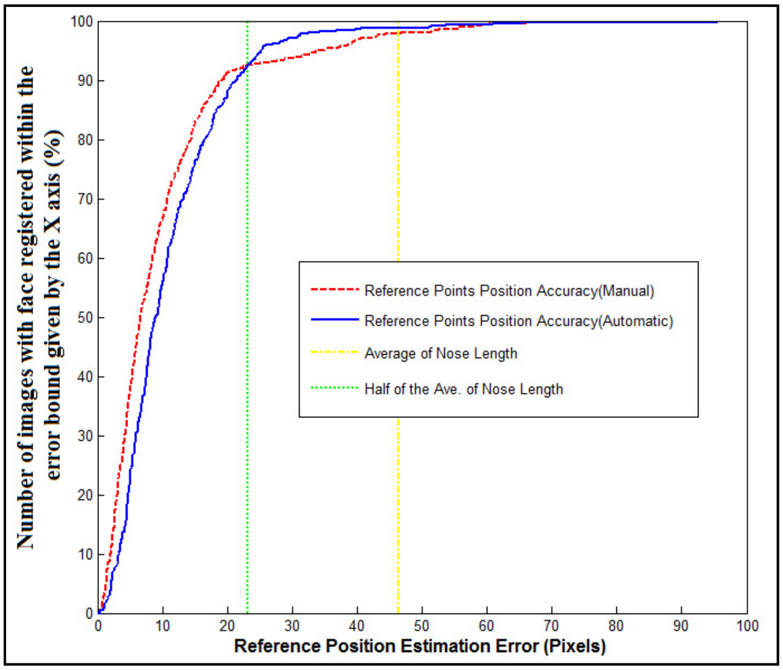


Figure 6.6. Comparison of face registration accuracy obtained on the FERET training set.

eye and mouth not clearly seen. Thus the correspondence matching is poor causing error in the *reference vector* estimation.

To determine the performance of our proposed approach statistically, we compare the accuracy of face registration which comprises face detection and localization in the first dataset, the FERET dataset. The benchmark performance for face registration is obtained by manually selecting the reference image and manually labeling the two *reference point*s of the selected reference image. The experiment is then repeated using our proposed approach where the reference image and the *reference vector* are automatically determined. The accuracy of face registration for the benchmark (manual) and automatic approaches are plotted in Figure 6.6 using the red dashed line and blue solid line, respectively. These graphs are obtained by comparing the location of the estimated base points to the ground truth (manually located) for all the 500 FERET images used. The result shows that the performance between the two are comparable and are almost the same if the commonly used acceptance criteria for face registration, which is half of the average nose length, is used.

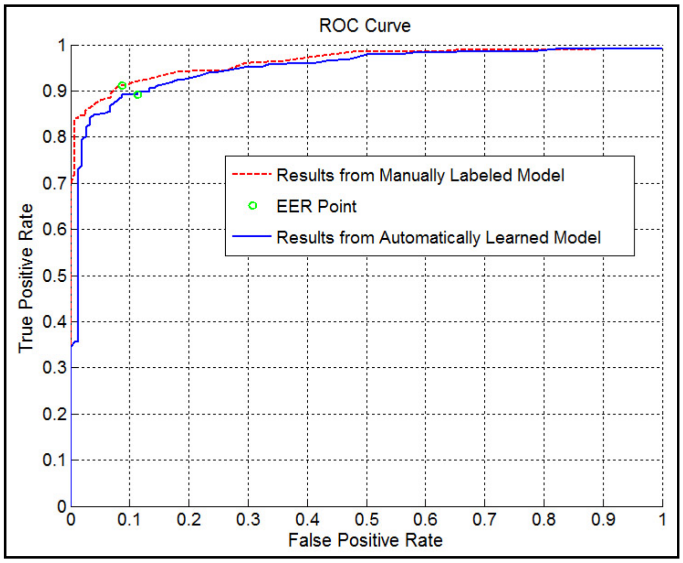


Figure 6.7. ROC curves obtained by testing the automatic and manually learnt models from 500 random images with the other 500 randomly selected images from the FERET dataset.

We also evaluated the performance of our proposed approach versus the benchmark (manual) approach by computing the ROC curves as illustrated in Figure 6.7. The automatic and manual constructed models are learnt from the previous experiment which uses 500 randomly selected images from FERET dataset. The other 500 test images were then selected randomly from the FERET dataset different from the training images. The result obtained again shows that the performance of the proposed automatic approach is only slightly lower compared to the benchmark manual approach, with the EER reducing by 3%. The reduction is still reasonable considering the full automation obtained using the proposed approach.

We repeated the experiment using the third dataset which is obtained from the web search. For each person we searched the web with his/her name and downloaded 400 images for the training as described before. Some sample results for the initial and final *reference vector*s obtained are shown in Figure 6.8. The accuracy of the *reference vector* estimation for the 9 people in database 3 is shown in Figure 6.9. It is calculated using the ratio of *reference point*'s location estimation error based on the Euclidean

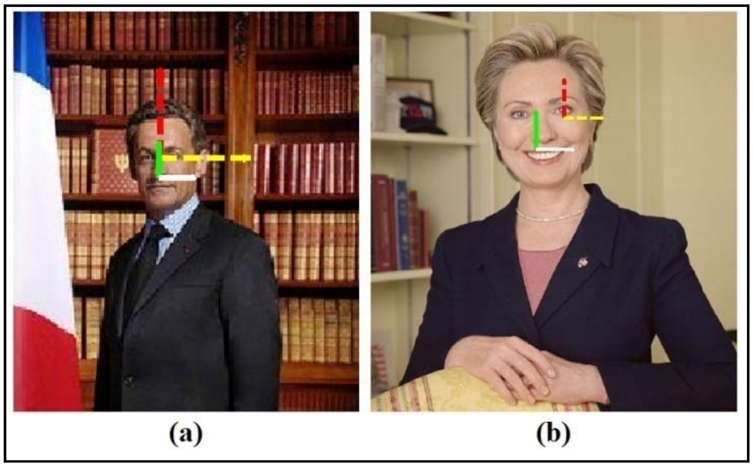


Figure 6.8. Initial and final *reference vector*s detected in two sample images obtained from the web search and collected in dataset 3. The dashed red-yellow perpendicular vectors show the initial estimation and the solid green-white perpendicular vectors show the final estimation.

