Feature-based affine-invariant detection and localization of faces

Miroslav Hamouz

Submitted for the Degree of
Doctor of Philosophy
from the
University of Surrey

UniS

Centre for Vision, Speech and Signal Processing
School of Electronics and Physical Sciences
University of Surrey
Guildford, Surrey GU2 7XH, U.K.

April 2004

ALL MISSING
PAGES ARE
BLANK
IN
ORIGINAL

Summary

The accuracy of human face detection and localization in images and video is a crucial factor influencing the performance of biometric face authentication and recognition systems. Recently this subject attracted a lot of attention by researchers and companies and its applications emerged in various areas including surveillance, security, and computer games. This thesis describes a novel person-independent method for finding and localizing faces in authentication scenarios. Such scenarios involve situations where a person stands or sits in front of a camera in order to gain access.

The objective was to develop an algorithm which uses only still grey-level images, copes well in the presence of cluttered background and accurately localizes faces including eye centres. Many of the methods that have been reported in the literature only partially fulfil these requirements, in particular, a few methods focus on precise eye localization. To address these issues, we propose a novel bottom-up face detection and localization algorithm which exploits statistical feature detectors as the means of image capture effects removal. Our method uses both, a constellation (shape) model and shape-free texture model to select the best face location hypothesis among multiple hypotheses generated by the feature detectors. The constellation model utilizes a distribution of the transformation from a proposed model space into the image space. The texture (appearance) model is based on a cascaded Support Vector Machine classification. Both, an extensive analysis and a performance evaluation on several realistic face databases will be discussed in this thesis. We show that by utilizing the proposed verification of hypotheses, a significant performance boost is achieved compared to the performance of feature detectors alone.

Key words: face localization, face detection, face verification, face authentication, face recognition

Email: m.hamouz@surrey.ac.uk

WWW: http://www.eps.surrey.ac.uk/

Acknowledgements

First, I wish to express my deepest thanks to my supervisor, Professor Josef Kittler for giving me the opportunity to complete this research work. His professionalism and experienced guidance during the work were invaluable. Also the cooperation with Lappeenranta University of Technology proved extremely fruitful, I have to thank Joni Kämäräinen, Ville Kyrki, Pekka Paalanen and Professor Heikki Kälviäinen for their contributions. Namely, the expertise and enthusiasm of Joni was very important for the success of the method. Thanks also to Jiří Matas, Petr Bílek and Kieron Messer for their collaboration and discussions. I also wish to express my gratitude to all my old and recent friends for their support. I also thank my colleague James Short for a very careful scrutiny of this manuscript.

Finally, many thanks to my family and most importantly to Blanka.

This research was supported by the EU project BANCA and the University of Surrey.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the following chapters a novel face detection and localization algorithm will be presented. The focus of the method is face authentication, where a digital camera is used to verify a claimed identity. In order to put our task in context, we begin with an introduction to the field of computer face recognition, identifying its motivations, issues, and challenges.

1.1 Motivation

1.1.1 Biometrics

The field of biometrics has recently become a hot topic and significantly progressed towards real-life applications. The topic covers automated methods of recognizing a person based on his or her physical or behavioural characteristics. Among the most commonly used features belong face, fingerprints, hand geometry, handwriting, iris, retina, and voice. Biometric technologies are quickly becoming the foundation of secure identification and person's identity verification solutions.

1.1.2 Face Recognition and Verification

The goal of face recognition is to create a computer-based system which identifies people by using visual facial data. Face verification is a task which deals with situations when

an identity claim is made and the system should decide upon its correctness. Face recognition is a more general task, where the system, given an image or video sequence, attempts to establish the identity of the person by searching through a database of stored face templates and corresponding IDs.

Since faces appear in arbitrary positions and orientations in the image (depending on the scene and capture setup), they have to be first found and precisely localized. Before face verification/recognition algorithms can be applied, face(s) found in the image need to be registered. The registration involves geometric normalization (warping) of the face from the image into a predefined coordinate system which enables meaningful measurements to be made on the faces for their comparison. Often, the eye positions are used to register faces.

The focus of this thesis is face detection and localization. A face detection/localization algorithm takes an image, or a video sequence as the input and returns the position of the found face(s) and facial features. A diagram showing a face verification system is depicted in Figure 1.1.

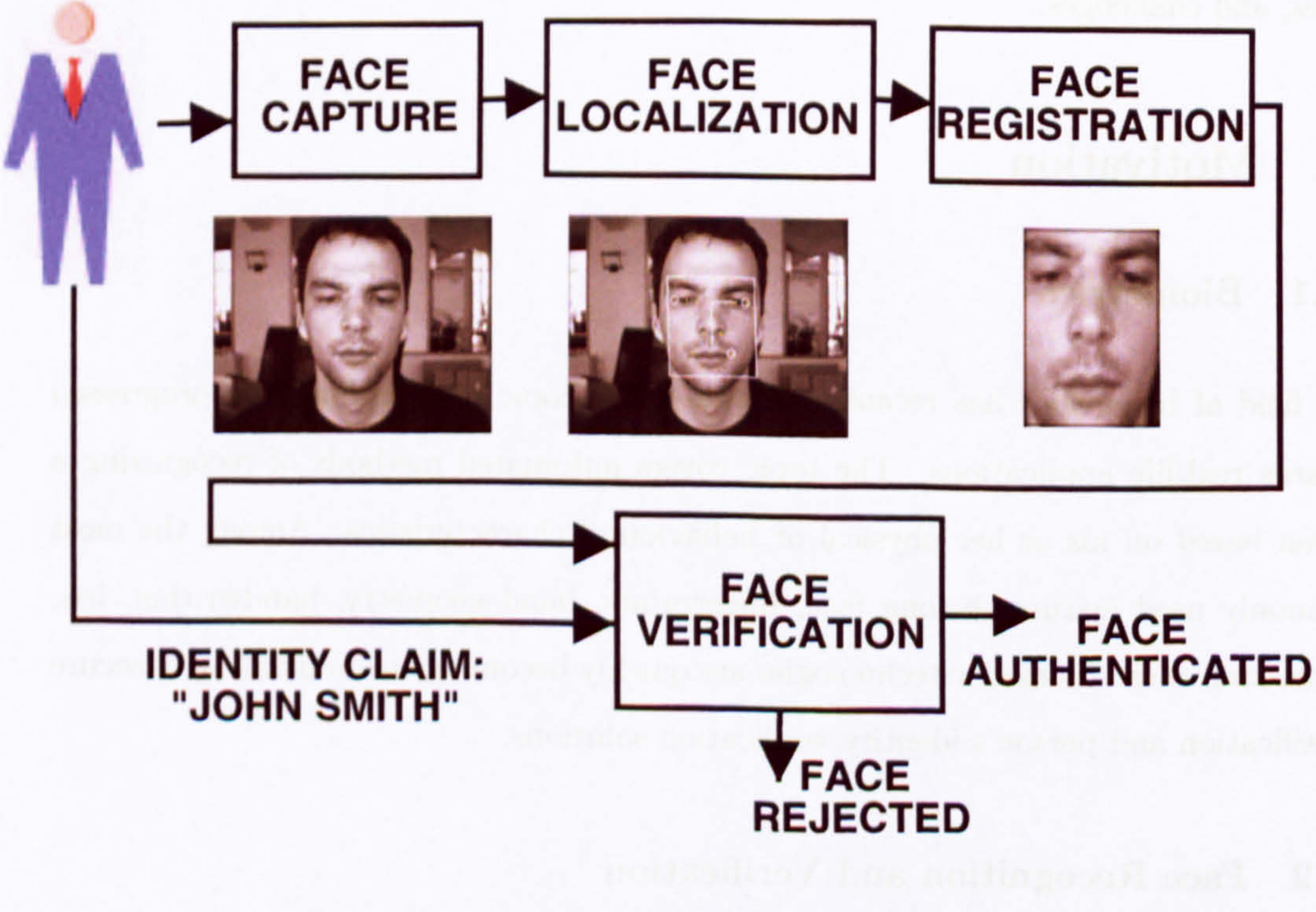

Figure 1.1: Diagram of a face verification system

It can be shown that the majority of the existing face recognition/verification algorithms

are very sensitive to registration errors. Since the localization process provides the input for face registration, its success, therefore, heavily affects the performance of the consequent phases of the system.

Sometimes the term "face detection" is used in the sense of localization by researchers. In this thesis we will refer to a rough estimate of face location in the image as "face detection" and to a precise localization including the position of facial features as "face localization". Such choice of terminology is consistent with most of the existing publications on this topic.

1.1.3 Face Localization

As mentioned above the accuracy of face detection and localization significantly influences the overall performance of a face recognition system. The reason is that a face has to be successfully registered before the verification procedure can be performed. Therefore this topic attracts great attention from researchers in academia and industry. In spite of the considerable past research effort, face detection and localization still remain challenging problems because faces are non-rigid and have a high degree of variability in size, shape, colour, and texture. The challenge also stems from the fact that face detection is an object-class recognition problem, where an object to be recognized is not just a previously seen entity, but rather an instance from a whole class of objects sharing certain common properties.

The method advocated in this thesis works on still greyscale images and exploits feature detectors, which intentionally generate multiple (possibly false) facial feature candidates in order not to miss the true ones. Triplets formed out of these detected candidates are then tested for constellation (shape) constraints and only the sound hypotheses are then passed to a face appearance verifier. The appearance verifier is the final test designed to reliably select the true face location based on photometric data.

1.2 Contributions

In this thesis a novel and successful face localization algorithm is presented. Its performance is assessed on difficult realistic benchmark data and extensive analysis performed. The algorithm could possibly be exploited in real-world applications and can compete with industrial solutions. Our goal is to give the reader exhaustive insight into this modern and attractive field as well as to discuss the proposed approach to face localization in detail. Our solution is versatile and facilitates integration of various feature detectors, and appearance and shape models.

The following three main contributions can be identified:

Novel detection and localization paradigm A versatile algorithm robust to feature detector failure exploiting detected features as the means of geometric normalization was developed. This method treats feature detection false alarms as a naturally occurring phenomenon and the final decision on the presence of the face is based on a shape-free appearance verification.

Gabor-based feature detector A functional design invariant to imaging effects for the detection of ten facial features based on Gabor filters will be presented. Although Gabor filters have previously been used in face detection and recognition our method integrates in a novel way the Gabor filter responses with a cluster-based classifier using a complex-valued statistical model. The currently used feature detector implementation comes from a close cooperation with Joni Kämäräinen from Lappeenranta University of Technology, Finland [KKK+02, HKKK03, Käm03].

