Semantic Face Retrieval

By

Karthik Sridharan

July, 2006

A thesis submitted to the
Faculty of the Graduate School of
the State University of New York at Buffalo
in partial fulfillment of the requirements for the
degree of

Master of Science

Department of Computer Science and Engineering

Acknowledgements

I wish to thank Dr. Govindaraju for his continuing support and guidance, extending me the greatest freedom in deciding the direction and scope of my research. It has been both a privilege and a rewarding experience working with him for the past two years. I would also like to thank my other committee member, Dr. Matthew Beal for his valuable guidance and readiness to help anytime. I would also like to thank my family for their continuous encouragement and support. My special thanks goes to Sharat Chikkerur with whose valuable guidance aided me in choosing the problem to work on for my thesis. I also thank all student members of the CUBS research group for engaging me in productive discussions and debate. In particular I extend my thanks to Sankalp Nayak, Sergey Tulyakov, Faisal Farooq, and Shamalee Deshpande among others. I would also like to extend a special thanks to all the volunteers whose help solely made my experiments possible.

Contents

List of Figures

List of Tables

Abstract

The description of a face given by people is almost always semantic in nature using verbal terms such as "long face", "thick lipped" or "blonde haired". Existing Face Retrieval systems are mostly image based and hence do not capture these semantic features directly. Further since they perform image matching or other computationally intensive processes for retrieval of faces they are generally inefficient even on relatively small databases of face images. We propose a probabilistic and interactive semantic face retrieval system that retrieves face images based on verbal descriptions given by users. The system is also capable of prompting the user to provide information about facial features of the targeted face that best distinguishes the person from the top choices dynamically at any given time in the query session. The proposed system can supplement systems like Identikit (or other mug-shot retrieval systems) by providing an effective and efficient filter. Since the semantic query process is essentially a table search; the system is efficient and can operate in real-time.

The proposed system automates the process of tagging face images with semantic labels which are then used for retrieving images that best fit the verbal description provided by the user. During enrollment, a critical step that needs to be performed for the automated semantic tagging of the face is facial feature localization and parameterization. We present two approaches for the task. First, in cases where mug-shot images are enrolled, primitive vision techniques and heuristics are used to locate and parameterize the facial features. Alternatively, in cases where unconstrained frontal face images are enrolled, a hybrid linear graphical model is proposed to model the relationship between the facial features and their locations based on a training set. This graphical model is then used to locate and parameterize facial feature extraction in a given face image. During the retrieval process Bayesian Inference is used. The system is interactive

and prompts the user at each stage of the query process to provide the description of a facial feature that is most discriminative. In our experiments the target face image appeared 77.6% of the time within the top 5 and 90.4% of the time within the top 10 retrieved face images.

Chapter 1

Introduction

One of the first visual patterns an infant learns to recognize is the face. Face is a natural means by which people recognize each other. For this reason and several other reasons, face recognition and modeling have been a problem of prime interest in the fields of Computer Vision, biometrics, pattern recognition and machine learning for decades. As a biometric Face has been very successful due to its unobtrusive nature and ease of use. It is suited for both overt and covert biometric applications. While biometrics like fingerprint and hand-geometry cannot be used covertly biometrics like gait often has very low accuracies. Face modality provides a middle ground where both high accuracy and covertness are achievable. Face recognition systems have high demand in airport and other public places for automated surveillance applications. Of late Facial expression analysis and detection are gaining significance in both law enforcement and human computer interaction applications.

Most of these above mentioned applications use face as a hard biometric for verification or identification of a person and mainly consist of the task of matching the actual image of a face with those stored in a database of face images. However, apart from their use as a hard biometric, the ”soft” traits of face modality are being used to group people instead of uniquely identifying

a person by his/her face. Face images have been used to identify ethnicity, gender and age of a person in [33, 64]. A more interesting application that views face as soft biometric is in the face retrieval systems.In many law enforcement applications, soft biometric [32] traits of the face have to be matched to retrieve the target face image from a dataset.

1.1 Semantic Face Retrieval

Many existing face retrieval systems require as input the image or sketch of the target face. Hence synthesis of the target face image must be first accomplished to initiate the retrieval process. However the most natural way in which people describe a face is by semantically describing facial features. Presently, not many automated systems exist which can directly take advantage of the verbal/semantic description of a face in searching face databases. Such a system helps in easily using the user descriptions of a face in retreiving a smaller subset of candidate face images from a large database of images.

Semantic Face Retrieval refers to retrieval of face images based on not the raw image content but the semantics of the facial features like description of the nose or chin of a person. For instance ”A round faced person with blonde hair and mustache” is a verbal semantic description of a face. It must be noted that there exist many mug-shot retrieval systems that retrieve face images based on users choice of similar faces from a pool of face images. While these systems do retrieve faces based on semantic descriptions, they do not directly deal with semantically describing the face or retrieving faces according to semantic contents. A classic example of semantic face retrieval system is to automatically build a sketch or synthetic image of the target face based on the semantic description of the face and then performing an image match of the composed image with those in the database.

However in such systems the retrieval process is time consuming. In intuitive approach that

is proposed here involves automatically tagging faces in the database with semantic descriptions and subsequently retrieving faces that match verbal queries such as ”blonde haired”, ”mustache”, and ”spectacles”. It should be noted here that mug-shot retrieval systems where a face image is first synthesized based on the user descriptions and subsequently used for image retrieval are often more accurate. However the drawback is the computationally intensive process of image matching against numerous images in the database. Our approach quickly narrows down the possible images from a large database of images based on matching verbal description of the face with the tagged description of the faces in the database. Thus the system can be used as a first step in a search process and all further searches can be performed on the smaller set of images retrieved by the system for obtaining more accurate results efficiently.

1.2 Proposed Approach

To allow the users to directly use verbal discriptions to query for the target face image, during enrollment we extract semantic attributes of the face like whether the face in the image has a mustache or not or the size of the nose etc. These semantic attributes of the faces are then stored in a meta database which is then matched against the verbal description of the face given by the user during retrieval process. An illustration of the proposed system is shown in Figure 1.1. The user can select a particular feature on which he wants to query and choose one of the descriptions of the feature from a list. For example, the user can select the feature ”mustache” and provide the description that the person has a mustache. For relative queries like ”lip thickness”, the user can view the top few retrieved images and with respect to these images can select a relative description such as ”normal”, ”thick” or ”thin” for the lip of the target person. Thus the user can base his query on the top few images retrieved at each step. The system also prompts the user to provide a description about the feature that is most discriminative among the subset of

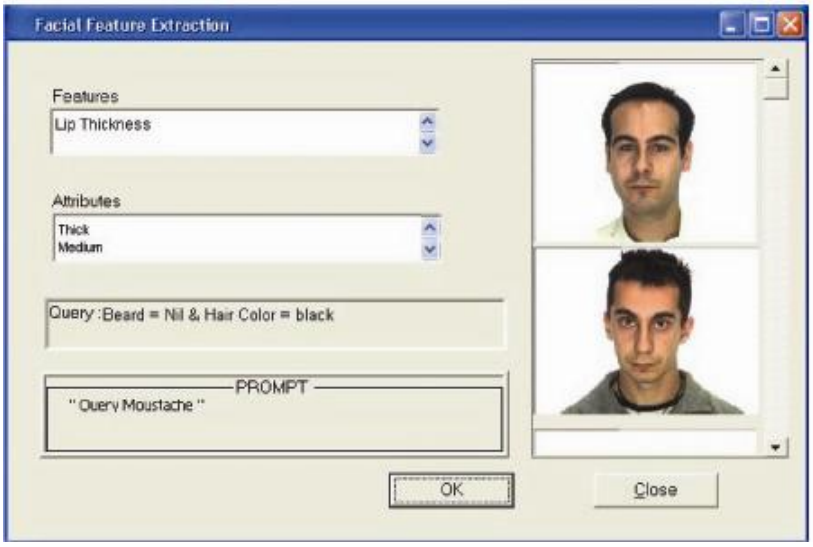

Figure 1.1. Snapshot of the System.

contenders at any given stage.

Our system has two operating modes (during enrollment stage), one is mug-shot image enrollment and the other is unconstrained frontal image enrollment. Given the application, one can decide which mode is to be used. For example, for applications where a witness is describing the suspect so that the police record of the suspect can be retrieved, mug-shot enrollment system is useful since all images in the police records are generally mug-shots against plain background. However for surveillance application queries are verbal and the target image is unconstrained. While in the first mode, primitive vision techniques and heuristics about the face can be used for the facial feature localization and automated semantic tagging process; advanced vision techniques are required for the unconstrained mode. In this thesis a novel hybrid linear graphical model is proposed for learning the model of the face. Hybrid Markov Chain Sampling is used to perform localization and parameterization of the facial features. Then using heuristics of the face it is automatically tagged with semantic lables. During retrieval Bayesian inference is used

to retrieve the faces in database that are most probable of being the target image. The system also picks the most discriminative attributes and prompts the user to query about that attribute for more efficient retrieval.

1.3 Applications

The key area of application of the proposed system is in Law Enforcement systems. An obvious application is to use the system in aiding witnesses to pick out suspects from a large database of face images stored in the police records by using verbal description of the suspects face. The description of the suspect given by the witness is generally verbal in nature. In many law enforcement agencies sketch artists who sketch the suspect's face image based on the verbal description provided by the witness. Even if the matching of this form of sketch with face images in a database of suspects is automated, to match a sketch with a huge set of face images would be time consuming. However the proposed approach matches the verbal descriptions of the suspect's face given by the witness directly with the semantic descriptions of faces in the database thus making retrieval fast. The proposed system provides a method by which we can narrow down the number of possible face images on which a detailed image search (or image match between sketch and the image in the database) needs to be performed. It is worth noting here that the face image fitting witness's description need not always be searched for in a database of face images but can also be searched for in surveillance videos.

An alternative application of the proposed system is in surveillance applications. One might want to search to see if any of the people who appeared on a surveillance video are present in a database of, say, wanted criminals. Obviously matching every person on the video with faces in the database is impractical. However the proposed system can be used in picking out possible people on the surveillance video and respectively the possible images from the database

that match the semantic description of these people's faces. In such an application the faces on the surveillance videos are semantically tagged and the semantic descriptions of the faces in the database and surveillance video are matched.

1.4 Thesis Outline

Chapter 2 presents background and related work in Face Retrieval systems and Chapter 3 provides a brief overview of the proposed system, Chapter 4 describes in detail the Enrollment sub-system which is responsible for tagging face images to be added into the query database. Chapter 5 discusses the Retrieval sub-system which is responsible for query response and prompting of the user. Chapter 6 provides performance analysis of the system and also provides an example query to show the effectiveness of the system. Chapter 7 discusses the key contributions of the work and delineates directions for future work.

Chapter 2

Related Work

Face recognition and identification of a particular person from a database of images by matching the face image of the required person with that stored in the database has been an application of interest for decades. Pentland and Turk [59] introduced the concept of eigenfaces which has ever since been popularized for the task of face recognition. Subsequently many methods for face recognition have been proposed which in a naive sense were the first completely automated face retrieval systems which retrieve face images by matching the instance of face image in the database with an other instance of the same face.

2.1 Introduction

In reality face retrieval systems that retrieve faces based on user descriptions have been around since the 1980s. Law enforcement agencies have been using sketch artists and Identikits [9, 37] for composing a rough sketch of the face of suspects which is then used for matching against the face images stored in the police record. Identikit consists of a set of transparencies of various facial features that can be combined to build up a picture of the person sought. Forensic artists transform the verbal description of the suspect given by the witness into rough sketch of the

suspects by putting together these transparencies to fit the description. Once the rough sketch is made, large image databases are searched manually to find faces that resemble the sketch. The process is iterative and laborious which is once of the main motivations for automating the process by building mug-shot retrieval systems and semantic face retrieval systems.

2.2 Content Based Face Retrieval

One solution to automate face image retrieval system is to extend the idea of the traditional content based image retrieval systems to face images. The users can query for faces by submitting face images or by simple keywords about the basic image qualities. Systems like QBIC (Query by image content) [22] and MIT Photobook [46] have been used for such applications.