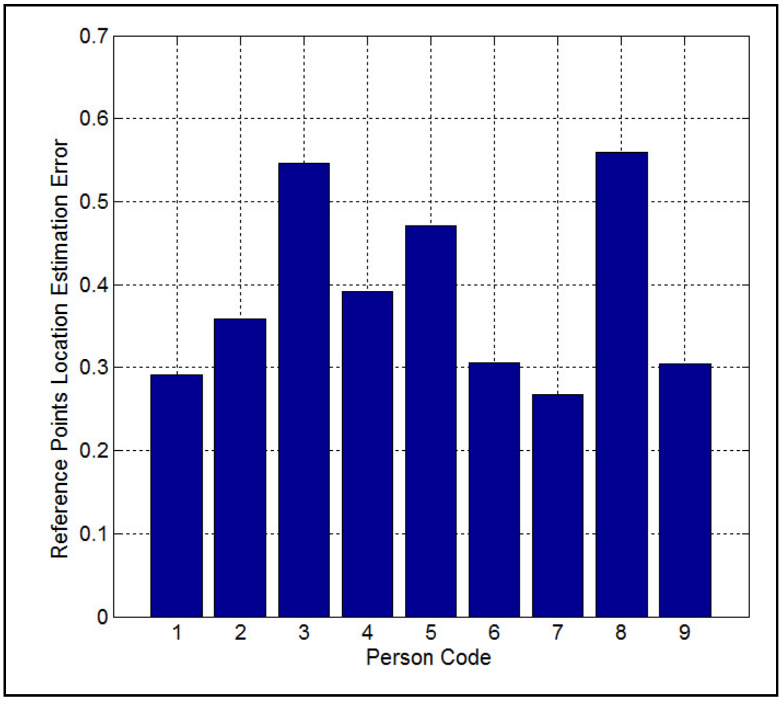


Figure 6.9. The ratio of the Euclidean distance error between the estimated *reference point* location and the ground truth (nose position of each person) with respect to the nose length for the 9 persons in dataset 3.

distance between the estimated *reference point* location and the ground truth (nose position of each person) divided by the nose length for the 9 persons in dataset 3.

Table 6.2. The Results of Testing the Model Constructed over 50 Different Individual Identities.

Person Code		1	2	3	4	5	6	7	8	9	10	11
100 Training images	Number of duplicate, fake, unrelated and low resolution images	50	26	34	39	47	51	32	26	22	29	35
	Number of features in the model	18	256	86	96	218	71	86	321	152	184	176
100 Test images	Accuracy (%)	32	85	67	67	71	60	58	77	76	71	74

Person Code	12	13	14	15	16	17	18	19	20	21	22	23	24
Number of duplicate, fake, unrelated and low resolution images	25	39	30	32	30	21	24	30	31	34	67	20	33
Number of features in the model	601	60	158	114	190	112	153	323	134	156	53	218	208
Accuracy (%)	87	59	75	77	76	72	70	86	86	71	43	91	78

Person Code	25	26	27	28	29	30	31	32	33	34	35	36	37
Number of duplicate, fake, unrelated and low resolution images	39	24	25	32	45	27	34	43	14	24	41	48	37
Number of features in the model	74	185	271	61	62	162	257	27	319	391	138	36	543
Accuracy (%)	60	75	77	63	58	77	64	15	88	84	67	50	79

Person Code	38	39	40	41	42	43	44	45	46	47	48	49	50
Number of duplicate, fake, unrelated and low resolution images	33	34	23	64	20	25	28	37	36	19	38	40	40
Number of features in the model	443	126	289	96	174	382	381	181	264	213	532	248	80
Accuracy (%)	82	82	81	60	83	75	81	76	72	79	80	86	65