Fast constellation and appearance based verifier A method designed to quickly select the best face location hypotheses, given a set of detected feature positions, using both, feature constellations and the photometric appearance information has been developed.

Extensive performance evaluation Unlike many other studies on face detection and localization, we evaluated the method using large face databases and stringent localization error measurement. We compared the results with the results

coming from other detection and localization methods designed for authentication scenarios [JKF01, KK02]. Our results are presented in chapter 7 where we show that our localization system is superior to the baseline methods. All our experiments are conducted on publicly available datasets, therefore an opportunity for performance comparisons is established.

1.3 Thesis overview

The state-of-the-art will be presented in chapter 2. In chapter 3, the proposed methodology for face detection and localization will be discussed and using the methodology the corresponding algorithm designed. In chapter 4, a detailed description of the advocated feature detectors will be given. Also challenges and problems of feature detection in general are discussed in the same chapter. Chapter 5 will introduce a new concept, a feature constellation model, and chapter 6 will give the reader an insight into image based appearance modelling. Exhaustive experiments and evaluation of the method will be presented in chapter 7 and the thesis will conclude in chapter 8.

Chapter 2

State of the art

In order to detect a face, a model of a face instance in the image has to be created. According to Hjelmås [HL01], the construction of a generic model of the face class has been tackled in the literature basically in two ways. We adopt his categorization which divides the methods into the *image-based approaches* and *feature-based approaches* and add one more category which we call *warping methods*. Some methods may overlap these categories, but this division roughly holds for most of the existing methods.

Although hundreds of methods have been proposed, claiming success on various data, no unified benchmark criteria exist. In many practical situations (e.g. face recognition) accuracy is definitely an issue, on which the performance of the whole system depends. The research experience bring us to the hypothesis that inaccurate detection is one of the main factors that limits the performance of the current face recognition systems and large-scale exploitation is therefore not feasible yet. Most existing methods just concentrate on an approximate face detection and its segmentation from the background. Often only an upright bounding box is presented as the output. Many face recognition/verification systems unfortunately require the face to be registered much more accurately. It is also true that in many cases only the largest face in the scene is required to be localized accurately and the other faces in the scene are not important. Such situations involve mostly face authentication systems with possible applications at cash points, home-working or online banking. On the other hand in the case of surveillance systems, it is desirable to localize all the faces in the scene

accurately regardless of their size or orientation.

In this study we focus on detecting faces in face authentication scenarios. It can be argued that for such purposes, a face has to have a certain size and facial details have to be recognizable. Sufficient resolution is therefore crucial for achieving not only precise localization, but mainly for subsequent face verification or recognition.

2.1 Detection versus localization

As mentioned above, the output of face localization algorithm can vary significantly from method to method. Fast, however imprecise localization of a face by an upright bounding box can be regarded as a successful result in the case of camera tracking but definitely not in face verification or recognition. There exist no common performance evaluation which would suit every situation. For authentication scenarios, accuracy is paramount and therefore the performance evaluation criterion should reflect that. As presented in chapter 7 we adopt a very stringent localization/detection criterion, which was proposed by other authors working in face authentication and is suitable for the target scenario. An important fact is that it takes into account position of facial features, in particular eye centres.

2.2 Image-based methods

In this group, faces are typically treated as vectors in some high dimensional space and the face class is modelled as manifolds in such a space. The vector space either uses pixel intensities directly, or usually some form of preprocessing is applied in order to reduce the dimension of image vectors. The separation of the face samples from a non-face class is carried out using various pattern recognition techniques (classifiers). Typically, huge training sets are required to learn the decision surface reliably and methods like bootstrapping exploited. These methods are holistic with regard to the face model, i.e. they do not decompose faces into features (parts), rather this representation models the entire face. The constellation of facial features is therefore implicitly encoded by

this model. The scene capture effects (scale, orientation, perspective) are removed in the upper-level of the system by using a so called "scanning window". Please note that by the word "orientation" we mean the rotation of the head in the image plane, not in the perpendicular direction. We will use the words "rotation" and "orientation" interchangeably. The concept of the scanning window is the root idea of these methods. To remove these imaging effects, exhaustive scanning with the window has to be carried out in multiple scales and rotations. This has huge implications for the model of the face class itself. Since it is not possible to scan all possible scales and rotations, discretization of scale and rotation has to be introduced. It is exactly this operation which makes the modelling of face appearance difficult and prone to false detections and misalignments. From the human point of view, the human face is a rather distinctive photometric object. When employing discretization in scale and rotation, the face model often has to cope with quite a big alignment error introduced by this operation. Put simply, this happens when a face in the probe image does not fit in the chosen size of the scanning window. This means that the face/non-face classifier has to learn all possible fluctuations of misaligned faces that do not fit exactly the chosen scale and rotation samples in order not to miss any face instance. As a result, not only the precision of localization decreases (since the classifier cannot distinguish between slightly misaligned faces) but the cluster of faces becomes much less compact and thus more difficult to learn.

One of the most successful methods in this group is represented by the work of Rowley et al. [RBK98] where an attempt is made to remove the scene capture effects by applying what they call "a router network". In this approach the face orientation angle is learned from the training data and thus exhaustive search through discrete rotations is avoided. Nevertheless, the discretization of the imaging parameters remains and with it the discretization inaccuracy since the router network output angles and scales are still discrete.

Other methods like the one of Osuna et.al [OFG97] involve a Support Vector Machine face/non-face classifier. A multi-view face detector coping with large changes of head pose has been proposed by Li et al. [LGL00]. Support Vector Regression is applied first to estimate the head pose and subsequently pose-specific Support Vector classification

used to reliably verify the presence of a face. Another multi-view face detector tackling large pose changes has been reported by Ng et al. [NG02], where the complex distribution of poses was modelled by a collection of view-based component SVMs and pose estimation was automatically performed by a single integrated process, which reduced computational costs.

A neural network using a low-dimensional feature space is used in the detector of Sung and Poggio [SP98]. Another example of a classifier used to discriminate between face and non-face patterns is the Sparse Network of Winnows architecture (SNoW) by Yang et al. [YRA00]. The method of Viola and Jones [VJ01] using Haar-like features and a hierarchical tree classifier has recently attracted a lot of interest due to its real-time processing capability. Nevertheless, the face in their approach is still delimited by an upright bounding box and a separate feature detector would have to be engaged to get a more accurate localization.

The algorithm of Fröba and Küblbeck [FK00] uses oriented template correlation to localize faces in grey-level images. A statistical model using edge directions within the area of human face, normalized in size and orientation, is learned from data. In the test phase the model is shifted over the image and for each position a similarity between the model and the underlying image patch is computed. The similarity measure involves normalized correlation between edges in the model and the image. The authors reported good localization capabilities in complex background scenes. Another method exploiting edge information was proposed by Jesorsky et al. [JKF01]. This method deserves a special mention as it will be considered as a baseline method in our comparisons due to its focus on face localization for face authentication. The method uses Hausdorff distance on edge images in a scale and orientation independent manner. Let $\mathcal{A} = \{a_1, \ldots, a_m\}$ and $\mathcal{B} = \{b_1, \ldots, b_n\}$ denote two finite point sets. Then the Hausdorff distance is defined as

$$H(\mathcal{A}, \mathcal{B}) = \max(h((\mathcal{A}, \mathcal{B}), h((\mathcal{B}, \mathcal{A})), \text{ where} \tag{2.1}$$

$$h(\mathcal{A}, \mathcal{B}) = \max_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \| a - b \| \tag{2.2}$$

$h(\mathcal{A}, \mathcal{B})$ is called the directed Hausdorff distance from set $\mathcal{A}$ to set $\mathcal{B}$ with the underlying

norm $\| \cdot \|$ on the points $\mathcal{A}$ and $\mathcal{B}$. The authors used a slightly modified version of the distance, which is tailored for image processing applications:

$$h_{mod}(\mathcal{A}, \mathcal{B}) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \min_{b \in \mathcal{B}} \| a - b \| \tag{2.3}$$

Their face model was optimized using genetic algorithms on a large set of face images (10,000). The detection and localization involve three processing steps, where firstly coarse detection is performed, then an eye-region model is used in a refinement stage and finally the pupils are searched for with the help of the Multi-Layer-Perceptron classification.

2.3 Feature-based methods

In this framework, most commonly local feature detectors are used. A face is represented by a shape (constellation) model together with models of feature local appearance. This usually implies that a priori knowledge is needed in order to create the model of a face (selection of features), although an attempt to select salient local features automatically was reported in the literature [WWP00, WEWP00].

One of the earlier pieces of work in this group involves the algorithm by Burl et al. [BLP95]. Their frontal face localization system exploits five feature detectors based on the Gaussian derivative filters (two eyes, the nose/lip junction and two nostrils). The effects of translation, rotation and scale are eliminated by mapping to a set of shape variables. A statistical shape model on these shape variables is then used to rank the feature constellations. The work of Weber et. al [WWP00] extends this approach to different viewing angles, where faces are again represented as constellations of rigid features (parts). Position variability is represented by a joint probability density function on the shape parameters of the constellation. This method automatically identifies distinctive features in the training set using an interest operator followed by a vector quantization. Another example of a feature-based method is the method of Vogelhuber and Schmid [VS00] where the shape model uses a distribution of angles between the lines connecting located features.

The method reported by Reinders et al. [RKG96] uses a neural network for accurate localization of facial features, mainly eyes. To overcome the problem of large variations between eyes of different people microfeatures are used. These are small parts of the feature sought, e.g. corners of eyes or parts of lids, which exhibit smaller variations than whole features. Microfeatures detected by a neural network are postprocessed by a probabilistic method which exploits the geometrical information of the microfeatures. The system needs an initial estimate of eye region and therefore could be used as a final step after face detection/localization.

Schneidermann and Kanade [SK00] model the face as a set of localized regions expressed as a statistic of wavelet coefficients and their position on the face. However a scanning window in discretized scale dimension has to be deployed. Sometimes configuration models are defined heuristically (capturing facts like the eyes are above the nose etc.) but usually a distribution of positions in a relative coordinate frame is used. The method of Yow and Cipolla [YC96] uses perceptual grouping of spatial-filter-based features found in the image.

An interesting method was proposed by Cristinacce and Cootes [CC03], where Adaboost is used in feature detectors and a probabilistic shape model is used to sort out the false combinations.