The QBIC (Query by image content) is a classic content based image retrieval system which allows querying of images in the database by using simple query images and image properties like average color, color distribution, texture etc. The user can provide a rough sketch of what he/she is looking for and can also provide search key like a rough sketch of the histogram of the target image expected. The QBIC system which performs basic content based image retrieval can also be used for face images trying to retrieve the face images the user is looking for. However as such a system clearly cannot capture the semantic aspects of a face.

Photobook [45] is a content-based retrieval system that provides methods for searching several types of related image databases including faces. The key advantage of Photobook is that it uses methods like PCA (eigenfaces) as one of the key image feature for performing the retrieval. Since PCA based compression of the images is statistical in nature, to some extent the semantic aspects of face image is preserved by these features thus providing better retrieval for face images.

The Computer Aided Facial Image Identification, Retrieval and Inference System (CAFIIRIS) [67] for criminal identification stores and manages facial images and criminal records and pro-

vides the necessary image and text processing and editing tools. It uses a combination of feature based PCA coefficients, facial landmarks and text descriptions to construct index keys for each image. The advantage of this system is that it is interactive as it uses relevance feedback from the user to retrieve the relevant images.

2.3 Mug-Shot Retrieval Systems

In the above content based face retrieval systems while we have a general method for face image retrieval we still do not have a specific method by which we can generate the initial query image using which to perform the search. For instance if we need to retrieve faces that fit a particular semantic description like ”blonde with mustache ... ” the best we can do is to pick amongs a list of available images the face image that is closest to what we want and try to use that for query. Clearly this is not an efficient way to query when the user is specific about the description of the face he/she is looking for. To overcome this issue two solutions are commonly available. The first is a mug-shot retireval system where at any given time the user is given a set of face images amongst which he/she chooses the most similar one to the target. The system based on the selection synthesizes images that include more of the features of the selected image. Thus as the user picks out images the face image synthesized evolves and would hopefully give a good representation of the target face to be retrieved. Once the syntesized face is close enough to wanted target, the synthesized image is matched against the images in the database to finally retrieve the target image. Evofit [23] is one such system which performs face retrieval by evolving the required face based on user feedback from faces present in the database. FacePrints [5] and PROfitis another such system which again used genetic algorithm to evolve the required synthesized face.

The other solution is similar to Identikit where the user picks out specific features like the

nose, eyes etc. of the person he/she is looking for and the system synthesizes a face composite image that combines these individual features and this synthesized composite image is matched against the ones in the database to perform the retrieval. The Phantoma [1] is an automated facial database search system which uses Elastic Graph Matching to recognize faces. The system takes as input the composed face images and retrieves images from the database in the order of their similarity to the composed picture. Systems like E-fit [8], PRO-fit [23] and Photofit [43] can be used for combining various feature selections of eyes, nose etc. to form a composite face which is then used by systems like Phantomas [1] to perform retrieval. [4] proposes another mug-shot retrieval system that uses eigenface approach and composite face image synthesis for retrieval of mug-shot images. A combination of composite face and face evolution is also commonly used in practice. For instance in the use of EVO-fit often the first set of images to start with are syntesized using PRO-fit.

2.4 Semantic Face Recognition

While mug-shot retrieval systems capture to some extent the semantic aspect of the face by allowing users to choose or compose their faces based on the semantic aspect of the faces, these systems do not by themselves directly extract or use the semantic aspects of the face. For instance these systems do not try to infer if the person in the image has a mustache or not directly. [30] proposes a face recognition system which instead of storing the actual face image in the enrollment dataset extracts the semantic description of the enrolled face images. In the system the semantic descriptions of the enrolled face images are organized as a semantic face graph derived from a 3D face model containing facial components in the spatial domain. The semantic face graph visually look something like a simple line sketch of the face image which basically captures only the semantic aspects of the face like general shape of eyes and nose. Aligned

facial components are then transformed to a feature space spanned by Fourier descriptors of facial components for face matching. The semantic face graph allows face matching based on selected facial components. The system proves that the simple semantic features of the face are strong enough to even perform the task of face recognition.

2.5 Discussion

The basic content based image retrieval systems used for the special task of face image retrieval especially when the query is some form of user description is not very effective since it does not capture any specific semantic aspects of the face. While systems like Photobook [45] use eigenfaces for features which preserve some semantic ascpects of the face they do not capture the specifications of the face given by the users. Mug-shot retrieval systems that evolve the synthetic face created using user feedback address the problem of using the user specifications directly for face retrieval. However often users may have specific details like the kind of nose or whether the user has a mustache and the process of face image sythesis by evolution of the face based on feedback may take a while to actually produce the kind of face image the user is looking for. Further a potential problem with the system is that a user can keep on cycling through the same set of images if the genetic algorithm gets stuck in local maxima. Also, since the eigenfaces are generally used whose parameters are learnt from a specific training set, it would be difficult synthesize the target face if the desired face is quite different from those present in the training set. Composite face synthesis for retrieval purpose definitely addresses the problems. However the problems associated with synthetic face composition is making the synthesized images look real with no artifacts like false edges where the different facial components are joint.

The main drawback in most of the mug-shot retrieval systems is the fact that they do not directly extract any semantic features of the faces in the database. While composite face synthesis

allow users to choose specific kinds of various facial features to construct the face that captures the semantic aspects of the target face, the matching process for the actual retrieval does not match the semantic discription but only the entire face image. The semantic face matching system [30] addresses this issue since it extracts semantic features from the images in the database. Another drawback of the methods discussed here including the semantic face matching system is the fact that user description is often limited to verbal descriptions of the face. Simple verbal descriptions of a person like “thick lipped” or “blonde haired” help in narrowing down the candidate faces efficiently. Further, the enrollment or data population stage is generally not constrained by time but during the query stage, we need the immediate results. Our system extracts simple verbal descriptions of the face and saves them in a meta database, thus speeding up query retrieval process significantly [54].

Chapter 3

System Overview

The proposed system can be divided into the Enrollment sub-system and the Retrieval sub-system. The Enrollment sub-system accepts as input images that contain frontal view of a face. It in turn outputs semantic descriptions (consisting of 14 attributes given in Table 4.1) of the face such as whether the person is wearing spectacles, whether the person has a long nose etc. Thus the semantic tagging of the images in the database is all done in the enrollment phase. The Retrieval sub-system accepts the verbal descriptions of a person's face given by the user (or by running the semantic tagging on another face image) and retrieves images from the database by matching the descriptions given in the query probabilistically with the semantic description entries of the faces in the database. The retrieval sub-system also prompts the user at each stage on which feature to query about next by selecting the most discriminative feature amongst the remaining ones. The block diagram of the proposed system is shown in Figure 3.1. Though both the enrollment sub-system and retrieval sub-system are shown together in the block diagram, in reality the enrollment subsystem is used only in enrollment phase where the semantic description of the faces in the image to be enrolled is extracted and stored in the meta-database. The retrieval subsystem is in action only during query retrieval and the only way in which both these sub-

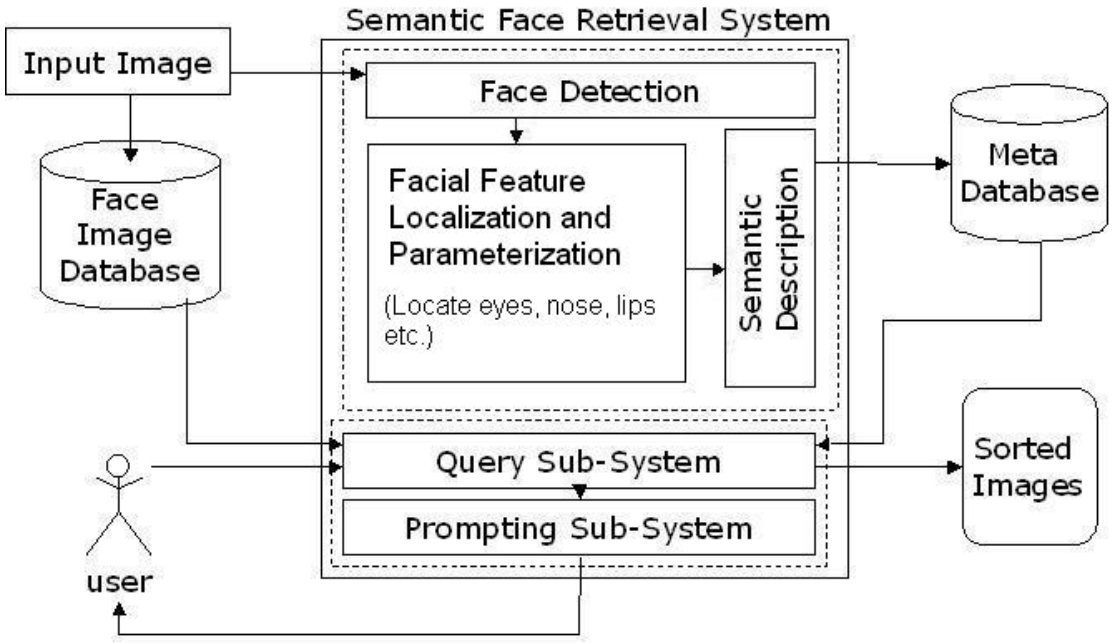

Figure 3.1. System Overview

systems are linked is via the meta database. This independence of the sub-system gives us the freedom to replace or experiment with one of the sub-systems while the other need not be altered.

The enrollment sub-system's main task is to automatically tag face images with semantic labels. The first step of the subsystem is face detection where given an image to be enrolled which contains one or more faces in it, the task is to extract all faces in the image. The faces are detected and marked in the image as rectangular regions in the image consisting of the face. These rectangular regions consisting of the faces are then extracted as separate images and are all scaled to a fixed size. The next step in extracting the semantic descriptions of the face is localization and parameterization of facial features. This step consists of finding the locations of eyes, nose and lips (eyebrows in case of unconstrained image enrollment). The parameterization of these facial features are performed by bounding them with simple polygons that describe them (like triangle for nose and rectangle for lips). The parameters of the bounding polygons are used as the parameters describing the features. For instance the height of the rectangle bounding the

lips represent lip thickness and the width of the rectangle the width of the lips. The final step is the semantic tagging itself. Based on the basic facial features located and heuristics about the face, the features that help in semantic description are approximately located and detected and hence the face is semantically tagged. For instance, by knowing the locations of nose and lips the approximate location of mustache can be found and based on simple vision techniques we can label the face as having or lacking a mustache. Thus the face is tagged with semantic lables.

The Retrieval sub-system accepts as input from the user the semantic description of the face of the person and retrieves the images in the order of how well they fit the description. The user selects the feature and the appropriate verbal description (keyword) describing that feature. The feature then converts the description into the same format as that stored in the meta database. For instance for binary attributes like whether the user is wearing spectacles or not, the users answer is converted to 0/1 for yes/no. For the continuous attributes like nose size, if the user selects “big”, the system converts the description to a specific numeric value that represents the concept “big” for nose. This converted description of the user is then directly compared with the values in the meta-database to perform retrieval. The rerieval process is probabilistic in that the system uses Bayesian inference to returns n (in our case 15) images to the user which are most probable of being the target images according to the description given by the user so far. Initially of course all images are equally probable but as the user describes the various features of the face the system calculates the posterior probability of the images in the database given the verbal description of the user so far. An interesting thing to note here is that the user can use the n images he/she sees to give relative descriptions of the face. For instance if the target face the user has in mind has a smaller nose than the top retrieval he sees, then even if the user feels that the target face’s nose is not small in general, in a relative context the description of the nose as small would work. At each stage, the sub-system also finds amongst the non-

queried features the feature that is most discriminatory prompts that to the user so that he/she can provide description about that feature next to retrieve the target face efficiently. The prompting sub-system is expecially useful when the user is not sure about which feature to query about next.

Chapter 4

Enrollment Sub-System

It is the responsibility of the enrollment sub-system to create the database of semantic descriptions of the faces by automatically tagging them.

4.1 Face Detection

The first step in the process of image enrollment is the detection of face to be enrolled. In our system we use the classic method of skin color based detection combined with blob analysis for detecting the face in the image. It must be noted that this method is not very effective in complex background with skin color patches present in backgound. Hence we only consider images relatively clean background where the method works effectively.