6.4.3 Face Tagging of Multi-Pose Images

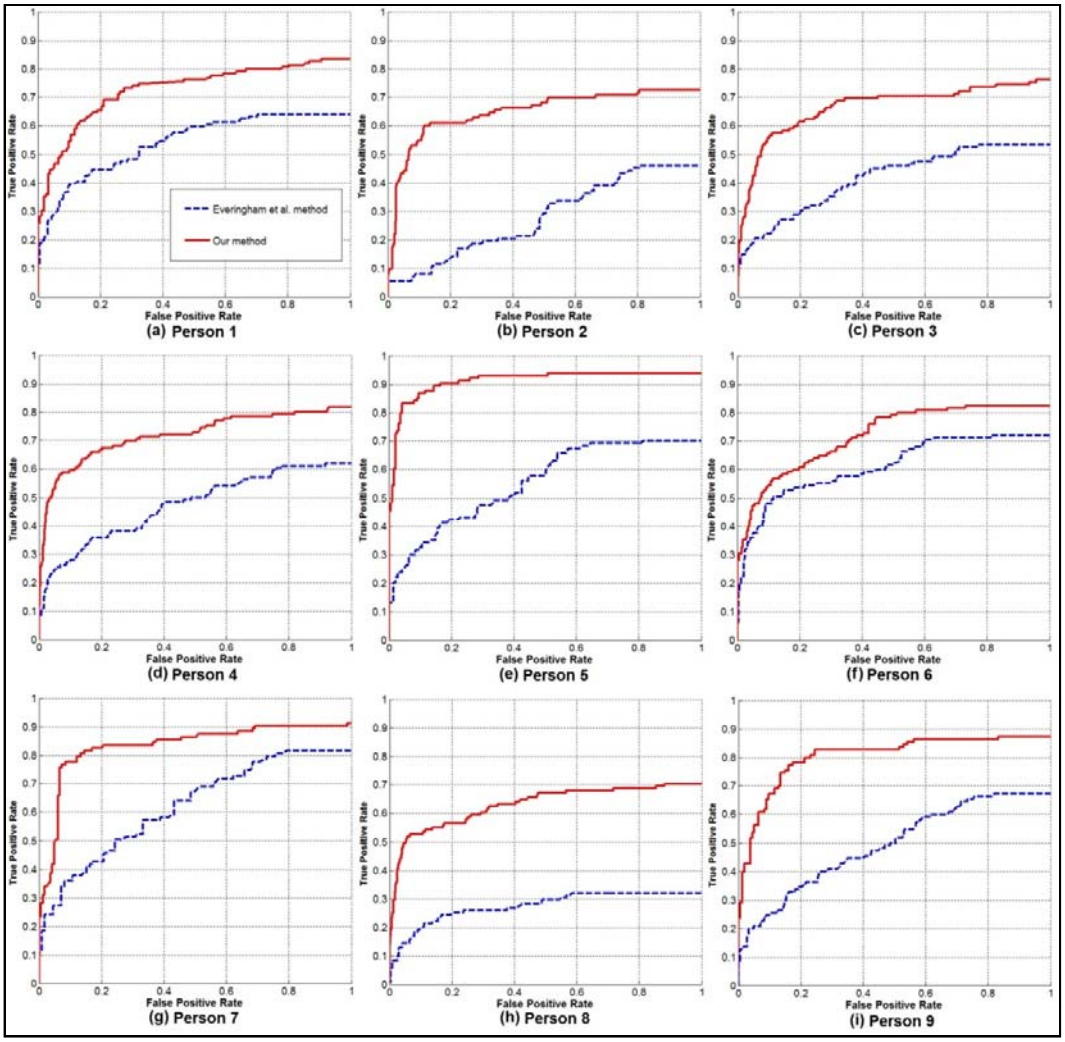
Here, we present the capability of the proposed automatic learning approach to learn a model of a queried person (candidate) and find its face in the query images. To ensure that the proposed method is capable of supporting general query, we performed query for 50 individual identities.

Table 6.3. The Results of Testing Some Queries' Models after Training with 200 Images.

Person Code		1	22	32	36
200 Training images	Number of Duplicate, Fake, Unrelated and Low Resolution Images	88	131	99	114
	Number of Features in the Model	118	145	148	218
100 Test images	Accuracy (%)	80	64	63	63

For each query, we used the candidate's name and query the web. The top 200 results were downloaded and divided into two groups of 100 members randomly, one as a training set and the other as the test set. Using the 100 training images, the candidate's face model is developed and used to evaluate the remaining 100 test images. Table 6.2tabulates the number of unusable images in the training due to duplicates, fake, unrelated and low resolution images versus the performance of face recognition achieved using the model generated. Note that the statistics of the unusable images are obtained manually but not excluded from the training sets.

We let the proposed approach classify them and thus if these unrelated images are not properly removed, it will reduce the accuracy of the face recognition. This is clearly seen in Table 6.2. For instance, the candidate 1, 22, 32 and 36 have more than 40% of unusable images. In addition, these images do not have high number of features. Thus the face model obtained is weak, resulting in face recognition performance below 50%. This can be improved if more training images are used, and is easy since our proposed approach is fully automated. To validate this, we retrained the model for these cases with another 100 images randomly selected from the next top 200 training images obtained from the web search. Then the system is tested on the same 100 testing images as before. The results are tabulated in Table 6.3. We can see that the performance of face recognition increases as the candidate model becomes more stable with additional training images and with the increase in the number of features. Another important factor that improves the accuracy is the diversity of the face pose in



(a) Person 1 (b) Person 2 (c) Person 3

(d) Person 4 (e) Person 5 (f) Person 6

(g) Person 7 (h) Person 8 (i) Person 9

Figure 6.10. Shows the ROC curves obtained from applying the models of nine different people to their photo albums. The comparison shows that our method outperforms the Everingham et al. [52] method significantly.

the training images. The higher the number of images with various poses, the higher is the ability of the face recognition system to correctly recognize the candidate in different poses.

6.4.4 Face Tagging in Digital Photo Album

We furthermore evaluated our proposed approach for face tagging of the candidates in their digital photo albums provided in the third dataset. We follow the state-of-the-art

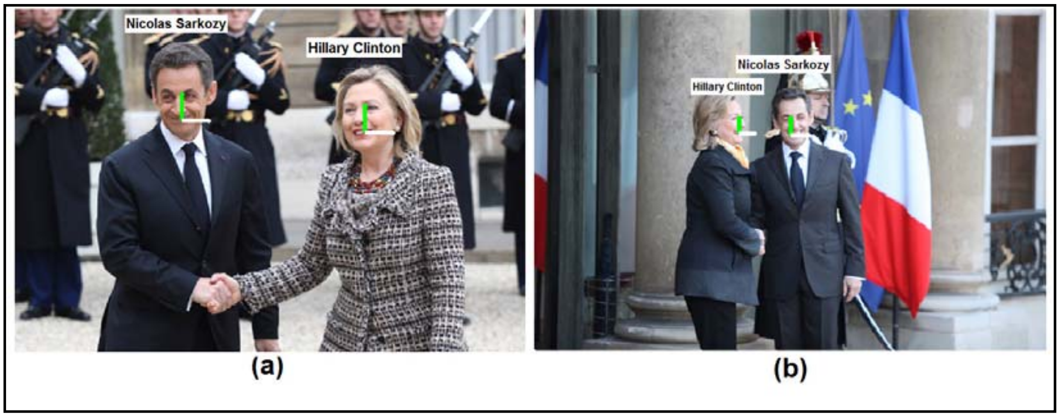


Figure 6.11. Shows the two candidates have been identified from multi-poses in different scales. (Image in (a) with photo number 5549582433 and image in (b) with photo number 5536783127 were adopted from the US secretary of state website labeled as "Public domain" under the free copyright license.)