The main drawback of the above-mentioned approaches seems to be that they do not exploit all the available photometric information and use only small patches of the face, which leads to an increased false-alarm rate. Also feature detectors are often an ad hoc solution, without a proper analysis or design. The design of a feature detector is not an easy task, since facial features still exhibit quite a lot of appearance variation. Also the removal of scene capture effects (like scale and rotation) seems not to be an integral part and focus of the algorithms, but rather an intuitive construction.

2.4 Warping methods

The following methods stand out from the previous two categories, so that we felt to introduce them as a new category. This very important group of methods are the

approaches where the facial variability is decomposed into a shape model and the model of local appearance or the texture model in a shape-normalized space. Active Shape Models (ASMs) and Active Appearance Models (AAMs) proposed by Cootes et al. [CCTG95, CET98, ETC98, CWT00, CT01] should be regarded as the main representative.

In the training phase of AAMs, shape and face appearance variations can be learned independently, using the means of PCA, kernel PCA or other statistical models. However, usually these two characteristics are combined to produce a single parameter vector capturing both shape and appearance variations. In the test phase, the aim is to put the model in correspondence with the probe image, i.e. to find an optimal set of shape and appearance parameters for the given input image. This is done through an optimisation of a score function. The score function is based on the difference between the synthesised appearance and the appearance of the region in the image determined by the shape parameters. It is important to mention that the iterative search which results in the model being "warped" onto the image is directed exploiting a learned relationship between the model parameters and the residual error induced between a training image and a synthesised model example. These methods seem to be ideal as a final localization step, however reliable face detection using another method has to be employed first, since the iterative nature of the methods requires a good initial position and size estimate to converge. They have been applied to the problem of tracking and numerous analyses of face patterns.

The work of Gong et al. [GPR97, GOM98], Romdhani et al. [RGP99, RPG00], Li et al. [YLGL01] and Sherrah et al. [SGO01] focuses on modelling face variability connected to large pose changes of the human head (e.g. rotation in depth). Some of these methods [GPR97, RGP99] extend linear AAMs, where due to the nonlinear nature of shape variations between frontal and profile or near-to-profile views, PCA is replaced by a non-linear kernel PCA model.

A Pose Invariant Active Apperance model able to capture both the shape and the texture of faces across large pose variations from profile to profile views was proposed by Romdhani et al. [RPG00]. The work explores the problem of face reconstruction

and recovery from 2D view projections. A *2D Generic-view Shape Template alignment technique* using simple affine transformations for alignment of shape and texture is compared to a *Generic 3D shape model based alignment* exploiting local feature-based transformations.

The method of Li et al. [YLGL01] uses a dynamic multi-view model including even a sparse 3D Point Distribution Model. Their acquisition process [GMP00] avoided the use of 3D range data and allowed the sparse 3D model to be learnt from 2D images of different views. The view-based representation is used in the person-specific face tracker of McKenna and Gong [MG98], where focus is again on large pose changes. Appearance-based matching using a Gabor filter was developed for face tracking and pose estimation. This system exhibited real-time performance using a specialized hardware. Another view-based tracker has been reported by de la Torre et al. [dlTGM98], where colour-based segmentation and morphological operators are used to determine initial position of the face and face changes between frames are then modelled by the affine transformation.

Another group of methods dividing the facial variability into shape and local appearance is called Dynamic Link Architectures [LVB+93, WFKvdM97, KP97]. The preferred shape is defined by an energy function of a graph structure and local appearance of the image content associated with the nodes is modelled by Gabor jets, which are the responses of a Gabor filter bank on the image.

2.5 Summary

In this chapter a survey of existing methods was presented together with the analysis of their advantages and drawbacks. This analysis will be used in the next chapter where the methodology of our approach that aims at advancing the state-of-the-art in face detection and localization is developed. Let us summarize the drawbacks of the aforementioned approaches.

Scanning window Deploying a scanning window causes non-compactness of the face class, reduces the accuracy of localization and makes the detection process slow.

Due to the non-compactness of the face class, training the face/non-face classifier requires a huge set of representative examples. Moreover, the search for a face is not usually directed, but rather exhaustive, so there is a higher chance of false alarms due to the enormous number of hypotheses to be tested. The reduced speed capability and poor localization accuracy call for employing separate feature detectors (e.g. eye-centre detectors) as a postprocessing step.

Feature constellation methods Most local-feature approaches tackle the problems caused by the sliding window concept, but it can be argued that the configuration of features found in the image is not sufficient to distinguish between faces and non-faces. Thus the number of false alarms is often high. We believe that it is barely possible to construct a fully discriminative feature detector and therefore false alarms will always be a big problem with these methods. Imaging effects introduced by the scene capture (scaling, head orientation, etc ...) are removed using intuitive solutions rather than a systematic approach.

Warping methods These methods can be regarded as very successful from the localization accuracy point of view when starting from a good initial position. However modelling of the face-class appearance and shape is often only client-specific. These limitations mean that the use of an outside face detector becomes necessary in order to get a close-enough starting position and robustness against cluttered background in uncontrolled scenes. These methods are suitable to be used as a last step following a "sliding-window" based face detector.

Chapter 3

Methodology

In this chapter our methodology to face localization will be presented. We will recall the drawbacks of the state-of-the-art and present the criteria which the designed algorithm should satisfy so that an improvement could be achieved. Taking the predefined criteria, the structure of a suitable localization algorithm will be deduced.

3.1 Critique of the state of the art

Existing object representation schemes provide models for global appearance (we referred to them as image-based methods) or for local features and their relationships. In the context of generic face detection and localization, the human face should be regarded as an object class, i.e. set of similar objects.

Image-based approaches (section 2.2) use the scanning window technique in order to locate faces regardless of scale and rotation. This concept causes a significant localization inaccuracy due to scale and rotation discretization as explained in chapter 2 (scale and orientation are continuous variables and only limited number of quantized discrete samples can be exploited). There is an important conclusion arising from this observation. The face/nonface model should not incorporate scale and orientation discretization invariance in order to achieve good localization accuracy (it should be able to discriminate between slightly misaligned faces). The scale and orientation variability

should therefore be fully removed from face/non-face templates before they are used in the learning stage, in other words this model should be fully scale-and-orientation free. If this condition is met, the scanning window technique cannot be used to localize faces, since if the true scale and orientation is missed, face would most likely be missed too.

An important observation is that the scanning window models based on powerful pattern recognition techniques are reported to be superior to the existing feature-based approaches in the case of still grey-level images [HL01]. It was argued that these models are better at coping with cluttered-background. Feature-based systems on the other hand seem to be suitable only for scenarios, where motion, colour or controlled conditions are available.

Feature-based approaches mainly suffer from the fact, that the constellation of features alone does not guarantee the discrimination from background (section 2.3). There are however several advantages of using features in localization. Firstly, is the ease of illumination distortion modelling. This is due to the fact that local features are spatially smaller than the whole object and therefore simple lighting invariance models can be successfully used in the modelling of unknown lighting variations. For scanning window methods, illumination actually presents a hard problem to solve. Although this topic is currently being heavily researched, we will not focus on it.

Secondly, although the modelling of local features like eye and mouth corners is still regarded as a hard problem [HL01], the overall modelling complexity of local features is smaller than in the case of image-based methods. Again due to the size, the modelling of a smaller object is usually less complicated than the modelling of a large object. This observation is indirectly related to the problem of training set size in the statistical learning and the curse of dimensionality. Having a less complex object to model means that the chance of success using all the available means is higher than in the case of a large and potentially highly variable object. Scanning window methods use modern pattern recognition algorithms to capture both facial shape and appearance implicitly. Although feature-based approaches actually decompose object structure into local appearance and configuration (shape), it is still possible to exploit all the existing

powerful pattern recognition tools, that were successfully applied in scanning window methods.

The third reason which makes local feature models interesting is their robustness to feature detector failure. If, for whatever reason, a local feature detector fails, it usually does not mean that the object will not be recognized. Proper local feature models use a redundant representation facilitating more than one local feature. Thus, an isolated feature detector failure does not necessarily result in the failure of the whole detector.

In our view a promising method of localization could be a combination of the aforementioned main streams. To give a simple example of such a system, imagine an approach using a colour feature-based technique as a preprocessor to the multiresolution window scanning technique in order to reduce its time complexity.

The discussion above can help us to draw the set of criteria a localization algorithm has to meet in order to be superior to the existing methods.

1. To maintain accuracy, scale and orientation discretization error invariance should not be incorporated in the face/nonface model. In other words, it means that the face model should regard even slightly misaligned faces as nonfaces.

2. Local features should be used due to their favourable properties regarding object-class variability and illumination correction (they are easier to model than the whole face).

3. Since the constellation of local features is not discriminatory enough for a reliable separation of faces from a cluttered background, powerful pattern recognition approaches, commonly used in the scanning-window methods, should be considered for the final decision on the presence of a face in the image.

4. If used, the local feature model should be robust to feature detector failure.

Let us discuss these four requirements and draw conclusions from them.

Firstly, the scanning window technique cannot be used, since it would require scale and rotation discretization error invariance being incorporated into the model.

From the first item on the list of requirements, i.e. that the face/nonface model should be fully scale and orientation free, a certain kind of registration of the tested templates that removes scale and orientation from faces has to be considered. A natural way is to introduce a scale-and-rotation invariant space, where the images (or their parts) will be registered. The face/nonface model can then be trained using registered templates and this should maintain the localization accuracy, since a slightly misaligned face would be likely to be classified as a non-face.

If the model is made fully scale and orientation (rotation) free, one cannot use a scanning window search through a set of rotations and scales during the detection (since faces not fitting the chosen scale and rotation would likely be missed). Here the local features can play an important role. Due to their nature, they could be used to facilitate the estimation of real scale and rotation of faces in the scene, as e.g. in many approaches dealing with stereo vision. The first item of the list of requirements effectively means, that a registration of faces into a special coordinate system is required (for removing scale and orientation from model faces). The use of the constellation of features on the face could give us an opportunity to estimate the scale and rotation and subsequently perform the registration of the data.

However, if local features are to be used we have to expect a large number of false positives, because the approach cannot avoid them in a cluttered background, as mentioned earlier. Also since some features are likely to be undetected, the proposed model has to be redundant in order to cope in a robust way.

Once facial hypotheses based on the constellation of features found in the image are generated and scale and orientation removed (i.e. registration performed), techniques from image-based approaches can be used to verify the presence of face in the given position (i.e. face appearance tested).

Combining the above arguments results in the following structure of the localization process (Figure 3.1).