4.1.1 Related Work

Face detection has been a problem of great interest in the vision community for a long time. Existing face detection methods can be mainly divided into two categories; image based and feature based. Image based methods essentially treat the whole face as a single pattern and trys to perform face detection by finding regions in an image that have a pattern of face in

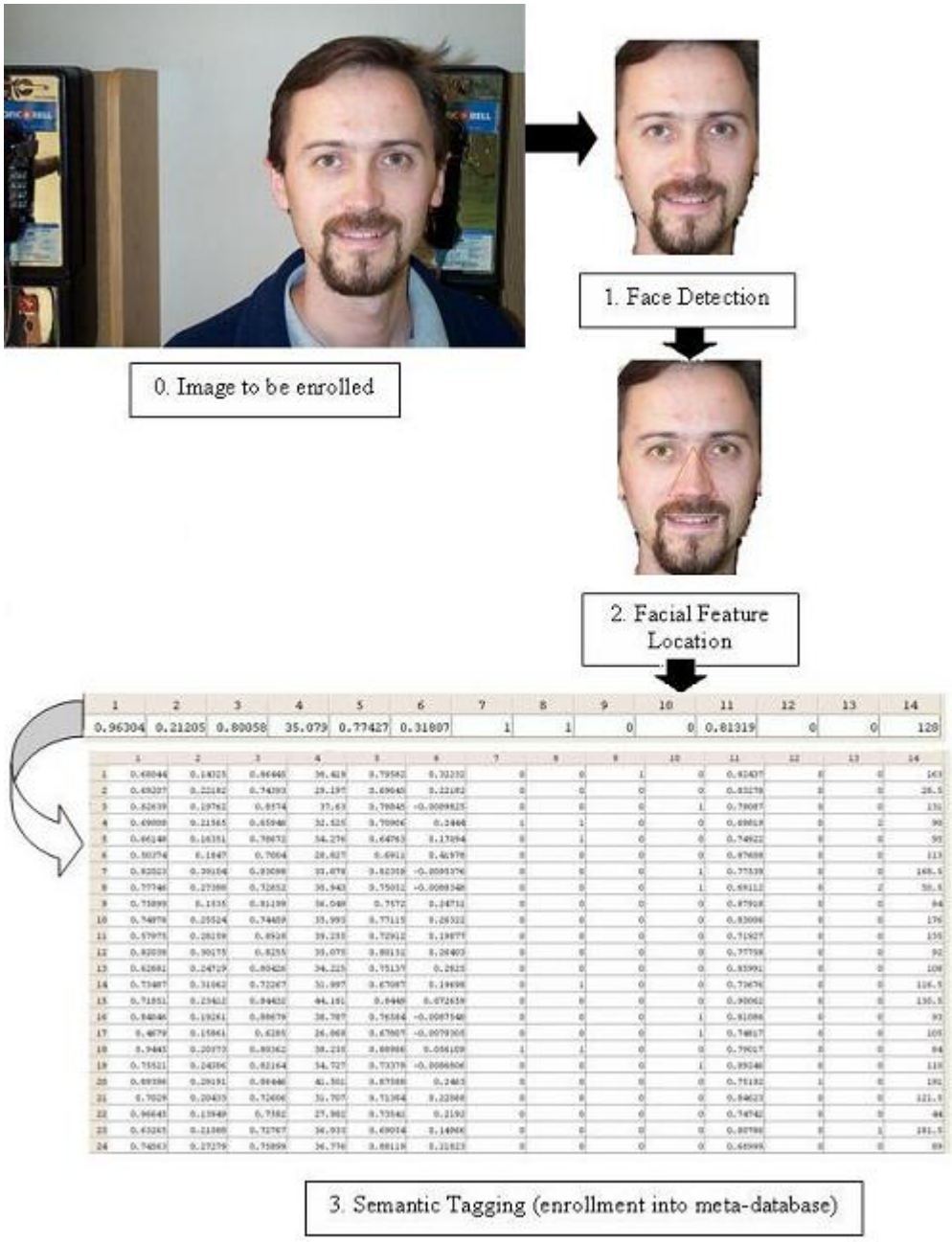

Figure 4.1. Example of Enrollment Process

them and labling them as faces. In [59] PCA (eigenface) based method for face detection is proposed where the various regions of the image are scanned and labled as face or non-face by using eigenface projection to determine whether the region is a face or not. In [10, 50] a back-propogation neural network for classification of patterns as faces/non-faces is proposed for the task of face detection and recognition which can handle slight rotations of the faces in the image. In [42] face detection and recognition using SVM's for classification is proposed. While image based approaches are more robust and can handle even gray scale images and complex backgrounds better they often are not very efficient. In any image based method we need to scan the entire image at different scales classifying the various parts of the image as face/non-face.

In a few feature-based approaches this problem can be avoided. Feature based approaches can be based on high level features like seperately detecting eyes nose etc. or can rely on low level features like color or edge feature analysis. One important property of human face is that even when the complextion of a person varies the basic skin color composition (proportion of Red, Green and Blue colors) falls in a fairly small range. Methods like [13, 27, 31, 36, 39, 48, 61, 69] have made use of this property along with basic heuristics of face to perform face detection. While most of these methods use skin nolor based detection they also combine cues from motion or blob analysis like basic density and shape of blob to decide if the region is a face or not. The advantage in these methods is that we avoid scanning through entire images at different scales trying to make decision each time. Rather we only extract the regions most probable of being faces and try to verify if they are indeed faces or not.

4.1.2 Proposed Approach

Since we consider color images to start with, a color based detection of face along with blob analysis was used for face detection. More complicated and superior methods for face detection

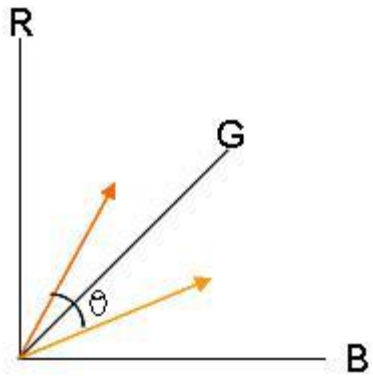

Figure 4.2. RGB Vector Angle Based Detection

can be used. However for our experiments we used the relatively simpler methods since all the images considered are frontal or almost frontal images and therefore skin color based face detection worked adequately. Each pixel in the image can be viewed as a vector in the 3D RGB color space [62]. Skin color based region segmentation is done by thresholding the angle (θ) between the mean skin color vector and unit color vectors per pixel (see Figure 4.2). Since the intensity holds no color information, the thresholding angle helps in robustness against varying intensities. Skin color blobs, of right area and concentration are detected as face regions and extracted by a rectangular box. Figure 4.3 shows an example of the face detection process.

4.2 Facial Feature Localization and Parameterization

The next step which consists of locating basic facial features is crucial. Once the basic facial features are located based on the information the semantic features like hair color or thickness of eyebrow etc. can be determined using simple heuristic rules and primitive vision techniques. Since the process of enrollment is offline we are not as constrained on time as we would be during query. Hence we can actually use advanced techniques and extract exact contours of the facial features. However this is not required for the application and Hence to parameterize these

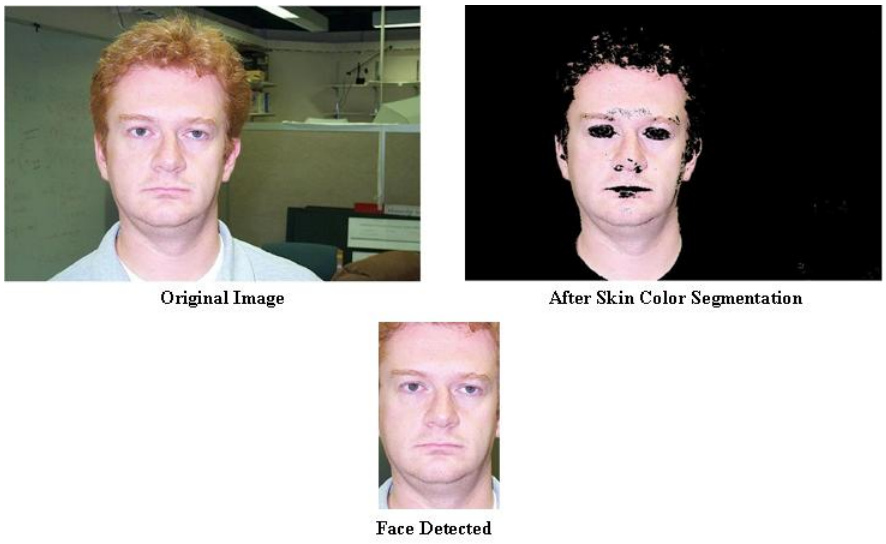

Figure 4.3. Example : Face Detection

features, we fit the lowest order polygon (bounding the feature) that can effectively describe them. For example, fitting a triangle for the nose can describe its length width and size. Further, in the location and parameterization of facial features, based on the kind of image enrolled we have two methods. The first is when mugshot images are involved, based on simple vision techniques the lips, eyes and nose are detected and parameterized. When we have unrestricted frontal face images, we use a hybrid graphical model to locate and parameterize the eyes, nose, lips and left eyebrow. The left eyebrow is also located to increase the precision of semantic tagging in unconstrained images where noise due to illumination effects are involved.

4.2.1 Related Work

Various algorithms and face models have been proposed to locate/parameterize facial features. Just like in face detection these methods can be image based or feature based. Feature based approaches rely on either lower level features like color of facial feature, grey level, histogram etc or on higher level features like shape of the feature using active shape modelling or edge

analysis. Image based models directly try to model the facial feature as a pattern and try to locate the particular pattern in the face image. Image based methods are more robust to noise in images but are often not as efficient as feature based methods. The choice of method for facial feature localization and parameterization generally depends on the quality of image we expect. When the quality and resolution of the image is relatively high feature based methods are often enough.

Most feature based facial feature localization methods combine color based segmentation and edge analysis or blob analysis to locate facial features. The fact that one of the prime methods of face detection is by skin color segmentation and facial features like eyes and lips are not skin color is often used to aid in effective facial feature localization. For instance in [52, 26, 60] facial feature localization is performed by color based segmentation and blob analysis. In [26], the mere fact that eyes and lips have different color composition than face along with connected component analysis is used to detect facial features. In [52] a similar method with the use of morphological operators to remove non-facial features is proposed, In [60] a similar method to the one proposed in our system for mug-shot image enrollment is proposed. In [60] once face is detected, eye position is estimated by finding eye shaped and eye sized region in areas where there is sharp change in red channel. Then lips are located by usinf lip color segmentation and blob analysis. This is followed by nose localization and parameterization by template matching. In systems like [49] the natural geometry of face like proportion of distance between eyes and lips and distance between node and eyes are used along with feature based methods for facial feature localization.

Though feature based methods are easy to implement and effective in clean images, in images affected by bad illumination of variations in pose and other such effects feature based methods fail. In such a case image based approach is prefered. Model based feature localization and

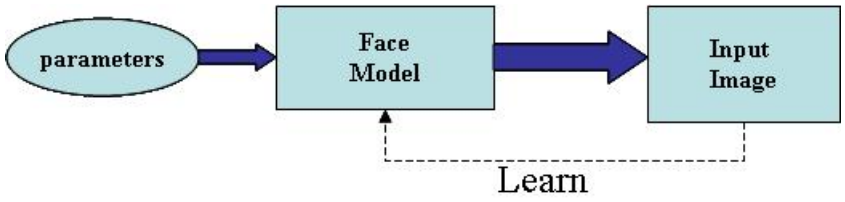

Figure 4.4. Model Based Methods

parameterization is a classic method is image based approach where the pattern (facial feature) we are looking for is identified by building a model for it, learning parameters for the model using training examples and using this model to decide if a test pattern is valid feature or not. Figure 4.4 shows the basic concept of model based approaches. Eigen feature based approaches belong to the category of model based approach where the model is a simple linear Gaussian one [44]. Model based approaches are not confined to only Image (appearance) based models but can also model features like shape. For instance in [12] Active Shape Models are proposed that model the shape details of the face. In [20, 18] Both shape and appearance details are modeled in a combined fashion using PCA. When the models used are probabilistic they can often be modeled as graphical models. The advantage of viewing the model as graphical models is that we can easily represent the relation between the various facial features by adding corresponding edges in the graphical model. Figure 4.5 shows an illustrated example of such methods. [55] uses such graphical models for the very same purpose of facial feature localization and modelling.