[7, 8] and used the method proposed by Everingham et al. [52] to compare the performance of the face tagging.

To make sure that the obtained model has sufficient number of features for the candidate, we used the top 400 downloaded images to train the candidate face model. The main difference between our method and Everingham et al. is that after we have downloaded the training images automatically, we fed all of them to our framework without any manual intervention, including removing unrelated or unusable images. However, for Everingham et al., all the 400 faces were manually cropped and the unrelated and poor faces discarded before using them for training. Such manual intervention was repeated for all the nine people used in the query, requiring manual processing of 3600 training images.

We plotted the ROC curves for each candidate in his/her photo albums as shown in Figure 6.10. The results show that our proposed method totally outperforms the method proposed by Everingham et al. [52]. The major shortcoming of their method is that it is not able to handle non-frontal images. However, our proposed method is capable of identifying the multi-view faces as shown Figure 6.11 and Figure 6.12(a). Figure 6.12(b) shows a situation where our method failed due to inappropriate lighting

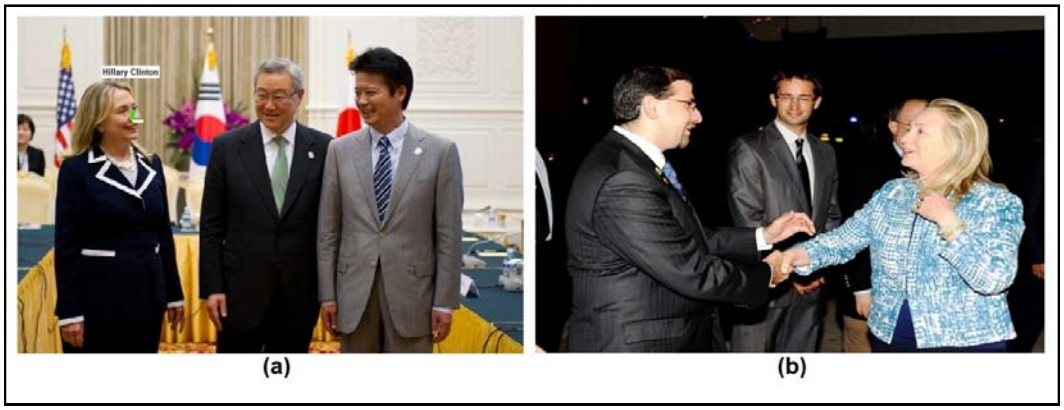


Figure 6.12. In (a) it shows the queried person has been identified from multi-poses where other state of the art are not able to recognize her from these views. In (b) shows a situation where our method failed to identify the candidate due to poor lighting condition. (Image in (a) with photo number 7557303538 and image in (b) with photo number 7583423068 were adopted from the US secretary of state website labeled as “Public domain” under the free copyright license.)

condition. The dark background causes the face region to be over-exposed resulting in the lack of reliable features needed for correspondence and matching.

It is worthwhile to note that the proposed approach is general and not limited to obtaining the training images using web search. In fact, a large unlabelled photo album can be used and the proposed method will be able to cluster them and use the largest cluster to develop the face model for recognition, assuming that the owner of the photo album will have the most frequent appearance.

6.5 Concluding Remarks

A fully automatic and unified framework is introduced to detect, localize and tag faces in query images where the faces could be rotated, occluded or appear in multi-poses and scales. Given the name of the candidate, the proposed method automatically learns the face model of the candidate. This is done by clustering the most likely representative images of the candidate from the output of the web search engine. Then the method will localize the face regions in these images and determines the model of the candidate’s face. With the face model of the candidate, given a query image, the proposed approach will be able to recognize and tag the image where the candidate’s face occur, independent of the scale and pose. Despite being fully automatic, the experimental results on publicly available images show that the proposed approach outperform the state-of-the-art approach in terms of recognition accuracy. Therefore, it supports fully automatic photo album search without the user having to manually train the face recognition system so long as there are sufficient images of the person in the web. Such images can be learnt without requiring the images to be tagged, but instead just uploading the images into proper entry such as users uploading their photos to their personal photo album or Facebook account.

Chapter 7

Conclusion and Recommendations

7.1 Conclusion

In this study, a multi-stage framework has been proposed which detects, localizes and identifies faces from multi-views. The main focus of the project is to enhance the unsupervised machine learning approach to detect, localize and recognize multi-view faces with minimal or no human intervention during the training stage. The proposed framework has 3 key contributions, Firstly, an enhanced matching algorithm is introduced which is able to find the reliable correspondence points between the face images of multi-poses. It is able to find the real correspondence points between comparing images even though the number of real matched points is small compared to all the found matched points. Secondly, a registration method is proposed that uses a constellation connection to register face images of multi-poses to a predefined reference map. The reference map allows the images of different scales and views to register properly to a reference image through their links in the constellation connection. Thirdly, a probabilistic classification algorithm is proposed to detect and localize multiple faces of different view in a given test image. The method is also able to handle partially occluded faces. The experimental results obtained have shown that the proposed method is able to detect rotated, occluded and multi-view faces from multi scales more accurately than the state of the art face detection methods. This is achieved despite the proposed method requiring manual labeling of only 2 points in only one reference image at the training stage.

Furthermore, we extended the proposed framework to learn a face model fully automatically without even any manual intervention, so long as there are some labeled images available, such as those in the web. This is then extended to include developing a face model for a person which is then shown to be able to be use for face recognition from multi-view. Such proposed method could then be applied to detect and identify the person of interest in his/her digital photo albums. The proposed method uses the ordinary images from the web and constructs a multi-view model for the person of interest. Unlike conventional methods which uses different stages in cascade (e.g. face detection stage, face alignment stage and face identification stage) to identify a person, the proposed method unified these stages to allow for multi-view face recognition. Consequently, its performance is not limited to the performance of each stage individually. The experimental results conducted have shown that the proposed method is able to learn based on images from the web and subsequently used the learnt model to detect and recognize multi-views faces of the candidate better than the state of the art methods.