Let us go into more detail now:

Figure 3.1: Diagram of the proposed algorithm

3.2 Feature detectors

The problem of detection of object parts has been attracting quite a lot of attention and many algorithms have been designed for this purpose. For simple image primitives like corners, edges and circles, well established algorithms exist. As discussed in chapter 4, at the beginning of our research we carried out experiments using these simple image primitives, in particular the Harris corner detector, combined by a simple statistical model of local appearance, but it finally proved to be incompatible with our further requirements.

In contrast to the Harris corner detector, a Gabor filter bank representation (described in detail in chapter 4) exhibited good ability to perform scene capture invariant modelling and at the same time extract the discriminatory part of the visual information in a computationally affordable way. As we discuss later we designed Gabor-filter based statistical feature detectors for accurate facial generic-feature detection. In our latest setup we aim to detect ten facial features. We came across a great deal of difficulty in modelling these features over a variety of existing human faces. Even simple features like "eye corners" or "eye centres" are visually highly variable as seen in Figure 3.2.

3.3 Face space

In order to allow geometric registration of the patterns and provide at least scale and rotation invariance ability, a special coordinate system will be introduced. As we describe later, in such a space even affine distortion (including scale and orientation as two major scene capture effects) is removed from the face patterns. We call this concept "face space". Although the details regarding its construction and motivation will be explained later in chapters 4 and 5, let us stress here that the main purpose

Figure 3.2: Visual variability of eyes. Samples taken from the BioID database

of this space is to enable the following two operations. First, by using this space we want to perform a geometric registration and normalization of possible face patterns by exploiting detected features in the image. Second, using this normalized data we want to perform a shape-free appearance verification as suggested in the proposed schema shown in Figure 3.1.

3.4 Hypothesis generation

As mentioned before, we believe that it is impossible to detect facial features in the image without false positive detections. Simple objects can appear similar to other visual stimuli in the scene so it is natural that the local feature detection should result in many features found in the image. Every statistical model captures just a part of the object visual information content, and therefore a complete discrimination from other objects is hardly achieved in practice.

In chapter 5 we will show, that by taking triplets of detected image features as face hypotheses, the removal of the scene capture effects can be performed (by facilitating the "face space" for this purpose). Such geometrical normalization then enables a shape-free test of appearance.

Let us mention here, that the separation of the shape and appearance variability is the crucial point of the algorithm. The final decision whether a face is present or not is not made using only the constellation of detected local features (like many other feature-based algorithms do) but by comparing all the underlying photometric information against geometrically normalized face appearance model (as explained in the next section).

3.5 Appearance verification

It was already mentioned earlier and our experimental evidence will confirm, that just configuration of local features does not carry enough discriminatory information to distinguish the face from the background. Many existing techniques just using a

configuration model built over the detected features rely on the fact that their feature detectors will have low false alarm rate. This situation however does not reflect real life scenes with cluttered background, where lots of false alarms would be encountered. In our algorithm we use a final appearance test exploiting powerful classification technique to avoid these false alarms. A whole chapter of this thesis is devoted to this topic.

3.6 Summary

In this chapter the motivations and the philosophy of the advocated approach to face detection and localization are presented and discussed.

We postulate in this thesis that in order to accurately localize a face, shape (expressed through feature constellation) and appearance properties have to be satisfied. We decompose the facial visual variability into a shape model (feature constellation) and shape-free appearance model. Unlike warping methods no iterative score function optimisation is required and facial hypotheses can be decided in two steps. Firstly, if a configuration of a group of detected features in the image (in particular three features form the group as we show later) does not satisfy learned constellation model, it is discarded. Then only the promising face hypotheses are passed on for the appearance test. Before that, the detected features are used to remove the scene capture effects and shape variability, i.e. the geometric registration of the underlying image patch is performed. The appearance model therefore uses geometrically aligned data and this fact represents an important difference as compared with the majority of image-based approaches where scale and orientation variability partially remains in the tested data and the classifier (model) must compensate for it.

As mentioned earlier, most of the existing classifiers are extremely sensitive to pixel misalignments (introduced by scaling, translation and rotation at image capture). Thus a significant degree of insensitivity (invariance) to slight shifts and changes of scale and rotation must be exhibited by the face model in order not to miss a hypothesis when the scanning window does not exactly fit the face. Such invariance, however, results into localization inaccuracy since the model is not able to distinguish between slightly misaligned versions of the target object. If the shape information is used, the

appearance model can be trained without this undesired discretization error invariance, because the invariance to scene capture effects and shape variations is incorporated in the shape (constellation) model itself.

We shall demonstrate that such an approach results in a higher localization accuracy and is consistent with the predefined criteria.

In the following chapters, the particular parts of the proposed solution will be explored in detail.

Chapter 4

Feature detectors

4.1 Face features

Since the feature detector is the initial step according to the proposed schema in Figure 3.1, the accuracy and reliability of the whole detection/localization system will critically depend on the accuracy and reliability of the detected feature candidates. There are two fundamental issues with feature detectors that need to be addressed.

- Which criteria feature detectors should satisfy
- Which in particular and how many features should be detected

In this thesis we regard the facial features to be informative sub-parts of faces represented by a reference point. They may appear in arbitrary poses and orientations as faces themselves and therefore the design of feature detector has to reflect this. As any detector in general, feature detectors have to be designed in an illumination and pose invariant manner. They also have to be able to detect the whole class of features over the entire population not only the features of a specific person.

Regarding the pose, the scene capture can of course introduce general perspective distortion to the face (and thus to the face features). When we attempt to model these scene capture variations, we have to realize that every additional parameter in the scene

capture model will increase the computation complexity of the detector. Therefore our aim should be to keep the number of parameters small. If we look at the problem closer, we find out that faces are quite flat objects apart from the nose. It is also true that in the authentication scenarios pose variations are small, faces are in fact always frontal (person standing/sitting in front of camera) and if not too close to the camera, the nose does not present a big problem. For such situations the simpler affine model would approximate the scene capture effects well. Although we deal with the affine model in detail in chapter 5, it should be mentioned here that with feature detectors it is desirable to go even further with simplification, since facial features are much smaller than the whole face. For this reason it is justifiable to approximate feature pose variations by a similarity transformation model (i.e. even less complex model than general affine or perspective). The resulting inaccuracies (errors introduced by this approximation) would have to be handled by the statistical appearance part of the feature model. The similarity model involves just translation, rotation and isotropic scaling, therefore our detector has to be made invariant to at least these operations. With such an approach to scene effects invariance we believe the computational feasibility will be maintained and at the same time any error introduced by this simplification will hopefully be manageable, since local features are small and thus less complex objects.

We also cannot expect that false positives will not occur. Features, as small parts of the face, can therefore be visually similar to other objects in the scene. Complete discriminativeness is hardly achievable in practice, therefore the proposed schema 3.1 reflects that.

Regarding the issue of which features to detect, first we have to identify what will be the features exactly used for. As suggested in chapter 3, features should be not only used to navigate the search for the face but also for scene capture effects (mainly scale and orientation) estimation and removal. For such a purpose, features could be used in a similar manner as they are used for example in stereo vision. In stereo vision, point correspondences are used to estimate the scene geometry taking two or more different views. Motivated by the use of correspondences in stereo vision, face features can help us to establish the size as well as orientation of faces by computing correspondences between the model face of a known size and orientation and a hypothesised face in the

test image. However, it has to be noted, that since face localization and detection deals with the whole face class (object-class modelling) the correspondence model should reflect that. For this purpose the concept of face space becomes useful, as mentioned in chapter 3. The introduction of face space will allow us to create a model, which can be used to estimate size and orientation of the face in the image (later it will be shown that in fact the whole affine geometry can be estimated). The details of this coordinate space will be described later in Chapter 5. Here in connection with the feature detectors, we just need to know that this space is defined uniquely for each face and removes scene capture distortion by the means of affine normalization. In other words, faces are geometrically normalized by mapping into the face space, where they have the same size and orientation. A very important fact is that in this coordinate system the facial features (like eye corners, eye brows, etc.) appear approximately in the same positions.

Having such a coordinate system, we can use the correspondences between features detected in the image and features residing in the face space (since most of them appear in the same position in the face space, a single point is a good representation of all of them) and we can register the underlying image patch (possibly face) into the face space. This process effectively means removing scene capture effects and possibly facial shape variations. A geometrically normalized photometric data can be then used in further steps as suggested in chapter 3.

The versatile design of the whole algorithm actually does not require the feature detector to be 100% successful. As mentioned earlier, false alarms are to be expected and sometimes undetection will also occur, e.g. due to occlusion. Our primal requirement then should be that the true feature will be included among the detector output in as many cases as possible (i.e. true negative error be kept low), and that the false alarms will be not excessive. To control the tradeoff of the two detection errors (false negatives and false positives), a threshold can be used. We should not even require that the real feature's rank has to be high in the feature detector output list. It is reasonable to require that for each feature detector, a fixed number of best feature detections will be taken as output and among them hopefully as many true features will be present as possible.

4.2 Harris-and-PCA-based feature detectors

Having a particular feature detector design we have to evaluate how accurate and successful it is using the face space coordinates and determine whether it is desirable to detect the particular feature or not. The reader is reminded here, that the feature detector itself of course works in the original image space coordinates and has to cope with scene capture effects, illumination and with population appearance variations.

In our early experiments [BMHK, MBHK02, HKMB02] our first feature detector was based on the Harris corner detector and subsequent PCA-based classification of the colour neighbourhood of Harris points (PCA as a method will be discussed in detail in Chapter 6). As we later show in Section 7.2 on the XM2VTS database the results were promising. However, we found later, that in the presence of cluttered background this detector performed poorly.

To choose which particular facial features to detect, an experiment was performed where the Harris corner detector was run in several scales on a set of face images, detected points were then projected into the face space and the places with highest occurrence of hits were labelled as feature candidates. Figure 4.2 depicts this result. The points which received the most hits were the eye inner and outer corners, the eye centres, nostrils and mouth corners - see Figure 4.3. They were selected as features to detect. These results were in accordance with our intuitive feeling, i.e. that smooth areas are not suitable for detection. The aforementioned features represent the areas of the face where a change of intensity is clearly visible and also the intensity forms some characteristic shape. It is worth mentioning that although we use the term "points" we actually mean areas (or parts) of face. A point is spatially infinitely small and thus virtually undetectable in the signal. As mentioned above, we represent each feature by a reference point (does not need to be a centroid in general) and some neighbourhood around it.

Examples of the Harris corner detector response on the face are depicted in Figure 4.1. In Figure 4.4, a typical feature detection result on images from the XM2VTS database is presented.