4.2.2 Proposed Approach for Mugshot images

The localization and parameterization of facial features for mugshot is carried out in a serial fashion. First, the lip is detected and parameterized using color based segmentation and histogram based segmentation. Next based on the lip location, approximate location of eye is

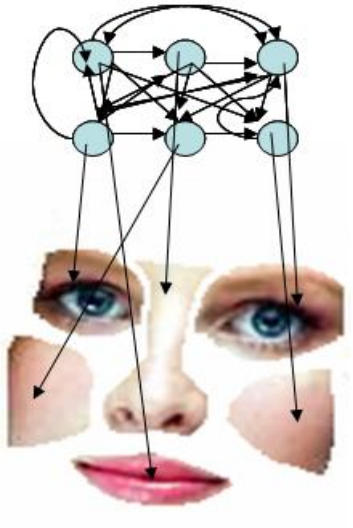

Figure 4.5. Graphical Model Based Facial Feature Modelling

considered. A Bayesian Hough transform based circle detection is used to locate eye centers. Next based on lip and eye location the approximate location of nose is extracted and based on color vector based edge detection the nose is parameterized by a triangle bounding it. This method is serial in nature and hence errors in detection of lip get propagated to the location of eyes and nose and errors in locating the eyes get propagated while finding nose. However since mugshot images are relatively clean this method is adequate. Figure 4.6 shows a mug-shot image in which the eyes, nose and lips are located and parameterized.

Lip Localization and Parameterization

From [62] we see that efficient lip detection can be done by segmenting image based on lip color on the face image region. Since lip color is distinct in the face region, we can effectively perform lip detection based on lip color. However the problem with this method is that the lip corners are not captured as they have very low intensity. We use histogram based object segmentation [51] method to overcome this problem. Since we do face detection based on skin color, once we

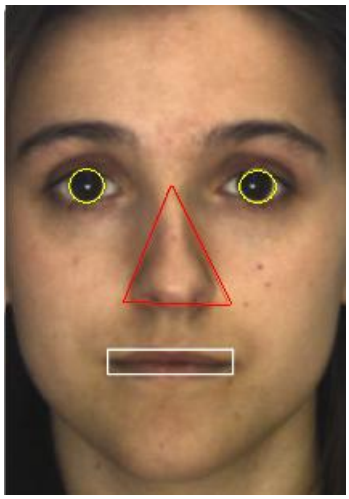

Figure 4.6. Example : Mug-shot Facial Feature Localization

Figure 4.7. Lip Localization

locate the face we subtract the skin color regions in the image. Thus we are left with the image of the face with only facial features like lips, eyes, eyebrows etc. Now by lip color detection, we find the approximate location of the lips. Next we find the histogram of this region of the image to perform segmentation. To segment the image, the histogram is first smoothed. Next the points at which the slope of histogram changes sign is termed as valleys and these points are used as threshold to segment the image. Once the quantization of the lip region of the image is performed, based on the quantized image the lip width is altered to fit the lip well. Figure 4.7, shows the initial color based detection results the histogram based segmented image and final detection result.

Eye Localization and Parameterization

Once lips are located we can reduce the area in which we are to search for eyes. We use the detail that the eyes are above the lips and hence we search for eyes only in a rectangular strip above the lips. We use circle detection based on Hough transform as proposed in [6] for eye localization. However, to use Hough transform for circle detection, we need to know the radius of the circle to be searched. In the database used, the radius of pupil of all the eyes varied from 9 to 13 pixels and the mean radius was 11 pixels. Using the set of 5 radii for each point in the image we get a set of 5 accumulator values. Usually to find out the location of the eyes, the point that has maximum sum of accumulator values for all the radii is chosen as the center for the pupil. However we know that in the database most pupils have radius of 11. Hence to improve the accuracy of eye detection, we use a probabilistic model to locate the most probable eye center. We convert the accumulator scores to probability of the point being the center of circle of radius r_j by dividing it by $2\pi r_j$, since the accumulator can have a maximum value of $2\pi r_i$ the circumference representing a complete circle. Now we need to find out for each point $point_i$, the marginal probability of it being the center of pupil. This is calculated as

$$P(point_i) = \sum_{j=1}^{5} (P(point_i|r_j)P(r_j))$$

where $P(point_i|r_j) = \frac{Accumulator(point_i, r_j)}{2\pi r_j}$ and $r_{1..5} = [9..13]$. The priors, that is the probability of a particular eye having a particular radius was set by trial and error as $P(r_1..r_5) = [0.1, 0.2, 0.4, 0.2, 0.1]$. Finally the point in the image with maximum marginal probability $P(point_i)$ was set as the center for the pupil. The process is done for both left and right eyes separately to locate the two eye centers.

Nose Localization and Parameterization

Once the eye centers and lip center triangle is formed we can use face proportions to locate all other features based on knowledge about the face that we have. For instance, from anthropological studies we know that distance between midpoint between the eyes and nose is about two-third of the distance between eyes and lips. Thus we get the approximate line of nose-ending. Then we use color based edge detection to trace out the curves of the nostrils and using these we fit a triangle for the nose. The color-based edge detection is performed by assuming that each pixel is a vector in 3D RGB space. Thus by thresholding angle between the color vectors of two pixels, using a sobel mask we detect edges [17]. By using color based edge detection, we can select threshold to detect even soft edges like the ones formed by nose. As nose edges are formed by transition of pixels from skin color to skin color, we can only detect edges that transition from skin color to skin color.

4.2.3 Proposed Approach for Unconstrained Face Image

Unlike mugshot images, error propogation in unconstrained images during facial feature location can be fatal. Hence we need a more advanced method that is robust against noise in the image. Facial feature location and extraction is like any object detection and extraction problem. Probabilistic modeling of objects for object detection has shown tremendous success in recent times due to their ability to deal with noise in image. The use of Probabilistic PCA [58] to model objects in images using a low dimensional latent variables has shown promising results. Indeed, the PPCA has been used for object detection in given image. [14] proposes a Bayesian approach for object localization in images. The method tries to model the object using the underlying distribution of both the latent variables and the observation models. To locate the object in the

image, the image is panned and probability at each position of the object being present is calculated. However this method does not take into account the scale variations in the object and further is not very efficient as it needs to calculate the probability at every point. [56] proposes the use of Markov random fields to locate objects. The method is efficient due to the fact that each pixel and its neighborhood is modeled using Markov random field and hence if a part of the object falls within the boundary of search, the method quickly closes in and locates the entire object. However the method is still computationally expensive as each pixel is modeled using Markov random field.

There have been numerous facial feature extraction methods that search for facial features in the given face image independently. However these methods are slow and can even return facial feature positions that are absurd. A few works like [57] apply heuristics to make the search more efficient. One main advantage of facial features is that there are correlations between the image, position and sizes of the different facial features. That is, given the position of eyes and lips we can infer the position of nose similar correlations exist in the image and size of the facial features. [29] uses a hybrid graphical model similar to ours to model objects and the interactions between them. Indeed modeling of the relationship among facial features and their position has been done in [55] where a directed graphical model is used to model the relationships between the facial features. However this directed relationship between facial features induces a false causality amongst facial features while there is no reason to believe that the facial features can be associated by any particular directed association.

We propose a hybrid linear graphical model [53] that captures the relationships between facial features by undirected connections between latent variables of the features. The proposed model is a linear Gaussian model. The advantage of the method is that the features are located simultaneously and locating one feature aids in locating the remaining features. Further unlike

the method used for mugshot images the methods does not propagate errors.

Method Overview

The system implemented by us tries to locate and extract eyes, nose, lips and left eyebrow from the given face image. Only the left eyebrow is considered as using the position and size of the eyes and the left eyebrow we can more or less extract the right eyebrow. To actually extract facial features we try to bound the features by polygons and hence the image in the bounding polygon becomes the extracted feature. For instance, we use rectangles for eyes, lips and eyebrow and isosceles triangle for the nose. Now given an image our task is to find the best position and parameters for the bounding polygon such that the facial feature image is captured in the polygon. Each of the bounding polygons used has 4 parameters; the x co-ordinate, the y co-ordinate, the height and the width of the rectangles and isosceles triangle. While the x and y co-ordinates capture position of feature, the height and width parameter capture the size of the facial feature. Now for any given instance of these parameters, we can in turn extract some image from the given image.

In the proposed method we try to use a probabilistic model such that any instance of these parameters has a corresponding probability according to that model in a given image. Further, it is obvious that the parameters are correlated. Given the position and size of the eyes and nose we can infer details about probable position and size of the lips. There exist correlations in the image of facial features in the image too. For instance if we locate the nose and it turns out to be an image with low intensity then with high probability the image of the lips in the face must also be of low intensity. To capture these probabilistic relations we use a Gaussian graphical model. The hybrid graphical model shown in Figure 4.8 is used to model the facial feature. The model is a generative one where, the latent variables generate the facial feature image , the correspond-

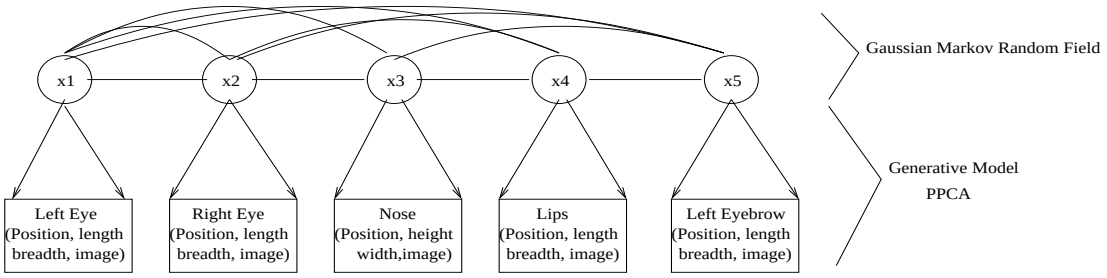

Figure 4.8. Hybrid graphical model used to model facial features

ing position and size of the feature. Since the latent variables are all connected by undirected edges to all other latent variable nodes, the correlations between the positions, sizes and images of different facial features is taken care of. We use PPCA (Factor Analyzer with spherical noise) to model the observed variables from the latent variables. Further, the undirected completely connected graph among the latent variables is modeled as a Gaussian Markov random field (autonormal model) [11, 7]. Since the entire hybrid graphical model is a linear gaussain model exact belief propagation is possible. Once the parameters for the graph are estimated based on the training data, we use a Hamiltonian sampling based searching method to efficiently find the facial features in the image. The method assumes that face detection is done and hence we initialize the search from positions in the approximate region where the facial features would be found. Now to locate a feature, we assume the remaining features are fixed to their current values and sample the position and size of the required feature conditioned on the remaining features' latent variables.

Modeling the Face

The hybrid graphical model used to model the face can be divided into two parts. The first is the directed part which relates the position size and image of each feature with the corresponding

latent variable. The second part is the completely connected undirected graph between the latent variables.

Generative (directed) part of the graph For each of the facial feature we use a linear Gaussian model (Factor analyzer) to model the observed variables (image of the feature, position of the bounding polygon and measurements of the size of the polygon) using the latent variable as,

$$y = Wx + \mu + \epsilon \tag{4.1}$$

where y is the observed variable, x is the latent variable distributed as a Gaussian $N(0, I)$ and ϵ is the spherical Gaussian noise distributed as $N(0, \sigma^2 I)$. μ is the mean of the observation given by

$$\mu = \frac{1}{N} \sum_{i=1}^{N} y_i \tag{4.2}$$

It can be shown that in case of spherical noise the factor loading matrix that maximizes the likelihood W_{ML} is the same as the set of principal components and is just the m eigen vectors of the covariance matrix of observations Σ with maximum eigen values, where m is the dimensionality of the latent variable [3].

Further, we can find the particular σ^2 that maximizes the likelihood σ^2_{ML}. This is known as Probabilistic Principal Component Analysis [58]. The σ^2 that maximizes the likelihood turns out to be the average of the remaining N-m eigen values given by,

$$\sigma^2_{ML} = \frac{1}{N-m} \sum_{j=m+1}^{N} \lambda_j \tag{4.3}$$

where λ_j is the j^{th} largest eigen value.