7.2 Recommendations for the Future Research Works

The proposed framework opens an exciting area of research on face and object detection and recognition from different views. Some suggested future research directions in this area include:

(1) Changing the feature extraction method

In the proposed method, we used the SIFT features and extracted the facial features using the method proposed by Lowe. However, the proposed framework is general and any other scale invariant features could be used. It is also important to find a way to fit the new feature to the framework, especially the feature of scale and rotation invariants. As most of the computation time for our method is spent on feature extraction and correspondence, any improvement in the accuracy and speed will enhance the propose framework further.

Our method also has weakness in dealing with very small, blurred and low resolution images. This weakness arise due to the limitation of the SIFT feature which requires images with proper resolution and quality. Since pixel based features such as Haar-like or LBP work better with small images, we suggest the fusion of these features with the proposed method. However, such pixel based features are not scale and rotation invariant. Therefore, it is important to investigate the algorithm to extract some features from the image that gives location, scale and orientation for each extracted feature besides the appearance descriptor similar to the pixel based methods.

(2) Modifying the method to estimate the pose angle

The proposed constellation connection links the connection between images of multi poses. Considering the first image as a frontal image, the next images linked to the reference image will have pose angle similar or close to frontal view. Consequently, the images added to the constellation in the next layer should be near to the frontal or semi-oblique faces. This phenomenon is repeated through all the images in the constellation. Hence we expect that the images which are added in the last layer have the view near to profile faces. If we could find a way to estimate the angle for each added image, the angle value would be useful in the detection stage. It will allow even the angle of the view to be computed or estimated.

(3) Extending the framework to the age estimation

The constellation concept introduced in this project has the capability to be extended for the other hierarchical tasks such as age estimation. In this case, each cluster of constellation could represent a certain age group. Hence, the constellation concept could be applied to make connection between faces of different ages. In each layer, faces of almost similar ages are added to the reference. However, the age difference between the images in the first layer and the last layer could be very large.

(4) Extending the idea for recognizing other objects

Although the proposed framework is used for face detection and recognition, it could be extended to other objects. The main issue with other objects in general is the wide variation between the object of the same class. For instance, cars have very different body shape and size, tire size, lamp shape and window size and shape. However, the overall shape for a particular car type (eg. sedan, hatchback etc) is almost the same.. The main challenge then is to handle the different parts with different designs to be classified under the same class. If the problem is solved, the proposed method could be generalized to almost all man-made objects as they follow the same concept.

Author's Publications

Conference Papers

1. S.M.H. Anvar, W. Yau and E.K. Teoh, "Fast face detection and localization from multi-views using statistical approach," *Proc. 8th IEEE International Conference on Information, Communications and Signal Processing (ICICS 2011),* Singapore, pp. 1-5, 2011.

2. S.M.H. Anvar, W.Y. Yau and E.K. Teoh, "Finding the Correspondence Points in Images of Multi-Views," *Proc. 8th IEEE Conference on Signal Image Technology and Internet-based Systems (SITIS 2012)*, Sorrento, Italy, pp. 275-280, 2012.

3. S.M.H. Anvar, W.Y. Yau, K. Nandakumar and E.K. Teoh, "Estimating In-Plane Rotation Angle for Face Images from Multi-Poses," *Proc. IEEE Symposium on Computational Intelligence in Biometrics and Identity Management (CIBIM 2013)*, Singapore, April 2013.

Journal Papers

1. S.M.H. Anvar, W.Y. Yau and E.K. Teoh, "Multi-View Face Detection and Registration Requiring Minimal Manual Intervention," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, pp. 2484-2497, 2013.

2. S.M.H Anvar, W. Y. Yau and E. K. Teoh., "An Automatic Multi-View Model Learning for Character Identification and Face Tagging in Digital Photo Albums," *Submitted to IEEE Trans. Circuits and Systemsfor Video Technology, September* 2013.

Bibliography

[1] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, pp. 511-518, 2001.

[2] H. Schneiderman and T. Kanade, “Object detection using the statistics of parts,” *Int. J. Comput. Vis.*, vol. 56, no. 3, pp. 151-177, 2004, DOI: 10.1023/B:VISI.0000011202.85607.00.

[3] C. Huang, H.Z. Ai, Y. Li and S.H. Lao, “High-performance rotation invariant multiview face detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 671-686, 2007, DOI: 10.1109/TPAMI.2007.1011.

[4] Y. Tian, W. Liu, R. Xiao, et al., “A Face Annotation Framework with Partial Clustering and Interactive Labeling,” *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, pp. 1-8, 2007.

[5] C. Ding-Hrong and T. Shiang-En, “A Name Recommendation Photo Album Using Probability Neural Network,” *Proc. 5th International Conference on Information Assurance and Security (IAS 2009)*, pp. 379-382, 2009.

[6] L. Chen, B. Hu, L. Zhang, et al., “Face annotation for family photo album management,” *International Journal of Image and Graphics*, vol. 3, pp. 1-14, 2003.

[7] M. Guillaumin, T. Mensink, J. Verbeek and C. Schmid, “Face Recognition from Caption-Based Supervision,” *Int. J. Comput. Vision*, vol. 96, no. 1, pp. 64-82, 2012, DOI: 10.1007/s11263-011-0447-x.

[8] P.T. Pham, M.F. Moens and T. Tuytelaars, “Cross-Media Alignment of Names and Faces,” *IEEE Transactions on Multimedia*, vol. 12, no. 1, pp. 13-27, 2010, DOI: 10.1109/tmm.2009.2036232.

[9] A. Kapoor, G. Hua, A. Akbarzadeh and S. Baker, “Which faces to tag: Adding prior constraints into active learning,” *Proc. 12th IEEE International Conference on Computer Vision (ICCV 2009)*, pp. 1058-1065, 2009.