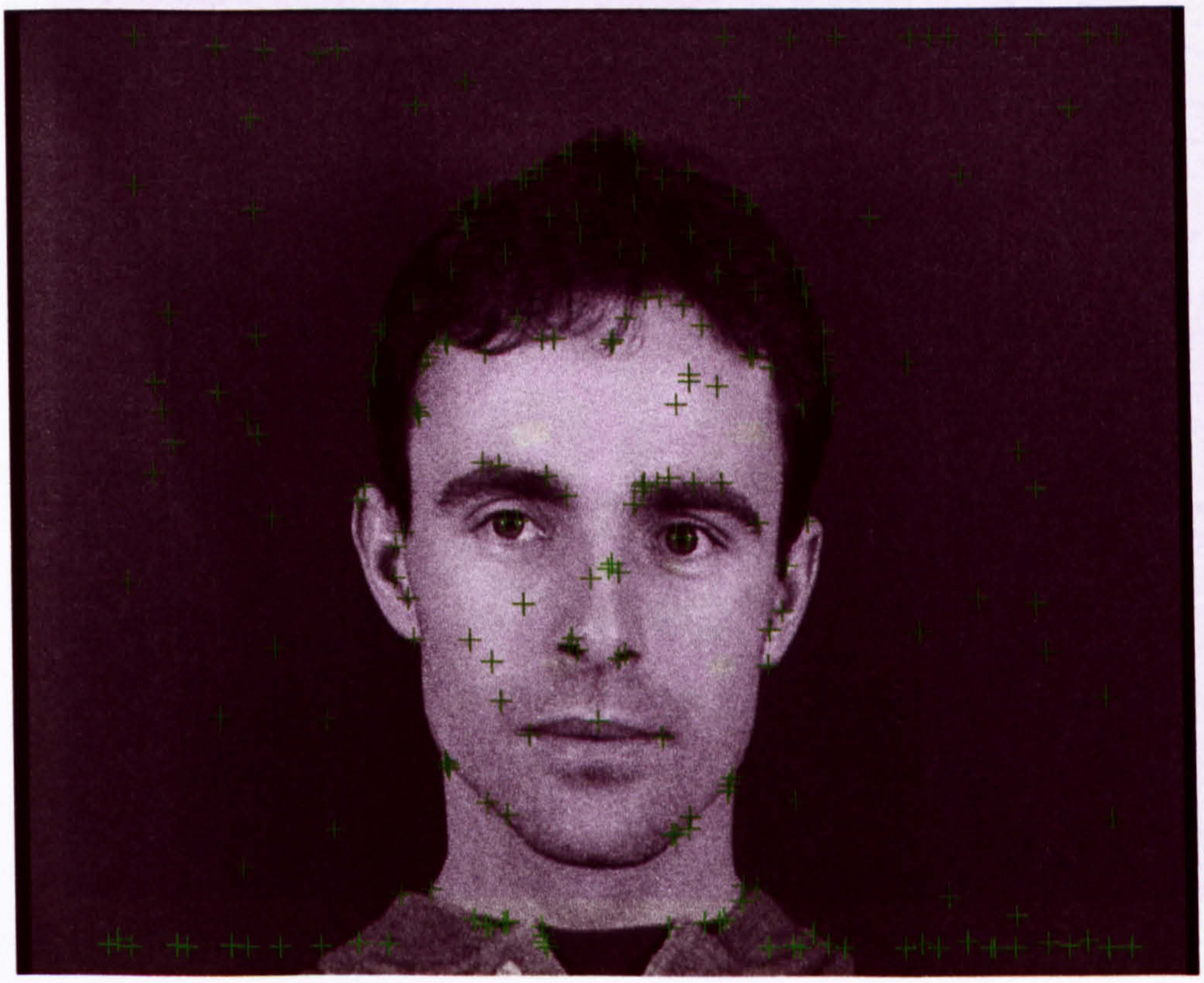

Figure 4.1: Typical Harris corner detector response on the face

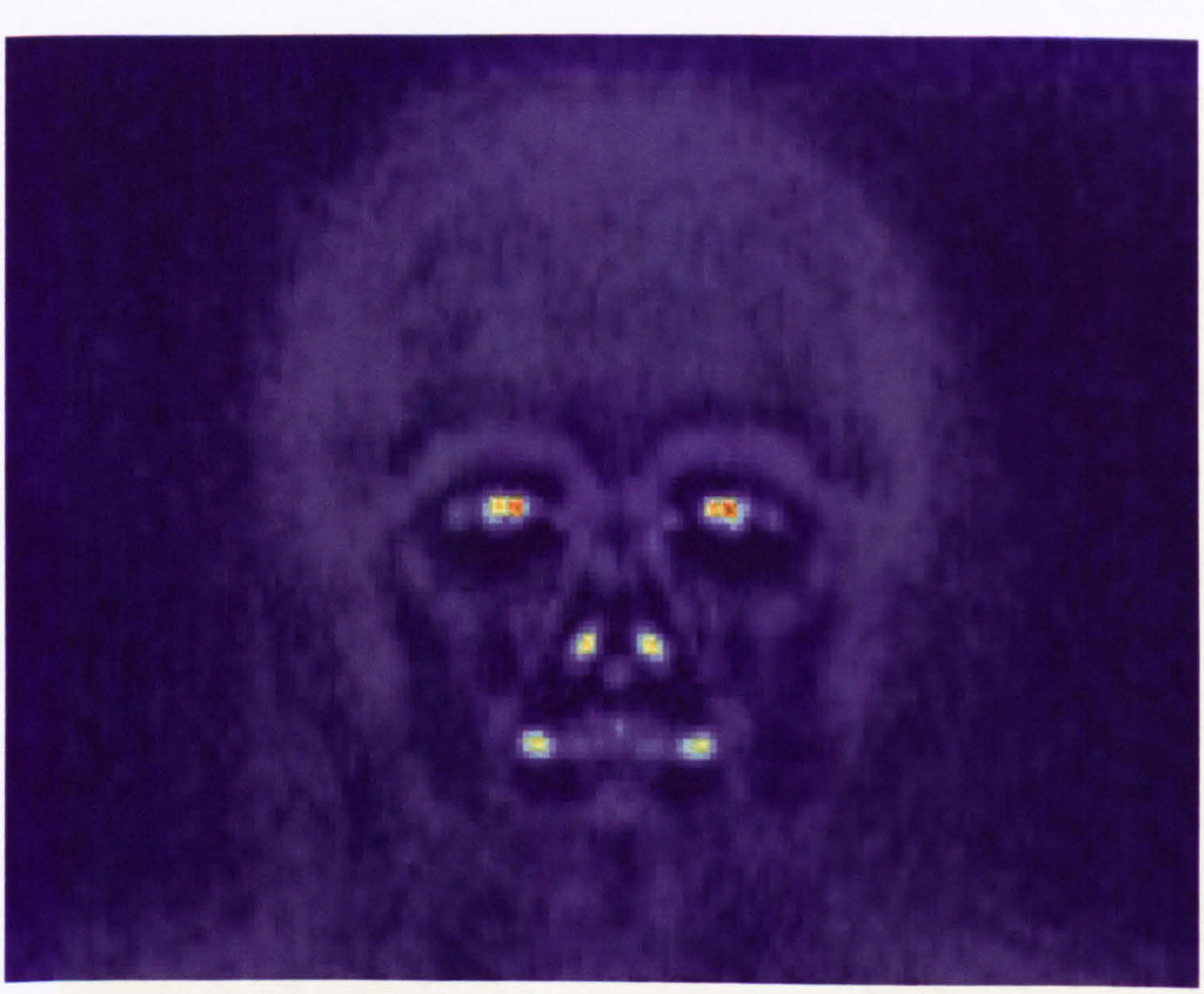

Figure 4.2: Harris corner detector success on various parts of the face, the brighter colour corresponds to higher occurrence of hits (experiment carried out by P. Bílek [BMHK])