When we are given an observation y_n, we can now find the likelihood of the observation under the PPCA model estimated. This is given by the Gaussian

$$y_n \sim N(\mu, WW^T + \sigma^2 I)$$

Further, given the observed variable y_n we can also find the lower dimensional latent variable x_n that maximizes the posterior probability of x_n as the mean of the posterior

$$\langle x_n | y_n \rangle = M^{-1} W_{ML}^T (y_n - \mu)$$

where $M = W_{ML}^T W_{ML} + \sigma^2 I$.

Undirected Part of the Graph To model the relationships between the facial features, we use a completely connected undirected graph. The graph is undirected as we cannot assume any causality as to which node causes which one. Further the graph is completely connected so that we can capture any relationship between facial features.

Markov Random Fields have been used to model undirected graphical models successfully especially in vision application. Markov random fields generalize undirected graphs with real valued variables. By the Hammersley-Clifford theorem [7] for Markov random fields any Markov random field is a Gibbs Random Field with joint probability of the form

$$P(x) = Z^{-1} e^{-U(x)} \tag{4.4}$$

where

$$Z = \sum_{x \in X} e^{-U(x)}$$

We use Gaussian distribution to model each latent variable conditioned on the remaining variables. Thus the joint distribution of the Gaussain Random Field is also given by a Gaussian

$N(\mu_{lat}, B^{-1})$ where B is the precision matrix. Since each latent variable is modeled by PPCA as a Gaussian (marginal distribution) , the assumption that the joint distribution of the latent variables is Gaussian is a valid one. Hence, the single site and pairwaise potentials of the multivariate auto-normal model is given by

$$V_1(x_i) = \frac{1}{2}(x_i - \mu_i)^T B_{i,i}(x_i - \mu_i)$$

and

$$V_2(x_i, x_j) = \frac{1}{2}(x_i - \mu_i)^T B_{i,j}(x_j - \mu_j)$$

such that

$$U(x) = \sum_{i \in S} V_1(x_i) + \sum_{i \in S} \sum_{j \in S} V_2(x_i, x_j)$$

($B_{i,j}$ is in turn a matrix which is the block corresponding to the ith and jth vectors in the precision matrix B).

During the training stage, after evaluating the latent variables for each training instance using PPCA we evaluate the Precision matrix for the joint distribution of the latent variables.

When we are presented with a new set of observations, we first evaluate the probability of observation according to the PPCA model and also evaluate the latent variables. Then we can evaluate the setting of latent variables by calculating the joint probability of the Gaussian Markov random field. Further conditional density of latent variable x_i given the remaining variables is given by normal distribution $N(\mu_i + \sum_{j \in N_i} B_{i,j}(x_j - \mu_j), B_{i,i}^{-1})$ where σ_i^2 is the spherical noise calculated for use in PPCA for the ith latent variable.

Training Dataset In the above model we can creat a training set by manually locating and parameterizing facial features for a subset and using this for training. In system trained so, if for instance the eye is located and parameterized correctly it then causes the polygons for other

features like the nose and lips to move towards more probable locations. However imagine an instance where the rectangle for left eye infact has half the left eye in the top half of the rectangle. In this system since our training set only had full left eye, the system cannot by itself know that it must move the polygon for left eye towards the top of the image. In such a situation the only factor that allows us to locate the eye is the randomness in the search algorithm. However we would prefer a more informed search since we could have all the data we want during training (half eye, full eye etc.). For this purpose we propose a simple method by which the problem is alleviated. Essentially instead of using training set as just the exact manual parameterization of the facial fearures per image we use the manual parameterization (ie. the x,y co-ordinates of polygon and its scale as lenth and height) as a mean for a Gaussian with a set variance and draw a few samples from the distribution. Now the training set is all these sampled parameters with their corresponding feature image (ie. Image bounded by the polygons). In this way when the graphical model sees half an eye it can use gradient descent to move in the right direction and capture the whole eye.

Locating Facial Features

In the model, the probability of observation for a particular feature y_i and its corresponding latent variable x_i is given by

$$P(y_i, x_i | x_j, y_j \forall j \in N_i)$$

$$= P(y_i, x_i | x_j \forall j \in N_i)$$

therefore

$$P(y_i, x_i | x_j \forall j \in N_i) = P(y_i | x_i) P(x_i | x_j \forall j \in N_i) \tag{4.5}$$

One possible way of facial feature extraction is to try out all possible parameters for the bounding boxes of facial features and choose the set of parameters with maximum joint probability. However this would be prohibitive due to the infinite number of possible settings of all the parameters together. We could instead use an iterative algorithm where we fix the parameters for the bounding polygon and hence the observation for all but one feature and vary only the parameters of that feature around its original position. If we find a new set of parameters for the feature with a higher conditional probability of the observed and corresponding latent variable then we set the parameters for the feature to the new value and repeat the procedure for the next feature keeping the parameters for all other features fixed. If we cycle around updating parameters in this fashion, we would end up locating and extracting features faster.

However this procedure too would take a long time since it would more or less be performing a blind search. We can make the search more efficient by using a search technique based on Hamiltonian sampling [41, 19] to sample the parameters for the bounding polygon for each feature from the joint distribution of observed and latent variable for that parameter Equation 4.5 keeping parameters and hence observations for all other features fixed. Note here that since we are sampling parameters for each feature keeping others fixed, and doing this iteratively, we are actually performing Gibbs sampling [24] on the joint set of parameters for all variables. Hybrid sampling is a kind of Markov Chain Monte Carlo sampling method which uses gradient information of the probability density we want to sample from to perform sampling more efficiently. The use of gradient descent helps in guiding the Markov chain toward higher probability areas while the Gaussian noise added at each step helps the MCMC method to sample around the high probable region. At each state p the next possible state p' is accepted with probability of $min(1, \frac{P(p')}{P(p)})$. In the hamiltonian sampling method, an analogy to physical system is made. Each state p is associated with a particle having kinetic energy dependent on the momentum and

potential energy. Thus for each state there is a hamiltonian H which is the sum of potential and kinetic energy, which is conserved. The probability at each step p is given by

$$P(p) \propto e^{-H(p)}$$

where $H(p)$ is the hamiltonian of the particle with sate p.

In our method H(p) is the negative log likelihood of the observation and latent variable together for a particular feature as given by taking log of Equation 5. Since we know the open form of $P(y_i|x_i)$ and $P(x_i|x_j \forall j \in N_i)$ at each state (Gaussian) we can calculate the derivative of the log likelihood ($\propto$ momentum) and hence follow the gradient while adding small Gaussian noise.

To search for the best parameters for each feature, we simply perform hybrid monte carlo sampling as usual and separately store the parameter for the given feature at which the maximum likelihood until now occurred. Thus to localize all facial features we simple cycle around the parameters of the various facial feature polygons several times, each time sampling from one facial feature's polygon parameter keeping all other parameters fixed. By doing only a few steps of hamiltonian sampling in each cycle for each feature, we can converge quickly towards the optimal parameters to extract features. Since we use gradient information to aid the hybrid sampling the search method is efficient and provides results almost immediately.

The algorithm for the location and extraction of facial features using the hybrid graphical model can be summarized as

1. Initialize parameters for the bounding polygons of all the features

2. for n = 1:number of cycles

3. for i = 1:number of facial features

(a) Calculate H_i the loglikelihood of the ith feature using current parameter setting

(b) for j = 1:number of samples per cycle

i. Calculate gradient at the current parameter setting

ii. Sample a new parameter p_i' for the feature by adding gradient and random Gaussian noise to the old parameter p_i.

iii. Accept parameter p_i' with probability $min(1, exp(H(p_i) - H(p_i'))$

iv. If sample was accepted then set $p_i = p_i'$

v. if sample probability for feature is higher than the best sample q_i upto now then $q_i = p_i'$

(c) end

4. end

5. end

6. The final parameters for the bounding polygon of the ith facial features is given by q_i

4.3 Semantic Tagging of Face

Once the key features like eyes, nose and lips are parameterized we are already half done with the task of semantic tagging. For instance the height of the triangle that models the nose is directly representative of length of nose and hence this value can be used in calculations when the user description of the nose is say ”long nose”. However we still need to tag other features like whether the person has a mustache or whether the person is wearing spectacles or not.

4.3.1 Related Work

Many systems that try to achieve part of the goal of semantic tagging of the face have been proposed. For instance in [34, 66, 35] the task of glasses detection (whether the person is wearing any eye-glasses or not) has been worked upon. However in all these works glasses detection has been the only task that had to be tackled. In [65] a system that can represent faces using elastic graphs has been proposed and this system can model if the person is wearing glasses or not, whether the person is male or female and if the person has a beard or not. In [68, 47, 2] semmantic tagging is performed for video explaining at a higher level the scenario contained in the video.

4.3.2 Proposed Approach

For the semantic tagging two kinds of descriptions are considered. The first is discrete and the next is continuous. The list of semantic features extracted and their type is summarized in Table 4.1. For the discrete features that are binary we simply try to detect whether the feature is present or not by locating the approximate region of the feature using the parameters and location of basic facial feature and using color information of that region try to decide if the feature is present or not. For instance, consider the detection of mustache. Based on location of node and lips we can determine approximate region of the mustache and based on the fact of whether the region contains enough percentage of skin color we can decide if the person has a mustache or not. For the continuous features, using color edge map in the approximate region of the feature, the features are first parameterized and based on the parameters the continuous value of description is assigned. For instance consider eyebrow thickness feature for mugshot based image. Based on eye location and the rectangle bounding the face the approximate region

Table 4.1. List of Features

Feature	Type
Spectacles, Beard , Mustache, Long Hair and Balding	Discrete (yes/no)
Hair Color	Discrete (Black/Brown/Blonde)
Nose Width, Length and Size, Lip Width and Thickness, Face Length, Darkness of Skin and Eyebrow Thickness	Continuous

of eyebrow is determined and based on color based edge map, the edges between eyebrow and forehead are determined and hence the rectangle (bounding the eyebrow) is used to parameterize the eyebrow. The height of the rectangle is then used for eyebrow thickness. Thus the Table 1 of features are detected and appropriate values are found and stored for each face enrolled.

Chapter 5

Retrieval Sub-System

The query sub-system performs the retrieval based on semantic description given by the user. It also prompts the user about which feature to query next. Figure 5.1 gives a pictorial description of the retrieval process proposed.

5.1 Query Handling

The main tasks in query handling in the proposed system is first converting the verbal descriptions of the user to the value system of the semantic descriptions in the meta-database. Then the user query is matched with the entries of the database probabilistically and the top retrievals are displayed to the user.