[10] C. JaeYoung, W. De Neve, R. Yong Man and K.N. Plataniotis, “Face annotation for personal photos using collaborative face recognition in online social networks,” *Proc. 16th International Conference on Digital Signal Processing (DSP 2009)*, pp. 1-8, 2009.

[11] S. Kaneko, Y. Satoh and S. Igarashi, “Using selective correlation coefficient for robust image registration,” *Pattern Recognit.*, vol. 36, no. 5, pp. 1165-1173, 2003, DOI: 10.1016/S0031-3203(02)00081-X

[12] P. Viola and W.M. Wells, “Alignment by maximization of mutual information,” *Int. J. Comput. Vis.*, vol. 24, no. 2, pp. 137-154, 1997, DOI: 10.1023/A:1007958904918.

[13] E.D. Castro and C. Morandi, “Registration of translated and rotated images using finite Fourier transforms,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9, no. 5, pp. 700-703, 1987, DOI: 10.1109/TPAMI.1987.4767966.

[14] D.G. Lowe, “Distinctive image features from scale-invariant keypoints,” *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91-110, 2004, DOI: 10.1023/B:VISI.0000029664.99615.94.

[15] K. Mikolajczyk and C. Schmid, “A performance evaluation of local descriptors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1615-1630, 2005, DOI: 10.1109/TPAMI.2005.188

[16] H. Bay, T. Tuytelaars and L. Van Gool, “SURF: Speeded up robust features,” *Proc. 9th European Conference on Computer Vision (ECCV 2006)*, Graz, Austria, pp. 404-417, 2006.

[17] S.J. Thomas, B.A. MacDonald and K.A. Stol, “Real-Time Robust Image Feature Description and Matching,” *Proc. Asian Conference on Computer Vision (ACCV 2010), Pt 2*, Queenstown, New Zealand, pp. 334-345, 2011.

[18] K. Yan and R. Sukthankar, “PCA-SIFT: a more distinctive representation for local image descriptors,” *Proc. IEEE Comput. Soc. Conference on Computer Vision and Pattern Recognition (CVPR 2004)*, pp. 506-513, 2004.

[19] S. Belongie, J. Malik and J. Puzicha, “Shape matching and object recognition using shape contexts,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509-522, 2002.

[20] S. Lazebnik, C. Schmid and J. Ponce, “A sparse texture representation using affine-invariant regions,” *Proc. IEEE Comput. Soc. Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, pp. 319-324, 2003.

[21] L.J.V. Gool, T. Moons and D. Ungureanu, “Affine/ Photometric Invariants for Planar Intensity Patterns,” *Proc. 4th European Conference on Computer Vision (ECCV 1996)*, pp. 642-651, 1996.

[22] F. Schaffalitzky and A. Zisserman, “Multi-view Matching for Unordered Image Sets, or "How Do I Organize My Holiday Snaps?",” *Proc. 7th European Conference on Computer Vision (ECCV 2002)*, pp. 414-431, 2002.

[23] J.J. Koenderink and A.J.v. Doom, “Representation of local geometry in the visual system,” *Biol. Cybern.*, vol. 55, no. 6, pp. 367-375, 1987, DOI: 10.1007/BF00318371.

[24] W.T. Freeman and E.H. Adelson, “The design and use of steerable filters,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891-906, 1991.

[25] G. Dorko and C. Schmid, “Selection of scale-invariant parts for object class recognition,” *Proc. 9th IEEE Comput. Soc. International Conference on Computer Vision (ICCV 2003)*, Nice, France, pp. 634-640, 2003.

[26] D.G. Lowe, “Object recognition from local scale-invariant features,” *Proc. 7th IEEE International Conference on Computer Vision (ICCV 1999)*, pp. 1150-1157, 1999.

[27] K. Mikolajczyk and C. Schmid, “Indexing based on scale invariant interest points,” *Proc. 8th IEEE International Conference on Computer Vision (ICCV 2001)*, pp. 525-531, 2001.

[28] H. Schneiderman and T. Kanade, “Probabilistic modeling of local appearance and spatial relationships for object recognition,” *Proc. IEEE Comput. Soc.*

Conference on Computer Vision and Pattern Recognition (CVPR 1998), Santa barbara, Ca, pp. 45-51, 1998.

[29] L. Fei-Fei, R. Fergus and P. Perona, “A Bayesian approach to unsupervised one-shot learning of object categories,” *Proc. 9th IEEE Comput. Soc. International Conference on Computer Vision (ICCV 2003)*, Nice, France, pp. 1134-1141, 2003.

[30] M.A. Fischler and R.C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography,” *Commun. ACM*, vol. 24, no. 6, pp. 381-395, 1981, DOI: 10.1145/358669.358692.

[31] A. Goshtasby and G.C. Stockman, “Point Pattern-Matching Using Convex-Hull Edges,” *IEEE Trans. Syst. Man Cybern.*, vol. 15, no. 5, pp. 631-637, 1985.

[32] K. Mikolajczyk and C. Schmid, “Scale & affine invariant interest point detectors,” *Int. J. Comput. Vis.*, vol. 60, no. 1, pp. 63-86, 2004, DOI: 10.1023/B:VISI.0000027790.02288.f2.

[33] M.H. Yang, D.J. Kriegman and N. Ahuja, “Detecting faces in images: A survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 34-58, 2002.

[34] M.A. Turk and A.P. Pentland, “Face Recognition Using Eigenfaces,” *Proc. IEEE Comput. Soc. Conference on Computer Vision and Pattern Recognition (CVPR 1991)*, Lahaina, HI, pp. 586-591, 1991.

[35] H.A. Rowley, S. Baluja and T. Kanade, “Neural network-based face detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 23-38, 1998, DOI: 10.1109/34.655647.

[36] S.J. McKenna, S. Gong and Y. Raja, “Modelling Facial Colour and Identity with Gaussian Mixtures,” *Pattern Recognit.*, vol. 31, no. 12, pp. 1883-1892, 1998, DOI: 10.1016/S0031-3203(98)00066-1.