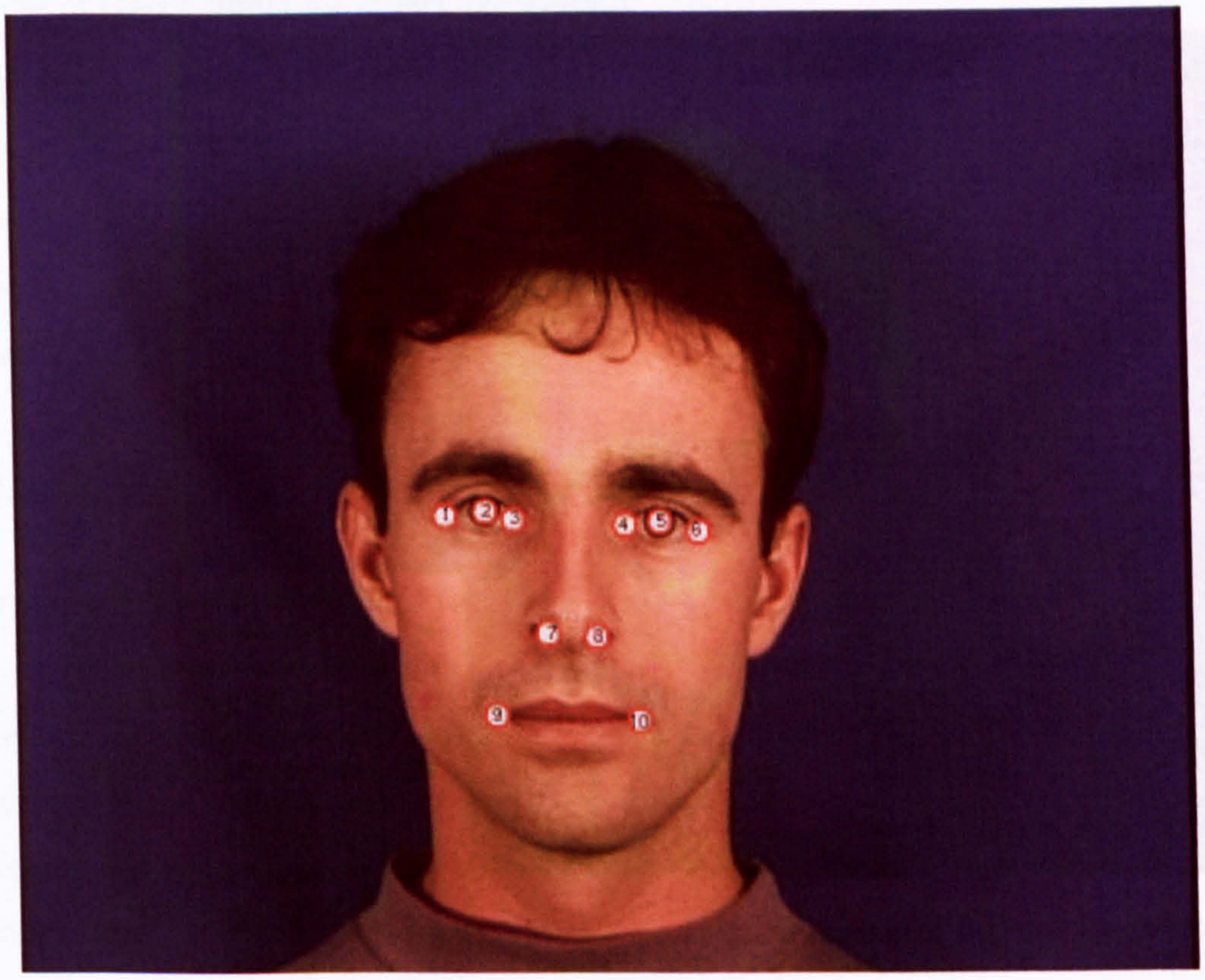

Figure 4.3: Features chosen for detection

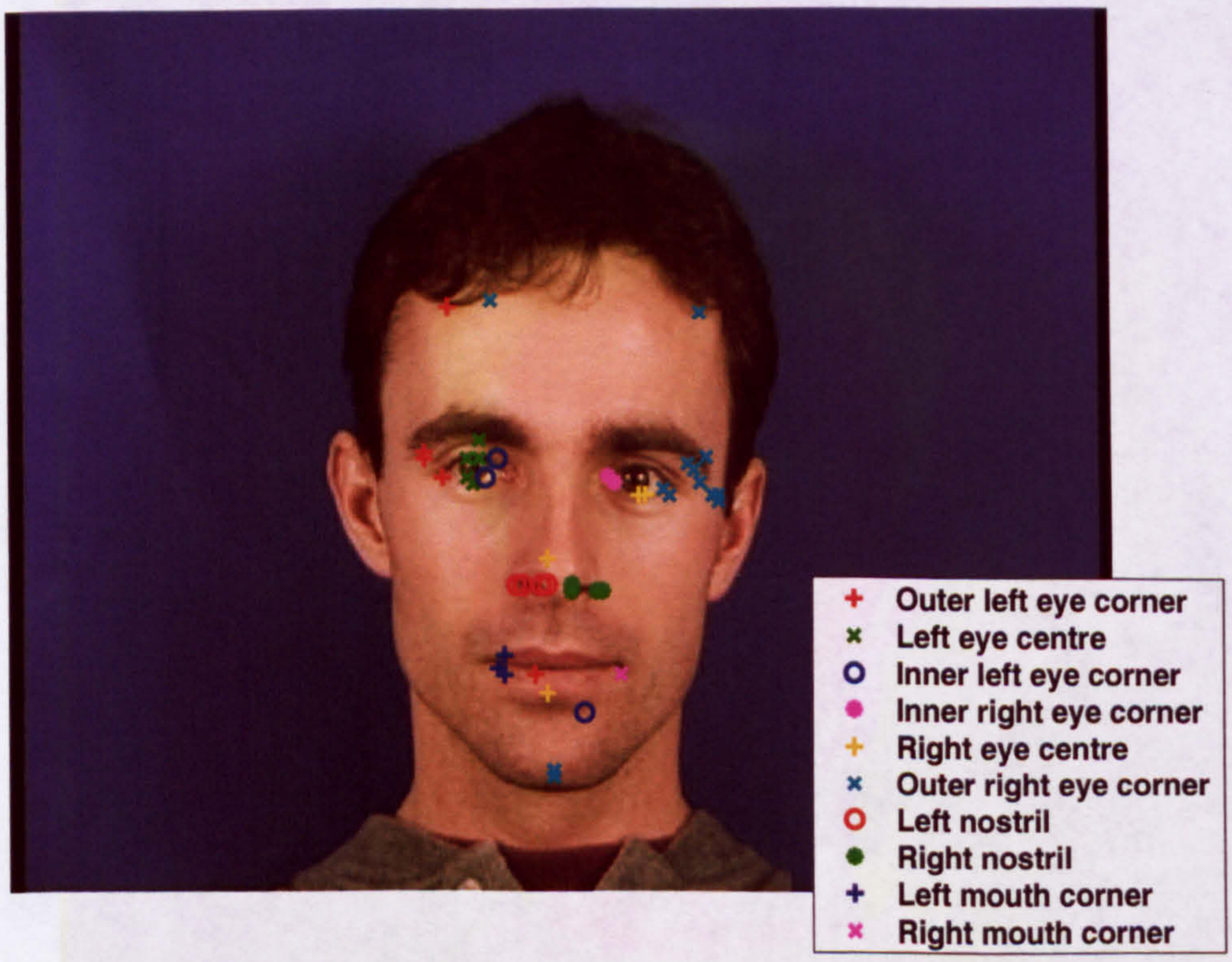

Figure 4.4: Example of feature detection using Harris-and-PCA-based feature detector

4.3 Gabor-filter-based feature detectors

Our early experiments revealed that the originally designed feature detector did not perform well on realistic data. Therefore a different method had to be exploited for feature detection. The solution which will be described here was developed in close cooperation with Joni Kämäräinen from Lappeenranta University in Finland [KKK+02, HKKK03]. In his thesis [Käm03], the author deals in detail with all aspects concerning the use of Gabor filters for object recognition.

While the design of the feature detectors had to be modified, we decided to use the same features as in the original case, leaving some space for comparisons and possible future improvements. We should also note that for these ten features the groundtruth was collected on a big dataset (XM2VTS, BioID and BANCA databases - see Chapter 7), which made the training process easier.

4.3.1 Gabor filters

Time-frequency (in vision space-frequency) analysis plays a major role in signal processing. It is a well established fact that the Fourier transform of a spatially or temporarily extended signal has a little value in analyzing the frequency spectrum in the signal. High frequency peaks cannot be easily identified from the transformed signal. Many task and especially signal detection called for the notion of frequency analysis that is local in time (or space). Windowed Fourier transforms, Gabor filters and wavelets are the main representatives of the local analysis approaches.

Due to their representation power Gabor filters have been previously used in a feature-based face detection and recognition [LVB+93] successfully. An origin centred normalized 2-D Gabor filter is defined as follows:

$$\begin{aligned} \psi(x,y;f,\theta) &= \frac{f^2}{\pi\gamma\eta} e^{-\left(\frac{f^2}{\gamma^2}x'^2+\frac{f^2}{\eta^2}y'^2\right)} e^{j2\pi f x'}, \\ x' &= x\cos\theta + y\sin\theta, \\ y' &= -x\sin\theta + y\cos\theta \ , \end{aligned} \tag{4.1}$$

where f is the frequency of the sinusoid plane wave, θ is the anti-clockwise rotation of the Gaussian envelope and the sinusoid, γ is the spatial width of the filter along the major axis, and η spatial width along the minor axis (perpendicular to the sinusoid).

In practice, applications usually utilize responses from several filters. These filters spread over several frequencies and orientations creating a feature space in which translation, rotation and scaling invariance can be realized [KKK04, Käm03].

Let us demonstrate these invariance properties of the Gabor filter. We will demonstrate translation and scale invariance on 1-D version of the filter, since generalization to 2-D is straightforward.

A normalized version of the 1-D origin centred filter is defined as

$$\psi(t) = \frac{|f|}{\gamma\sqrt{\pi}} e^{-(\frac{f}{\gamma})^2 t^2} e^{j2\pi f t} \tag{4.2}$$

where f is the base frequency and γ is the parameter controlling the sharpness of the filter (its time duration and bandwidth). The response of the filter to 1-D signal S_1 is generated via the convolution:

$$r_{S_1}(t, f) = \psi(t; f) * S_1(t) = \int_{-\infty}^{+\infty} \psi(t - t_\tau; f) S_1(t_\tau) dt_\tau \tag{4.3}$$

For the 1-D Gabor response in Eq. 4.3 and a translated version of the signal S_{1t},

$$S_{1t}(t) = S_1(t - t_1) \tag{4.4}$$

it can be proved that

$$
\begin{aligned}
r_{S_{1t}}(t;f) &= \int_{-\infty}^{+\infty} \psi(t-t_\tau;f) S_{1t}(t_\tau) dt_\tau \\
&= \int_{-\infty}^{+\infty} \psi(t_\tau;f) S_{1t}(t-t_\tau) dt_\tau \\
&= \int_{-\infty}^{+\infty} \psi(t_\tau;f) S_1(t-t_1-t_\tau) dt_\tau \\
&= \int_{-\infty}^{+\infty} \psi((t-t_1)-t_\tau;f) S_1(t_\tau) dt_\tau \\
&= r_{S_1}(t-t_1;f)
\end{aligned}
\tag{4.5}
$$

which shows translation invariance, allowing a translation-invariant signal detection.

It should be noted that the scale invariance holds due to the fact that the frequency and the width of the Gaussian envelope are inversely proportional through parameter γ. This actually guarantees that the filters tuned to different frequencies are scaled versions of each other. Please note that sometimes in Gabor-related studies filters are not defined as normalized and therefore scale-invariance cannot be guaranteed.

For a scaled signal

$$
S_{1s}(t) = S_1(at) \tag{4.6}
$$

it holds for the filter response that

$$
\begin{aligned}
r_{S_{1s}}(t;f) &= \int_{-\infty}^{+\infty} \psi(t-t_\tau;f) S_{1s}(t_\tau) dt_\tau \\
&= \int_{-\infty}^{+\infty} \psi(t_\tau;f) S_{1s}(t-t_\tau) dt_\tau \\
&= \int_{-\infty}^{+\infty} \psi(t_\tau;f) S_1(at-at_\tau) dt_\tau \\
&\hat{t}_\tau = at_\tau \quad dt_\tau = \frac{d\hat{t}_\tau}{a} \\
&\Longrightarrow \int_{-\infty}^{+\infty} \psi(t_\tau;\frac{f}{a}) S_1(at-\hat{t}_\tau) d\hat{t}_\tau \\
&= r_{S_1}(at;\frac{f}{a})
\end{aligned}
\tag{4.7}
$$

The interpretation of this result is clear: a response to a signal is the same as the response of a similarly scaled filter for a scaled version of the signal.