5.1.1 Related Work

As mentioned in Chapter 2, many face retrieval systems have been proposed which handle query retrieval and user feed-backs. However most of these methods are specialized for the image based matching and retrieval. In most of these methods the retrieval process is simply to peform image matching and display top n images. However our task is to match verbal queries to

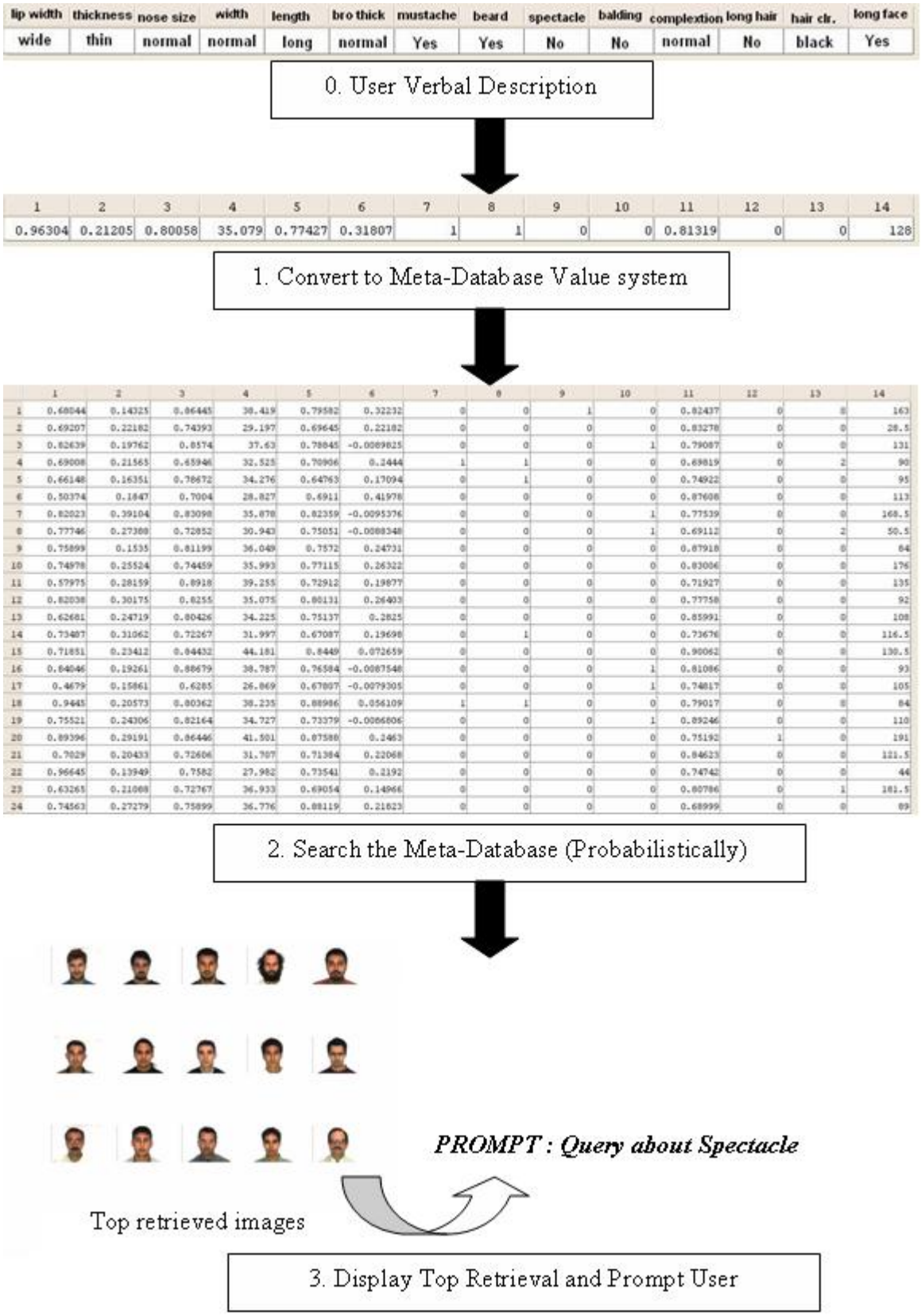

Figure 5.1. Example of Retrieval Process

the corresponding values about the semantics of the features stored in the meta database. In [25] a language model based query retrieval is proposed for general images and videos. However in our case the task is more specific and the user can provide exact details. However the matching instead of query formation needs to be probabilistic for effective retrieval.

Though no completely automated system for verbal query based face retreival exists, in [28] a system that can retrieve faces based on verbal descriptions of the face (very similar to ours) is proposed. However it must be noted that this system is not fully automated and the semantic descriptions of the enrolled faces in this system are manually (subjectively) evaluated. Our system automates the entire process. In this system unlike ours the semantic description of the images enrolled are all discretized. For instance for ”Nose Length” the entry in the meta database is long, short or normal. However in our system it is the actual normalized value of the length which allows better accuracies in retrieval. In [28] too a probabilistic retrieval algorithm is proposed.

5.1.2 Discussion

The most obvious method for retrieval in a verbal query retrieval would be by pruning. For instance when the user provides information that the target face has a beard then we can prune away all faces in the database without beard. However such a strategy for retrieval could prove dangerous in the proposed system for two reasons. First, errors in semantic tagging during system enrollent which may lead to potential target images to be pruned away. Second, user vagary in the verbal description. For instance, there is no exact defenition of what a long nose is. Hence we propose a probabilistic systems where the system sorts the face images according to their posterior probabilities of being the target given the user descriptions so far and displays to the user the top 15 images as retrieved images. There is a hidden advantage in this that users can

now make their judgement about the target faces relative to the top images they see. For instance, after the user has described say some 4 features, if he/she thinks that the target face has a longer nose than the ones displayed, the user can describe the nose as long. Such relative queries often help in effective retrieval. In our experiments with 25 users we found that absolute descriptions of the traget face by different users often do not match and that the relative description helped in better retrieval results.

5.1.3 Proposed Approach

Based on the description given by the user, the system at each stage orders the images according to their probability of being the face we are looking for. The system deliberately does not prune the images as pruning the images based on wrong information given by user would mean elimination of the required image from the list we are searching in. Initially, before user provides any information, we set the probability of a face being the required face $P(face)$to be $1/n$ where n is the number of faces in the database. Now as the user provides the description d_j about each feature f_i, we use this to update the probability using Bayesian learning as

$$P(face_k|f_i = d_j) = \frac{P(f_i = d_j|face_k)P(face_k)}{\sum P(f_i = d_j|face_k)P(face_k)}$$

After each query the prior probabilities for the faces are made equal to the posteriors found. The probability $P(f_i = d_j|face)$ of the feature f_i matching the description d_j for each face is set for binary attributes like whether the person has mustache or not by 0.9 if the face has feature f_i matching the given description d_j and 0.1 otherwise. The probabilities aren't set to 0 and 1 to make the system robust to user or enrollment system errors. The choice of values (0.1 and 0.9) can be varied by the user. These values represent the confidence in the importance of feature for retrieval. For continuous valued features, the probability is set by normalizing the continuous

value between 0 and 1.

5.2 Prompting the User

After each description provided by the user about facial features, the proposed system apart form displaying the top retrieved images also calculates which among the remaining (undescribed) features is most discriminative and proposes to the user this feature to query about if the user is clueless about what to query next.

5.2.1 Related Work

Generally Interactive retrieval systems achive user-system interaction by two approaches. The first is by user feed back where the user gives their opinion on the retrieval and the system evolves the retrieval results. The next is by system prompting the user about what to query next. Several relevance feedback based retrieval systems have been proposed [15, 16, 23] In [21] a face retrieval system is proposed where similar to the proposed system retrieval is performed by Bayesian inference and user feedback is given by the user selecting the images he/she feels is most relevent. The novelty of this system is that images are displayed in the order of their (eigen feature's) entropy. In this the system is very similar to proposed system that uses entropy for proposing which feature to query about next.

5.2.2 Proposed Approach

The system at each step prompts the user to enter a description about the feature that will help to effectively retrieve the required image. To do this the system should prompt the user to enter information about the feature having most entropy. More the entropy, more discriminative is the feature. For instance if half the people in the database wear spectacles and other half don't, it

would be a better feature to query about than a feature like fair skinned which most people in the database may be.

However there are two problems in doing this. Firstly, for finding the entropy we need to discretize the continuous values of features. However descretizing the values, we may loose relative information. For instance, it may happen that when we initially discretized nose length, the required person may have had a medium nose. But after a couple of queries it may happen that the required person's face has a longer nose among the more probable faces. Thus, by descritizing we cannot capture this information. The second problem is that we can't just find entropy of each attribute assuming that all descriptions of the feature are equally probable. The probability of a feature having a particular description is governed by the probability of faces having that description for the given feature.

To overcome the first problem, we discretize the continuous features into low, medium and high using appropriate thresholds and save them separately in a table while also keeping the continuous values in a table for calculating probabilities. Further, instead of assuming equal probabilities for all descriptions of a feature during calculation of entropy, we use the probabilities of the attributes given the current probabilities of faces. Given that each face k has probability $P(face_k)$ of being the required one, the probability of some feature f_i having description d_j is given by sum of probabilities of all faces which have $f_i = d_j$. For instance, the probability of nose being long is the sum of probabilities of faces with long nose. Thus we calculate entropy Hs_i for the ith attribute as

$$Hs_i = -\sum_{j=1}^{m} P(f_i = d_j|P(face))log_2(P(f_i = d_j|P(face)))$$

where m is the number of total values the attribute can take and

$$P(f_i = d_j|P(face)) = \frac{\sum_{k:f_{i,k}=d_j} P(face_k)}{\sum_{p=1}^{n} P(face_p)}$$

where $f_{i,k}$ represents feature i of face k.

5.2.3 Links with Decision Trees

Readers familiar with decision trees should have figured by now that the prompting sub-system has many commonalities with decision trees [40]. In decision trees root of the tree is the most entopic feature. As we traverse down the tree, the feature amongst the remaining features that is most entropic (wrt. posterior probabilities) is the next node. In the proposed system, if the user follows what the system prompts to query about next, he/she would infact be traversing down a path in the decision tree (formed by ID3 algorithm). However it must be noted that since some of the feature are continuous, creating the entire tree is very computationally intensive.

Chapter 6

Performance Analysis

A mixture of the AR database [38] and Caltech database [63] was used for testing the performance of the system. The AR database consists of 55 Female and 70 male subjects all in a white background. It has a total of about 250 mugshot images and remaining 3065 unconstrained images (with illumination effects, expressions and occlusion but white background.) The Caltech database consists of 450 images of about 27 subjects in varying illumination conditions and surrounding. It is to be noted that while all the images help in testing the enrollment subsystem, during query only one image per subject can be used for testing.

6.1 Example

An Example query is shown in Figure 6.1 where the user is querying about the person marked by the rectangular box. Each row in the Figure 6.1 shows the top five images after each query. Initially as all images are equally possible, the first five are just the first five images in the database. The required person was at the 31st place. Now when the information that the person was wearing spectacles was provided we see that images of people wearing spectacles appear in the top 5. The required person was at the 13th position. When the information that the person had

Figure 6.1. Example Query.

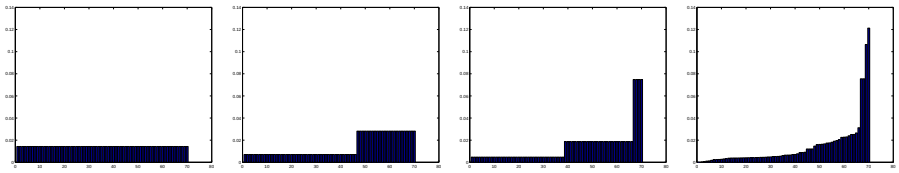

Figure 6.2. Plot of Probabilities

a mustache was provided he appears in the third position among the top five as seen in the third row of Figure 6.1. Finally when the information that the person had a large nose was provided we see that the person moves to second place among top five Figure 6.2 shows the probabilities of faces in sorted order for the above example query.

6.2 Evaluation

6.2.1 Evaluation of Enrollment Sub-System

Evaluation of Eye Localization using Bayesian Hough Transform To evaluate the improvement of the Bayesian approach to Hough tranform based eye detection over tradiotional Hough tranform based eye detection we manually marked 148 examples and then ran both the tradi-

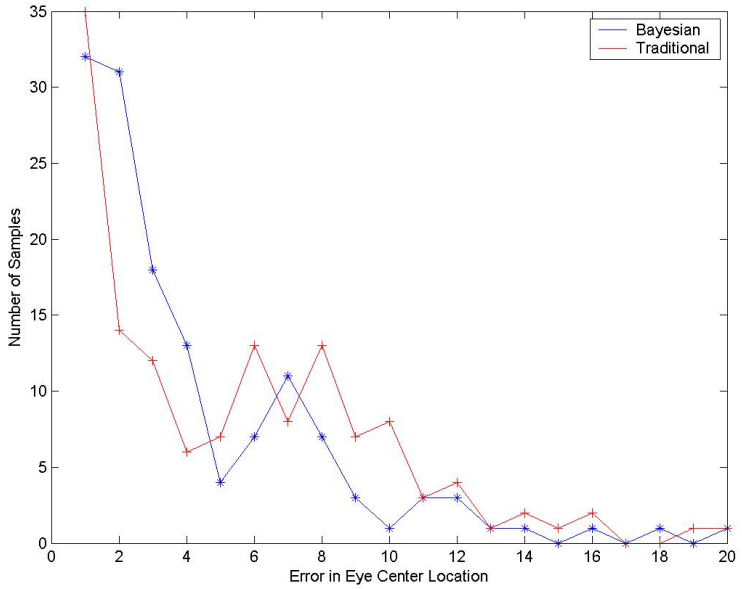

Figure 6.3. Errors With Respect to Manual Location of Eye

tional and bayesian approach of eye detection over the corresponding images. Figure 6.3 shows the histogram of errors on these manual instances. The error is calculated in pixel distances from manually marked eye centers and the ones found by the Bayesian method and Traditional method (where the mode of the accumulator values is chosen). We see that in the Bayesian approach most of the instances are in the relatively lower error region than compared to the traditional method. The total error of the bayesian method on all the 148 instances was 1451 px while that of the traditional method was 2725 px. Thus we see that the we gain significantly by the Bayesian approach.