[37] R.L. Hsu, M. Abdel-Mottaleb and A.K. Jain, “Face detection in color images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 696-706, 2002.

[38] R.E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” *Mach. Learn.*, vol. 37, no. 3, pp. 297-336, 1999.

[39] S.Z. Li and Z. Zhenqiu, “FloatBoost learning and statistical face detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1112-1123, 2004, DOI: 10.1109/TPAMI.2004.68

[40] M. Toews and T. Arbel, “Detection, Localization, and Sex Classification of Faces from Arbitrary Viewpoints and under Occlusion,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 9, pp. 1567-1581, 2009, DOI: 10.1109/TPAMI.2008.233.

[41] J.C. Chen and J.J.J. Lien, “A view-based statistical system for multi-view face detection and pose estimation,” *Image Vis. Comput.*, vol. 27, no. 9, pp. 1252-1271, 2009, DOI: 10.1016/j.imavis.2008.11.004.

[42] M.J. Jones and P. Viola, “Fast multi-view face detection,” Technical Report TR2003-96, Mitsubishi Electric Research Laboratories, 2003.

[43] D. Comaniciu and P. Meer, “Mean shift: A robust approach toward feature space analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603-619, 2002, DOI: 10.1109/34.1000236

[44] G.B. Huang, V. Jain and E. Learned-Miller, “Unsupervised Joint Alignment of Complex Images,” *Proc. 11th IEEE International Conference on Computer Vision (ICCV 2007)*, pp. 1-8, 2007.

[45] Y. Ma, X. Ding, Z. Wang and N. Wang, “Robust precise eye location under probabilistic framework,” *Proc. 6th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2004)*, pp. 339-344, 2004.

[46] M. Turk and A. Pentland, “Eigenfaces for Recognition,” *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71-86, 1991.

[47] P.N. Belhumeur, J.P. Hespanha and D.J. Kriegman, “Eigenfaces vs. Fisherfaces: recognition using class specific linear projection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711-720, 1997.

[48] J. Wright and G. Hua, “Implicit elastic matching with random projections for pose-variant face recognition,” *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pp. 1502-1509, 2009.

[49] T. Ojala, M. Pietikainen and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971-987, 2002.

[50] Z. Baochang, G. Yongsheng, Z. Sanqiang and L. Jianzhuang, “Local Derivative Pattern Versus Local Binary Pattern: Face Recognition With High-Order Local Pattern Descriptor,” *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533-544, 2010, DOI: 10.1109/TIP.2009.2035882.

[51] S.Z. Li, R.F. Chu, S.C. Liao and L. Zhang, “Illumination invariant face recognition using near-infrared images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 627-639, 2007, DOI: 10.1109/TPAMI.2007.1014.

[52] M. Everingham, J. Sivic and A. Zisserman, “Hello! My name is... Buffy -- Automatic Naming of Characters in TV Video,” *Proc. the British Machine Vision Conference (BMVC 2006)*, 2006.

[53] M. Guillaumin, J. Verbeek and C. Schmid, “Is that you? Metric learning approaches for face identification,” *Proc. 11th IEEE International Conference on Computer Vision (ICCV 2009)*, Kyoto, Japan, pp. 498 - 505, 2009.

[54] D. Fleck and Z. Duric, “Using Local Affine Invariants to Improve Image Matching,” *Proc. 20th IEEE International Conference on Pattern Recognition (ICPR 2010)*, pp. 1844-1847, 2010.

[55] M. Liyong, S. Yude, F. Naizhang and L. Zheng, “Image Fast Template Matching Algorithm Based on Projection and Sequential Similarity Detecting,” *Proc. 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP 2009)*, pp. 957-960, 2009.

[56] D. Chenguang, J. Song and Z. Yongsheng, “Multiple-View Matching AMMGC Model for Three-Line-Array Digital Images,” *Proc. International Symposium on Information Science and Engineering (ISISE 2008)*, pp. 139-142, 2008.

[57] M. Perd'och, J. Matas and O. Chum, "Epipolar Geometry from Two Correspondences," *Proc. 18th IEEE International Conference on Pattern Recognition (ICPR 2006),* Hong Kong, pp. 215-219, 2006.

[58] F.H. Cheng, "Point pattern matching algorithm invariant to geometrical transformation and distortion," *Pattern Recognit. Lett.*, vol. 17, no. 14, pp. 1429-1435, 1996.

[59] R.C. Lo and W.H. Tsai, "Perspective-transformation-invariant generalized Hough transform for perspective planar shape," *Pattern Recognit.*, vol. 30, no. 3, pp. 383-396, 1997.

[60] O. Chum and J. Matas, "Optimal Randomized RANSAC," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1472-1482, 2008, DOI: 10.1109/TPAMI.2007.70787

[61] J. Matas and O. Chum, "Randomized RANSAC with sequential probability ratio test," *Proc. 10th IEEE International Conference on Computer Vision (ICCV 2005)*, pp. 1727-1732, 2005.

[62] P.H.S. Torr, A. Zisserman and S.J. Maybank, "Robust detection of degenerate configurations while estimating the fundamental matrix," *Comput. Vis. Image Underst.*, vol. 71, no. 3, pp. 312-333, 1998, DOI: 10.1006/cviu.1997.0559.

[63] A. Vedaldi, "SIFT for Matlab," Available: http://www.vlfeat.org/~vedaldi/code/sift.html, 2013.

[64] K. Mikolajczyk and C. Schmid, Available: available at http://lear.inrialpes.fr/people/mikolajczyk/Database/index.html, 2011.

[65] C. Strecha, W. von Hansen, L. Van Gool, et al., "On Benchmarking camera calibration and multi-view stereo for high resolution imagery," *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008)*, Anchorage, AK, pp. 2838-2845, 2008.