Now since rotation invariance is a 2-D phenomenon we have to switch to two dimensions. A rotated version $S_{2r}(x, y)$ of a 2-D signal $S_2(x, y)$, an image, rotated anti-clockwise around a spatial location (x_0, y_0) by an angle ϕ can be written as

$$\begin{aligned} S_{2r}(x, y) &= S_2(\hat{x}, \hat{y}) \\ \hat{x} &= (x - x_0)\cos\phi + (y - y_0)\sin\phi + x_0 \\ \hat{y} &= -(x - x_0)\sin\phi + (y - y_0)\cos\phi + y_0 \end{aligned} \tag{4.8}$$

The filter response using Eq. 4.8 for the rotated signal is

$$r_{S_{2r}}(x_0, y_0; f, \theta) = \int\int_{-\infty}^{+\infty} \psi(x_0 - x_\tau, y_0 - y_\tau; f, \theta) S_{2r}(x_\tau, y_\tau) dx_\tau dy_\tau \tag{4.9}$$

which can be expressed as

$$\begin{aligned} \int\int_{-\infty}^{+\infty} & \psi([(x_0 - x_\tau)\cos\theta + (y_0 - y_\tau)\sin\theta], [-(x_0 - x_\tau)\sin\theta + (y_0 - y_\tau)\cos\theta]) \\ & S_2([(x_\tau - x_0)\cos\phi + (y_\tau - y_0)\sin\phi + x_0], [-(x_\tau - x_0)\sin\phi + (y_\tau - y_0)\cos\phi + y_0]) \\ & dx_\tau dy_\tau \end{aligned} \tag{4.10}$$

and by changing the integration axes to (x'_τ, y'_τ) which are correspondingly rotated clockwise around the point (x_0, y_0) by the angle ϕ, the following formula is obtained

$$\begin{aligned} & \int\int_{-\infty}^{+\infty} \psi(\hat{x}_\tau, \hat{y}_\tau; f, \theta) S_2(x'_\tau, y'_\tau) dx'_\tau dy'_\tau \\ & \hat{x}_\tau = (x_0 - x'_\tau)\cos(\theta - \phi) + (y_0 - y'_\tau)\sin(\theta - \phi) \\ & \hat{y}_\tau = -(x_0 - x'_\tau)\sin(\theta - \phi) + (y_0 - y'_\tau)\cos(\theta - \phi) \end{aligned} \tag{4.11}$$

and from that it can be derived that the previous form equals to

$$\int\int_{-\infty}^{+\infty} \psi(x_0 - x'_\tau, y_0 - y'_\tau; f, \theta - \phi) S_2(x'_\tau, y'_\tau) dx'_\tau dy'_\tau \tag{4.12}$$
$$= r_{S_2}(x_0, y_0; f, \theta - \phi)$$

Finally, taking all three invariance properties of 2-D normalized filter together, it can be proven that for a 2-D signal $S'_2(x, y)$ which is signal $S_2(x, y)$ translated from a location (x_0, y_0) to a location (x_1, y_1), scaled by a factor a and rotated anti-clockwise by an angle ϕ around the location (x_1, y_1) it holds that

$$r_{S'_2}(x_1, y_1; f, \theta) = r_{S_2}(x_0, y_0; \frac{f}{a}, \theta - \phi) \tag{4.13}$$

which is directly exploitable for the purpose of translation, scale and rotation invariant feature detection [Käm03].

Gabor filters are usually the first step in processing and refining data to extract informative features. For the purpose of feature detection, discrete versions of Gabor filters are used. As mentioned above the most common applications involve a combination of responses from several filters (so called filter bank). However, as the number of filters increases, the computational costs will also increase, therefore this trade-off has to be considered in design. The statistical features (filter responses) obtained by applying the Gabor filter bank to the 2-D signal (in our case image) are defined by a finite sampling grid of parameters of the filter, i.e. spatial coordinates $(x, y)_i$, frequencies f_j, and orientations θ_k. 2-D signal is therefore represented by these features as a grid in four-dimensional space [Käm03].

If objects are to be distinguished using the responses at a single spatial location, a translation invariant search can be performed by inspecting the responses at each location. This approach has been used in many studies. To decrease the computational cost, non-uniform sampling schemes have been proposed.

The statistical features (not to be confused with facial features) that consist of Gabor filter responses at each position are calculated by the use of the convolution

$$r_\xi(x,y;f,\theta) = \psi(x,y;f,\theta) * \xi(x,y) \\ = \iint_{-\infty}^{\infty} \psi(x-x_\tau, y-y_\tau; f,\theta)\xi(x_\tau,y_\tau)dx_\tau dy_\tau \tag{4.14}$$

where $\psi(x,y;f,\theta)$ is a 2D Gabor filter and $\xi(x,y)$ an input image. A response matrix in a single spatial location (x_0,y_0) can be constructed by calculating filter responses for a finite set of different frequencies f and orientations θ as

$$\mathbf{G} = \begin{pmatrix} r(x_0,y_0;f_0,\theta_0) & r(x_0,y_0;f_0,\theta_1) & \cdots & r(x_0,y_0;f_0,\theta_{n-1}) \\ r(x_0,y_0;f_1,\theta_0) & r(x_0,y_0;f_1,\theta_1) & \cdots & r(x_0,y_0;f_1,\theta_{n-1}) \\ \vdots & \vdots & \ddots & \vdots \\ r(x_0,y_0;f_{m-1},\theta_0) & r(x_0,y_0;f_{m-1},\theta_1) & \cdots & r(x_0,y_0;f_{m-1},\theta_{n-1}) \end{pmatrix} \tag{4.15}$$

In order to achieve illumination invariance, the feature matrix $\mathbf{G}$ in Eq. (4.15) is normalized as

$$\mathbf{G}' = \frac{\mathbf{G}}{\sqrt{\sum_{i,j}|g_{i,j}|^2}} \tag{4.16}$$

This operation effectively normalizes the energy of the filter response (feature matrix) to a constant. It exploits a simple linear-illumination model which is justifiable in the case of small objects where big cast shadows do not occur. We believe that the issue of illumination is one of the main advantages supporting the use of feature detectors. Unfortunately, this normalization also presents pitfalls since it also emphasizes areas of low response (amplifying noise), such as backgrounds, and typically induces an increasing number of false alarms. In some situations it might be desirable to remove the effect of low responses as suggested in [LVB+93].

4.3.2 Rotation and scale invariance

In order to search for facial features in different orientations and scales, simple matrix shift operations can be used [KKK04]. A column-wise rotation of the feature matrix can be defined as

$$\mathbf{G}^{(k)} = \Big(\mathbf{G}(1:m, k:n) \quad \mathbf{G}(1:m, 1:k-1)\Big) \tag{4.17}$$

where $\mathbf{G}^{(k)}$ denotes k-columns shifted matrix $\mathbf{G}$, $\mathbf{G}(i:j, u:v)$ represents the sub-matrix of $\mathbf{G}$ containing rows $i \ldots j$ and columns $u \ldots v$. Similarly a row-wise shift for scale manipulation can be defined as

$$\mathbf{G}_{(k)} = \begin{pmatrix} \mathbf{G}(k+1:m, 1:n) \\ \mathbf{G}(m+1:m+k, 1:n) \end{pmatrix} \tag{4.18}$$

The column-wise circular shift in Eq. (4.17) corresponds to searching over all rotation angles. It should be noted that the shift is circular and if the responses are calculated only for half the space, e.g. $[0, \pi]$, it has to be taken into account and the responses converted accordingly (by a complex conjugate). The row-wise shift in Eq. (4.18) corresponds to searching over all larger scales. This shift is not circular but the highest frequencies vanish as the filter is scaled up and new lower frequencies are mapped to the Gabor-feature matrix $\mathbf{G}$ as replacements.

By extracting Gabor filter responses using the feature matrix in Eq. (4.15), normalizing features by Eq. (4.16), searching features in different poses using the matrix shifts in Eqs. (4.17) and (4.18), translation, illumination, rotation and scale invariance can be achieved. Please remember that all response values are complex numbers. In our experiments we also tested real-valued responses (i.e. magnitude of the complex filter responses), however it proved to be insufficient for good performance. Complex valued responses carry the phase information which is crucial for the discriminative abilities of the detector [KKK+02].

The complex-valued Gabor feature matrix in Eq. (4.15) can be used to distinguish between different facial features. However such scheme would be useful only for the detection of features of one particular person. In face detection/localization we are dealing with object-class recognition and as mentioned above even simple face features are visually quite variable. Therefore a statistical model based on the feature matrix, $\mathbf{G}$,

has to be created. The Gabor-filter filter theory provides us with a rigorous approach dealing with invariance, but it does not model the statistical appearance variation of the object (feature) itself. Next sections will introduce two statistical models successfully used in our scheme. Examples of 2-D Gabor filters of different f_0 and θ are depicted in Figure 4.5