Evaluation of Feature parameterization using Graphical model To test the performance on unconstrined images, the graphical model used had latent variables with dimensionality 15. This dimensionality for the latent variables was chosen because the reconstructions with 15

dimensional latent variables by PPCA lead to adequate result.

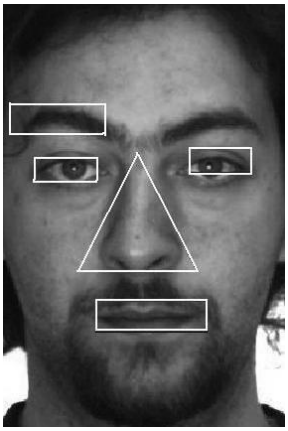

Figure 6.4. Example : Facial Feature Localization

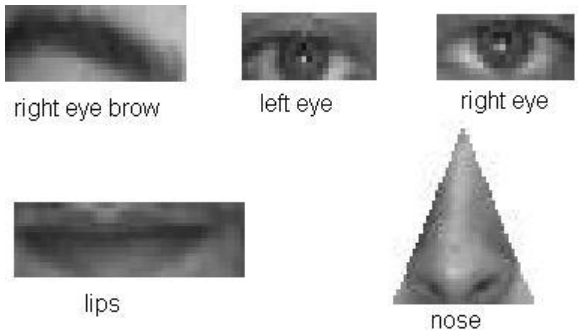

Figure 6.5. Extracted Features

Figure 6.4 shows the result of facial feature location on an example face image. Figure 6.5 shows the features extracted in that image. Figure 6.6 shows the result of facial feature location in a face image with illumination variation. Figure 6.7 shows the result of facial feature location in an image where the lips are occluded in the image. From these figures we can see the robustness of the system in cases of illumination variations and occlusion. It has to be noted that the training images contained only clean images with no illumination variations.

To evaluate the performance and efficiency of using the hybrid graphical model we tried facial feature detection,

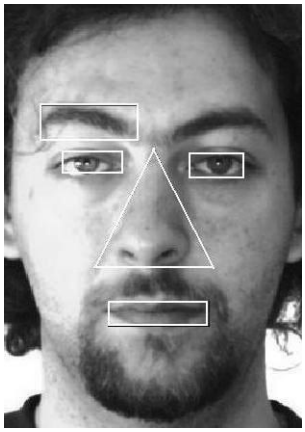

Figure 6.6. Example: Image with Illumination Variation

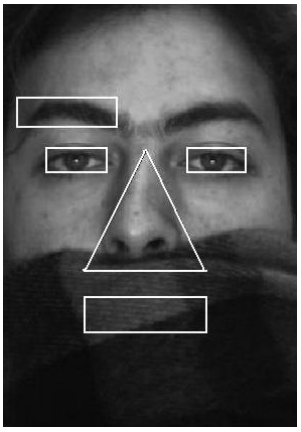

Figure 6.7. Example : Image with Occlusion

1. P1 : Based on PPCA for each facial feature independently without the use of gradient information

2. P2 : Based on PPCA for each facial feature independently but using gradient information

3. P3 : Based on the hybrid graphical model with gradient descent during sampling on the posterior

To test the performance of the system, the facial features where manually located and the parameters for the appropriate bounding polygons for each feature was found and stored. The

graph in Figure 6.8 shows the histogram of the errors for each of the 3 methods. Here by error we mean the squared distance between the polygon vertices manually marked and automatically found. We see that while P1 lies mostly in high error zone in the histogram, P2 and P3 cover areas in the low error zone. The total error (sum of errors on all the images) for P1 was found to be 14,306, for P2 was 12,202 and P3 was 10,473. Thus we see that P3 has a superior perormance.

Since the algorithms are randomized, their convergence rate cannot be directly compared as results vary each time. Hence, to analyze the efficiency and performance we plot the histogram of likelihoods. If many samples are drawn from high likelihood region it can be infered that the algorithm has faster convergence and also implies that the method does not get stuck in local maximas that often. Graph in Figure 6.9 is the histogram of likelihood for the three models. The x-axis corresponds to the likelihood and the y-axis shows how many samples were drawn from that likelihood. In all the three methods same number of samples were drawn with approximately the same number of iterations. Hence the plots can be directly compared. From the graph we see that P1 samples least from the regions of maximum likelihood. We also see that P2 has a peak at the high likelihood region but the peak is very steep which means that very little time is spent in the high likelihood region. Finally, it can be seen that P3 samples mostly from the high likelihood region. Thus we can see that the proposed model is much more efficient than independently extracting facial features. It also can be inferred that due to the hamiltonian sampling method which performs a gradient based sampling, the efficiency of the system is higher than panning through the entire image region in a brute force fashion.

Evaluation of Semantic tagging The Table 6.1 summarizes the overall results of the individual feature extraction of the enrollment sub-system. The performance on the continuous valued features like nose width can be evaluated by how well the polygons fit the features and how

Table 6.1. Performance of Query Sub-system on Discrete Valued Attributes

Feature	Number of False Accepts	Number of False Rejects
Spectacles	4	2
Mustache	3	4
Beard	4	1
Long Hair	2	10
Balding	1	0

Table 6.2. Average Queries Needed for Retrieval

-	Top 5	Top 10	Top 15
Average No. of Queries	6.64	4.59	2.72

easily the user can locate the required person.

6.2.2 Evaluation of Retrieval Sub-System

Experiments were conducted to test the usability and query capability of the system. 25 users were each shown pictures of 5 people in the database taken on different days and wearing different clothes from the ones in the database. Then the users were asked to input the verbal descriptions of the 5 faces to the system. Table 6.2 summarizes the average number of queries required to get the person we are looking for within top five, ten and fifteen images respectively for the 125 test cases.

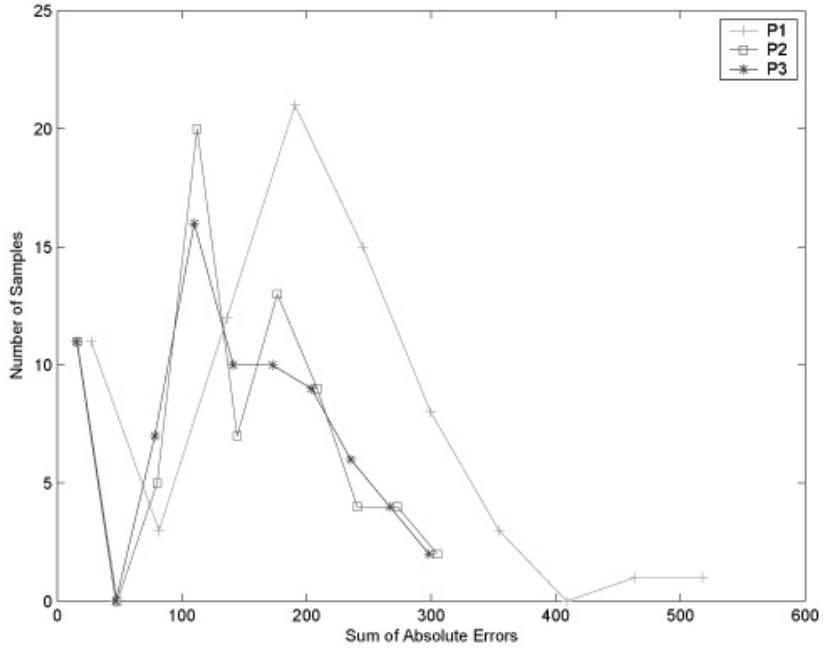

Figure 6.8. Errors With Respect to Manual Location

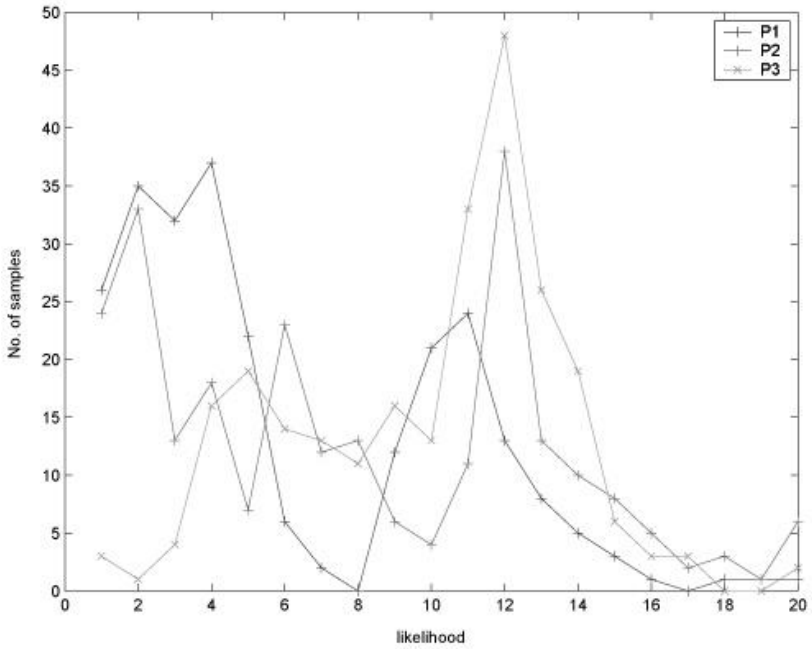

Figure 6.9. Histogram of Likelihood

Chapter 7

Conclusion

We have presented an interactive and probabilistic face retrieval system based on verbal queries. From the results shown above we can see that the system performs the extraction of semantic features from images effectively. The system described has applications in law enforcement for picking out the image of a suspect from a large database using the verbal description given by the witness. The system can work for datasets with mugshot images and unconstrained frontal images. It also has potential applications in soft biometric systems, suspect identification and surveillance tasks.

7.1 Contributions

The primary contribution of this work is that we give empirical proof that simple semantic features of the face which can be automatically extracted from color images of faces are sufficient for effective retrieval of faces the user is looking for. Our experiments with 25 volunteers prove that the strategy of probabilistic interactive retrieval is effective in the retrieval process. In the enrollment of mug-shot images, for locating the eyes we propose a Bayesian approach to Hough transform based circle detection which works effectively even in noisy conditions thus improving

over the basic method proposed in [6]. For the unconstrained image enrollment we propose the use of hybrid graphical model that not only models the relationships between facial features and their positions and scales but also unlike other methods [55] does not induce arbitrary causality by linking the latent variables for the facial features by directed edges. For the training of the graphical model we propose a siple technique of creating the training set such that the model can handle more variations in initial setting of feature parameters. The system we propose is interactive and helps the user when he/she needs to be prompted about what to query next.

7.2 Future Work

The system can be extended to work with video (surveillance) clips as well where along with semmantic video annotation, the semantic tagging of face images would prove valuable for easily spotting suspects based on witness descriptions. The graphical model proposed can be kernelized to achieve a non-linear modelling of features. Experiments to evaluate effectiveness of the system in surveilance mode where even the user is removed from the loop need to be performed.