[66] FERET, "The Color FERET Database," Available: available at http://www.nist.gov/itl/iad/ig/feret.cfm, 2011.

[67] H. Schneiderman and T. Kanade, "CMU Profile Face Images " Available: available at http://vasc.ri.cmu.edu//idb/html/face/profile_images/index.html, 2001.

[68] G.B. Huang, M. Ramesh , M. Berg and E. Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments," Technical Report 07-49, University of Massachusetts, Amherst, online available at http://vis-www.cs.umass.edu/lfw/, Oct. 2007.

[69] K.P. Murphy, *Machine Learning: A Probabilistic Perspective*, MIT Press, 2012.

[70] D.J.C. Mackay, *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, 2003.

[71] H.M. Wallach, "Evaluation Metrics for Hard Classifiers," Technical Report, Cavendish Lab., Univ. Cambridge, Cambridge, online available at http://www.inference.phy.cam.ac.uk/hmw26/, 2006.

[72] Z.H. Khan and I.Y.H. Gu, "Joint Feature Correspondences and Appearance Similarity for Robust Visual Object Tracking," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 3, pp. 591-606, 2010.

[73] T.F. Cootes, C.J. Twining, V.S. Petrovic, et al., "Computing Accurate Correspondences across Groups of Images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1994-2005, 2010, DOI: 10.1109/TPAMI.2009.193.

[74] B. Zitova and J. Flusser, "Image registration methods: a survey," *Image Vis. Comput.*, vol. 21, no. 11, pp. 977-1000, 2003, DOI: 10.1016/s0262-8856(03)00137-9.

[75] P.J. Phillips, H. Moon, S.A. Rizvi and P.J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090-1104, 2000, DOI: 10.1109/34.879790.

[76] P.J. Phillips, H. Wechsler, J. Huang and P.J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image Vis. Comput.*, vol. 16, no. 5, pp. 295-306, 1998, DOI: 10.1016/s0262-8856(97)00070-x.

[77] V. Jain and E. Learned-Miller, "FDDB: A Benchmark for Face Detection in Unconstrained Settings," Technical Report UM-CS-2010-009, Dept. of Computer Science, University of Massachusetts, Amherst, online available at http://vis-www.cs.umass.edu/fddb/, 2010.

[78] M. Toews and T. Arbel, "A statistical parts-based model of anatomical variability (vol 26, pg 497, 2007)," *IEEE Trans. Med. Imaging*, vol. 26, no. 5, pp. 757-757, 2007, DOI: 10.1109/tmi.2006.895907.

[79] V. Jain and E. Learned-Miller, "Online domain adaptation of a pre-trained cascade of classifiers," *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2011)*, pp. 577-584, 2011.

[80] J. Li, T. Wang and Y. Zhang, "Face detection using SURF cascade," *Proc. IEEE International Conference on Computer Vision Workshops (ICCV Workshops 2011)*, pp. 2183-2190, 2011.

[81] P. Viola and M.J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137-154, 2004, DOI: 10.1023/B:VISI.0000013087.49260.fb.

[82] K. Mikolajczyk, C. Schmid and A. Zisserman, "Human detection based on a probabilistic assembly of robust part detectors," *Proc. 8th European Conference on Computer Vision (ECCV 2004)*, Prague, Czech Republic, pp. 69-82, 2004.

[83] A. Vedaldi, "SIFT Implemenation in C++ (SIFT++)," Available: http://www.vlfeat.org/~vedaldi/code/siftpp.html, 2013.

[84] R. Chellappa, P. Sinha and P.J. Phillips, "Face Recognition by Computers and Humans," *Computer*, vol. 43, no. 2, pp. 46-55, 2010.

[85] M. Bicego, E. Grosso, A. Lagorio, et al., "Distinctiveness of faces: A computational approach," *ACM Trans. Appl. Percept.*, vol. 5, no. 2, pp. 1-18, 2008, DOI: 10.1145/1279920.1279925.

[86] M. Bicego, A. Lagorio, E. Grosso and M. Tistarelli, “On the Use of SIFT Features for Face Authentication,” *Proc. Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, pp. 35-35, 2006.

[87] L. Jun, Y. Ma, E. Takikawa, et al., “Person-Specific SIFT Features for Face Recognition,” *Proc. Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, pp. 593-596, 2007.

[88] D.R. Kisku, A. Rattani, E. Grosso and M. Tistarelli, “Face Identification by SIFT-based Complete Graph Topology,” *Proc. Automatic Identification Advanced Technologies, 2007 IEEE Workshop on*, pp. 63-68, 2007.

[89] P.J. Phillips, P.J. Flynn, T. Scruggs, et al., “Overview of the face recognition grand challenge,” *Proc. IEEE Comput. Soc. Conference on Computer Vision and Pattern Recognition (CVPR 2005)*, pp. 947-954, 2005.

[90] S. Baker, “The CMU Pose, Illumination, and Expression (PIE) database,” Available: http://vasc.ri.cmu.edu/idb/html/face/, 2013.

[91] T.L. Berg and D.A. Forsyth, “Animals on the Web,” *Proc. IEEE Comput. Soc. Conference on Computer Vision and Pattern Recognition (CVPR 2006)*, pp. 1463-1470, 2006.

[92] R. Fergus, L. Fei-Fei, P. Perona and A. Zisserman, “Learning object categories from Google's image search,” *Proc. 10th IEEE International Conference on Computer Vision (ICCV 2005)*, pp. 1816-1823, 2005.

[93] “US Secratary of States Photo Gallery,” Available: http://www.state.gov/r/pa/ei/pix/c27657.htm, 2013.

[94] “Council of the European Union Photo Gallery,” Available: http://www.consilium.europa.eu/council/photographic-library?lang=en, 2013.

[95] A. Vedaldi, V. Gulshan, M. Varma and A. Zisserman, “Multiple kernels for object detection,” *Proc. 12th IEEE International Conference on Computer Vision (ICCV 2009)*, pp. 606-613, 2009.