Appendix A

Parameter Estimation

In Section 4.2.3 we proposed a hybrid graphical model for modelling facial features and their positions. This graphical model was a linear Gaussian model. We described the directed model by the Probabilistic PCA formulation and the undirected graph of latent variables using the Gaussian random field formulation. Now for estimating the parameters of the model, consider the joint probability of a particular feature (say the i^{th}) and its corresponding latent variable given all the other features and their latent variables and model parameters Θ.

$$P(y_i, x_i | y_{-i}, x_{-i}, \Theta) = P(y_i, x_i | x_{-i}, \Theta) = P(y_i | x_i, x_{-i}, W_i) P(x_i | x_{-i}, B) \tag{A.1}$$

where W_i is the matrix relating latent variable x_i and feature observed y_i. B is the matrix relating the latent variables in the Gaussian random field. Now if we want to optimize the negative log likelihood of this joint probability then by Equation A.1 it would be equivelent to minimizing

$$\mathcal{L} = -log(P(y_i | x_i, x_{-i}, W_i)) - logP(x_i | x_{-i}, B))$$

Further note that given x_i, y_i is independent of x_{-i}. Hence if we want to optimize $P(y_i, x_i | y_{-i}, x_{-i}, \Theta)$ with respect to the parameters, we need to maximize

$$\mathcal{L} = -log(P(y_i | x_i, W_i)) - logP(x_i | x_{-i}, B))$$

From this it is clear that optimizing w.r.t. the parameters of the directed part of the graph is independent of other latent variables (of other features) and hence the optimization takes a form similar to the probabilistic principle component analysis formulation explained in Section 4.2.3. Similarly, while optimizing w.r.t. B the partial derivative of $\mathcal{L}$ is independent of the y_i and parameter W_i. Thus the optimization is similar to optimization in normal Gaussian random fields. Thus the finding parameters that optimize the joint probability of y_i and x_i given other variables is simply the same as independently estimating parameters for PPCA and Gaussian random fields. Note here that even in the Gaussian random field we have a prior of unit covariance 0 mean Gaussian prior of latent variables. To perform parameter estimation we simply initialize all latent variables and the parameters of the directed portion of the graphs (ie. W_is) with their (independent) PPCA initialization (using for features the training set). Then based on this initialization of the latent variables evaluate the parameters of the Gaussian random field. (This can be done because of the unit covariance 0 mean Gaussian prior on latent variables.)

Bibliography

[1] Phantomas elaborate face recognition. *Product description*.

[2] Q. Z. And. Semantic video annotation and vague query.

[3] T. W. Anderson. Asymptotic theory for principal component analysis. *Ann. Math. Stat.*, 34(1):122–148, 1963.

[4] E. Baker. *The Mug-Shot Search Problem - A Study of the Eigenface Metric, Search Strategies, and Interfaces in a System for Searching Facial Image Data*. PhD thesis, The Division of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts, January 1999.

[5] R. K. Belew and L. B. Booker, editors. *Proceedings of the 4th International Conference on Genetic Algorithms, San Diego, CA, USA, July 1991*. Morgan Kaufmann, 1991.

[6] D. E. Benn, M. S. Nixon, and J. N. Carter. Robust eye centre extraction using the hough transform. *AVBPA Conference Proceedings*, pages 3–9, 1997.

[7] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *J. of Royal Statist. Soc., series B*, 36(2):192–326, 1974.

[8] N. Brace, G. Pike, and R. Kemp. Investigating e-fit using famous faces. *In Forensic Psychology and Law*, 2000.

[9] V. Bruce. Recognizing faces. *Faces as Patterns*, 3(1):37–58, 1988.

[10] G. Burel and D. Carel. Detection and localization of faces on digital images. *Pattern Recogn. Lett.*, 15(10):963–967, 1994.

[11] R. Chellappa and S. Chatterjee. Classification of textures using gaussian markov random fields. *IEEE Tr. ASSP*, 33:959–963.

[12] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models : their training and application. *Comput. Vis. Image Underst.*, 61(1):38–59, 1995.

[13] J. L. Crowley and F. Berard. Multi-modal tracking of faces for video communications. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 640, Washington, DC, USA, 1997. IEEE Computer Society.

[14] R. Dahyot, P. Charbonnier, and F. Heitz. A bayesian approach to object detection using probabilistic appearance-based models. *Pattern Anal. Appl.*, 7(3):317–332.

[15] J. R. del Solar and P. Navarrete. Faceret: An interactive face retrieval system based on self-organizing maps. In *CIVR '02: Proceedings of the International Conference on Image and Video Retrieval*, pages 157–164, London, UK, 2002. Springer-Verlag.

[16] T. Deselaers, D. Rybach, P. Dreuw, D. Keysers, and H. Ney. Face-based image retrieval one step toward object-based image retrieval. In H. Müller and A. Hanbury, editors, *First International Workshop on Evaluation for Image Retrieval*, pages 25–32, Vienna, Austria, September 2005.

[17] R. D. Dony and S. Wesolkowski. Edge detection on color images using rgb vector angles. *Proc. IEEE Can. Conf. Electrical and Computer Engineering*, pages 687–692, 1999.

[18] F. Dornaika and J. Ahlberg. Efficient active appearance model for real-time head and facial feature tracking. *amfg*, 00:173, 2003.

[19] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid monte carlo. *Phys. Lett. B*, 195:216–222, 1987.

[20] G. J. Edwards, T. F. Cootes, and C. J. Taylor. Face recognition using active appearance models. *Lecture Notes in Computer Science*, 1407, 1998.

[21] Y. Fang, D. Geman, and N. Boujemaa. An interactive system for mental face retrieval. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 193–200, New York, NY, USA, 2005. ACM Press.

[22] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The qbic system. *Computer*, 28(9):23–32.

[23] C. D. Frowd, P. J. B. Hancock, and D. Carson. Evofit: A holistic, evolutionary facial imaging technique for creating composites. *ACM TAP*, 1(1), 2004.

[24] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. pages 611–634, 1988.

[25] A. Ghoshal, P. Ircing, and S. Khudanpur. Hidden markov models for automatic annotation and content-based retrieval of images and video. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 544–551, New York, NY, USA, 2005. ACM Press.

[26] N. Gourier, D. Hall, and J. L. Crowley. Facial features detection robust to pose, illumination and identity. In *International Conference on Systems Man and Cybernetics*, The Hague, oct 2004.

[27] H. P. Graf, E. Cosatto, D. Gibbon, M. Kocheisen, and E. Petaja. Multi-modal system for locating heads and faces. In *FG '96: Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, page 88, Washington, DC, USA, 1996. IEEE Computer Society.

[28] V. Gudivada, V. Raghavan, and G. Seetharaman. An approach to interactive retrieval in face image databases based on semantic attributes, 1993.

[29] G. E. Hinton, S. Osindero, and K. Bao. Learning causally linked markov random fields. *Artificial Intelligence and Statistics*, 2005.

[30] R.-L. Hsu and A. K. Jain. Semantic face matching. *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference*, 2:145–148, 2002.

[31] M. Hunke and A. Waibel. Face locating and tracking for human-computer interaction, 1994.

[32] Jain, Dass, and Nandakumar. Adaptive mixture of local experts. *Proceedings of SPIE*, 5404:561–572, 2004.

[33] A. Jain and X. Lu. Ethnicity identification from face images. *Proceedings of SPIE International Symposium on Defense and Security : Biometric Technology for Human Identification*, 2004.

[34] X. Jiang, M. Binkert, and B. Achermann. Towards detection of glasses in facial images. *Pattern Anal. Appl.*, 3(1):9–18, 2000.

[35] Z. Jing, R. Mariani, and J. Wu. Glasses detection for face recognition using bayes rules. In *ICMI*, pages 127–134, 2000.

[36] S. Kawato and J. Ohya. Real-time detection of nodding and head-shaking by directly detecting and tracking the ”between-eyes”. In *FG ’00: Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, page 40, Washington, DC, USA, 2000. IEEE Computer Society.

[37] K. R. Laughery and R. H. Fowler. Sketch artists and identi-kit, procedure for recalling faces. *Journal of Applied Psychology*, 65(3):307–316, 1980.

[38] A. M. Martinez and R. Benavente. The ar face database. *CVC Technical Report*, (24), 1998.

[39] S. J. McKenna, S. Gong, and J. J. Collins. Face Tracking and Pose Representation. In R. B. Fisher and E. Trucco, editors, *British Machine Vision Conference*, volume 2, pages 755–764, Edinburgh, September 1996. BMVA.

[40] T. M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.

[41] R. M. Neal. Bayesian training of backpropagation networks by the hybrid monte carlo method. *CRG-TR-92-1*, 1991.

[42] E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. In *CVPR ’97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR ’97)*, page 130, Washington, DC, USA, 1997. IEEE Computer Society.

[43] J. Penry. Photo-fit. *Forensic Photography*, 3(7):4–10, 1974.

[44] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'94)*, Seattle, WA, June 1994.

[45] A. Pentland, R. Picard, and S. Sclaroff. Photobook: tools for content based manipulation of image databases. *Proc. SPIE: Storage and Retrieval for Image and Video Databases II*, 2185.

[46] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases, 1994.

[47] D. Ramanan and D. Forsyth. Automatic annotation of everyday movements, 2003.

[48] M. U. Ramos, J. Matas, and J. Kittler. Statistical chromaticity models for lip tracking with b-splines. In *AVBPA '97: Proceedings of the First International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 69–76, London, UK, 1997. Springer-Verlag.

[49] M. Reinders, P. van Beek, B. Sankur, and J. van der Lubbe. Facial feature localization and adaptation of a generic face model for model–based coding, 1995.

[50] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on PAMI*, 20(1), 1998.

[51] S. C. Sahasrabudhe and K. S. D. Gupta. A valley-seeking threshold selection technique. *Computer Vision and Image Processing(A. Rosenfeld, L. Shapiro, Eds)*, pages 55–65, 1992.

[52] K. Sobottka and I. Pitas. Localization of facial regions and features in color images, 1996.

[53] K. Sridharan and V. Govindaraju. A sampling based approach to facial feature extraction. *IEEE AUTOID*, pages 51–56, 2005.

[54] K. Sridharan, S. Nayak, S. Chikkerur, and V. Govindaraju. A probabilistic approach to semantic face retrieval system. *Lecture Notes in Computer Science*, 3546:977–986, 2005.

[55] E. Sudderth, A. Ihler, W. Freeman, and A. Willsky. Nonparametric belief propagation. *MIT Artificial Intelligence Lab Memo 20*, 2002.

[56] Sullivan, Blake, Isard, and MacCormick. Bayesian object localization in images. *Oxford/Microsoft*, 2001.

[57] Z. Szlvik and T. Szirnyi. Face analysis using cnn-um. *Proc. of CNNA04 IEEE*, pages 190–196, 2004.

[58] M. Tipping and C. Bishop. Probabilistic principal component analysis. *Technical Report NCRG/97/010, Neural Computing Research Group Aston University*, 1997.

[59] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[60] V. Vezhnevets, S. Soldatov, A. Degtiareva, and I. K. Park. Automatic extraction of frontal facial features.

[61] J.-G. Wang and E. Sung. Frontal-view face detection and facial feature extraction using color and morphological operations. *Pattern Recognition Letters*, 20(10):1053–1068, 1999.

[62] T. Wark and S. Sridharan. A syntatic approach to automatic lip feature extraction for speaker identification. *ICASSP*, pages 3693–3696, 1998.

[63] M. Weber. Caltech faces. 1999.

[64] T. Wilhelm, H.-J. Bohme, and H.-M. Gross. Classification of face images for gender, age, facial expression, and identity. *ICANN*, 2005.

[65] L. Wiskott. Phantom faces for face analysis. In G. Sommer, K. Daniilidis, and J. Pauli, editors, *Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, number 1296, pages 480–487, Heidelberg, 1997. Springer-Verlag.

[66] B. Wu, H. Ai, and R. Liu. Glasses detection by boosting simple wavelet features. *icpr*, 01:292–295, 2004.

[67] J. K. Wu, A. Y. Hock, P. Lam, K. Moorthy, and A. D. Narasimhalu. Facial image retrieval, identification, and inference system. *Proceedings of the first ACM international conference on Multimedia*, pages 47–55, 1993.

[68] G. Xu, Y. F. Ma, H. J. Zhang, and S. Q. Yang. An hmm-based framework for video semantic analysis. *Circuits and Systems for Video Technology, IEEE Transactions on*, 15(11):1422–1433, 2005.

[69] J. Yang and A. Waibel. A real-time face tracker. In *WACV '96: Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*, page 142, Washington, DC, USA, 1996. IEEE Computer Society.