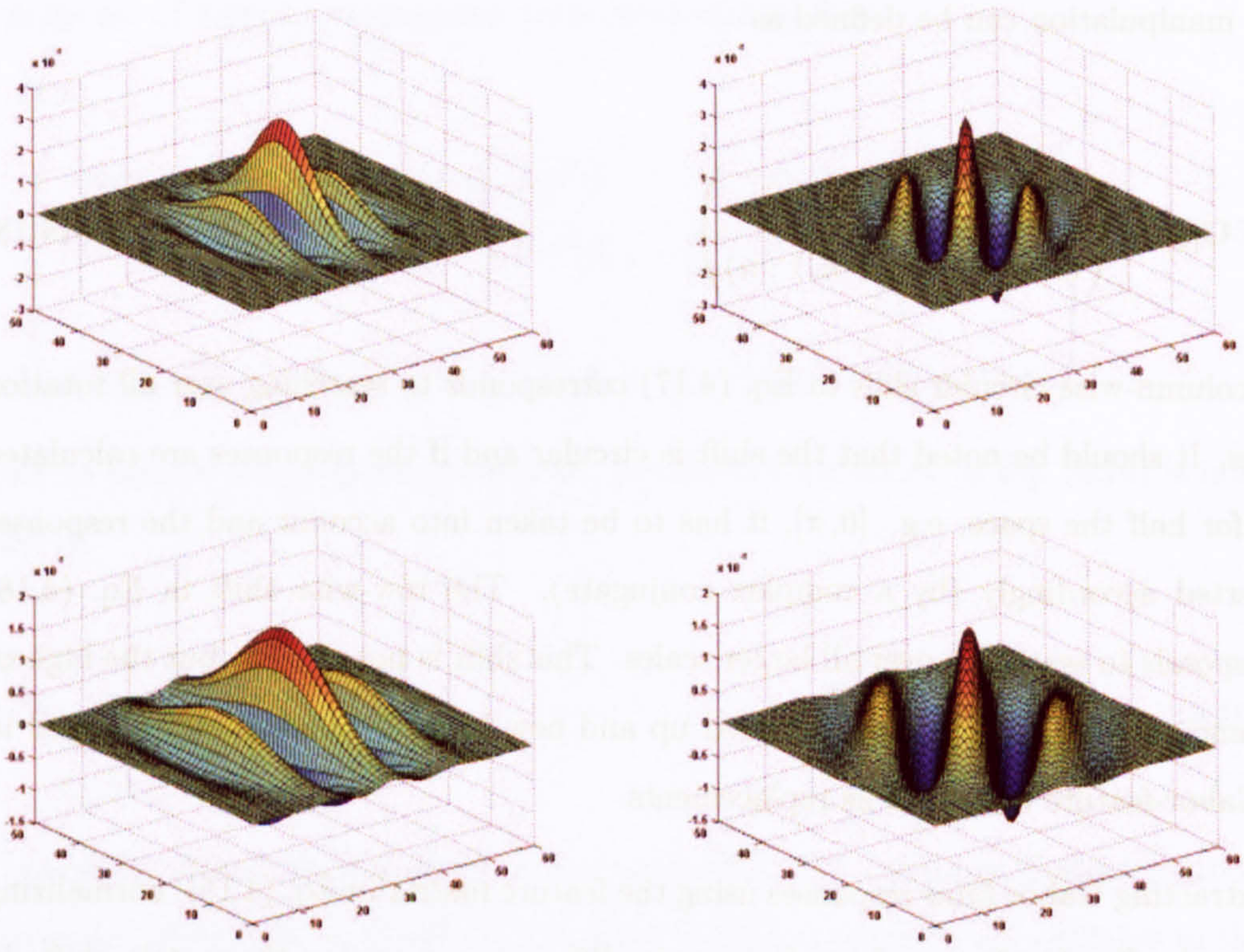

Figure 4.5: 2-D Gabor filters (real parts): rows depict different θ, columns different f_0

4.3.3 Sub-cluster classifier

If we compute features (feature matrices $\mathbf{G}$) for the ten facial features over a large number of faces of same scale and orientation we can learn their statistical distribution. Moreover as shown above, this representation gives us rigorous means how to achieve geometric invariance. The facial feature responses $\mathbf{G}$ form specific clouds in the feature space (if we perceive $\mathbf{G}$ as a vector). It is desirable that these clouds are separable from each other by some means of classification. We have opted for a cluster-based

method called sub-cluster classifier [KKK+02, HKKK03]. The name reflects the fact, that a cluster of a specific face feature responses (e.g. eye centre) can consist of several subclusters, signifying, that there are several groups of facial feature appearance among population. Also, for example, closed or open eyes are visually different, therefore it is very likely that there will be different clusters representing them. However we still want to detect an eye-centre regardless of which person it is or if the eye is closed or not, therefore the eye-feature cluster could possibly include several different subclusters.

Let us stress again that elements of the feature matrix **G** are complex numbers. A complex Gaussian distribution was previously studied in the literature and used in several applications [Goo63]. We performed an experiment comparing the distributions of feature vectors created by concatenating real and imaginary parts with that of the complex vectors. These two representations performed comparably and we chose the complex variant, because it actually has a lower dimensionality.

The pseudo-code of the subcluster-classifier is presented in Algorithms 1 and 2.

The C-means step (also referred to as K-means algorithm) in the training ensures that a good unsupervised (i.e. feature labels are not used) partitioning supported by the data is obtained. The partitioning produced by this algorithm is called Voronoi tessellation. The C-means algorithm converges fast, but it is sensitive to the initialization. If the generated partitioning produces clusters with a dominant majority of a single class then further steps follow, i.e. if each class is covered by at least one cluster, samples are reclassified to the closest cluster of correct-label and true estimated means and covariances returned. If no good partitioning is produced, the C-means is run again. As the results will show (see section 7.6) this algorithm performs very well for the intended purpose, however it should be regarded more as an engineering solution, than a theoretical contribution. To have a better theoretical rationale, we decided to replace the sub-cluster classifier (SCC) by the Bayesian classifier assuming Gaussian mixture model (GMM) probability densities, which will be described in the next section.

TRAINING PHASE

Data : Scale and orientation normalized faces with manually annotated features to detect, Nc = number of clusters, k = number of filter scales, l = number of filter orientations, CL = how many % of a single class has a cluster to contain
Result: Trained classifier =
set of triplets $\{T_i = (FeatureLabel_i, \delta_i, \Sigma_i), i \leq Nc\}$
for *All the face images* **do**
  Compute the illumination normalized feature matrix $\mathbf{G}(f_0, ...f_{k-1}; \theta_0, ..., \theta_{l-1})$ for all ten facial features as the Gabor filter-bank responses at the manually allocated feature locations and create a vector $\mathbf{g}$ by concatenating the columns of the matrix $\mathbf{G}$;
  Assign a feature label $L \in \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ to each response $\mathbf{g}$ according the feature type;
end
while $NOT(STOP)$ **do**
  Run C-means algorithm on the responses $\mathbf{g}$ with Nc clusters initialized by a random choice among $\mathbf{g}$;
  Assign each response $\mathbf{g}$ to a closest cluster mean;
  Compute how many percent of each cluster are features from a single class, then for each cluster choose the highest value and assign the feature label to the cluster accordingly (i.e. choose the class that most represents the given cluster);
  Throw away clusters that contain less than CL% of samples from a single class;
  if *Each class is represented at least by one cluster* **then**
    $STOP$=true;
  end
  else
    $STOP$=false;
    decrease value CL;
  end
end
Assign each sample of label q to the nearest cluster (using its mean) which has cluster label also q (cluster label is determined by the majority class label in the cluster);
Compute cluster sample means δ_i and sample covariance matrices Σ_i and create a triplet for each cluster $T_i = (FeatureLabel_i, \delta_i, \Sigma_i), i = \{1, ..., N\}$, where $FeatureLabel_i$ corresponds to the class label which had the highest occurrence in the cluster (see above) and N is the actual number of clusters;

Algorithm 1: Learning phase using SCC

DETECTION PHASE

Data : Test image I containing faces, N_{max} = how many best detections per class to output, SI = number of scale invariance steps, RI = number influencing the rotation invariance ($RI = 0$ means that no rotation invariance test will be performed), SCC classifier = $\{T_i = (FeatureLabel_i, \delta_i, \Sigma_i), i = \{1, ..., N_c\}\}$, d_{min} = minimum distance between detected features of the same class in pixels

Result: List of detected features ordered according rank in each class

for *All pixel positions* (x_0, y_0) *in* I **do**
  Compute illumination normalized feature response matrix $\mathbf{G}(f_0, ...f_{k+SI-1}; \theta_0, ..., \theta_{l-1})$ for each location (x_0, y_0) (k, l being the original parameters of the Gabor filter bank used in training);
  for *scale=1:SC* **do**
    for *rotation*$=-RI : RI$ **do**
      Compute $\mathbf{G}'$ as $\mathbf{G}(f_{scale-1}, ..., f_{k+scale-2}; \theta_{rotation}, ..., \theta_{rotation+l-1})$ (i.e. the rotated and scaled version of $\mathbf{G}$ utilising row and column shifts);
      Create $\mathbf{g}'$ as the concatenation of columns of $\mathbf{G}'$;
      for *cluster* $i = 1 : N_c$ **do**
        compute $DIST(i, scale, rotation) = (\mathbf{g}' - \delta_i)\Sigma^{-1}(\mathbf{g}' - \delta_i)^T$;
      end
    end
  end
  For the position (x_0, y_0) assign the class label according the decision rule $L = clusterlabel(argmin(DIST(i, scale, rotation)))$ and the $score = min(DIST)$;
  Record detected feature as quadruplet $(x_0, y_0, L, score)$
end
Perform non-minima suppression on the detected points of class with minimum distance $d_{min}[pixels]$;
Return the best N_{max} detected features for each class based on the value of *score*;

Algorithm 2: Detection phase using SCC

4.3.4 Gaussian mixture model

The Bayesian classification with GMM probability densities and the estimation of GMM parameters have been well covered in the pattern recognition literature. For the given features an unsupervised estimation of GMM is preferred and several different methods have been proposed, e.g., a greedy EM [VL02]. However, a method proposed by Figueiredo and Jain [FJ02] provided the best convergence properties and classification results in the experiments conducted. This method is capable of selecting the number of components automatically and does not require careful initialization. The Expectation-Maximization (EM) algorithm [DLR] is used to minimize a novel score function. The score function is derived using the Minimum Message Length (MML) criterion and is defined as follows:

$$\mathcal{L}(\theta, \mathcal{Y}) = \frac{N}{2} \sum_{m:\alpha_m>0} \log\left(\frac{n\alpha_m}{12}\right) + \frac{k_{nz}}{2} \log\frac{n}{12} + \frac{k_{nz}(N+1)}{2} - \log p(\mathcal{Y}|\theta) \tag{4.19}$$

where θ is the complete set of parameters needed to specify the mixture (in our case means, covariance matrices and mixing probabilities), $\mathcal{Y}$ are the data samples, k_{nz} denote the number of non-zero probability components, α_m are the mixing probabilities, N the number of parameters specifying each component, n the number of data samples, and $\log p(\mathcal{Y}|\theta)$ the log-likelihood. The algorithm starts with a large number of k_{nz} and uses the EM algorithm to minimize $\mathcal{L}(\theta, \mathcal{Y})$. As a part of M-step, components that are too weak (i.e. unsupported by data) are annihilated. The optimal solution corresponds to the minimum of $\mathcal{L}(\theta, \mathcal{Y})$.

Similar to the SCC case the posterior probabilities computed by the Bayesian classifier can directly be used as confidence values to sort spatial coordinates into the best ranking order for each facial feature. The algorithm for detection is identical with Algorithm 2, only the $DIST$ is replaced by the posterior probability given by the GMM model and minimization is replaced by maximization.

In our latest setup we output 200 best candidates per feature. Detailed performance evaluation and comparison of the two classifiers used will be presented in chapter 7.

Typical results of feature detection using GMM-classifier and feature matrix **G** with 4 scales and 5 orientations are depicted in Figure 4.6.

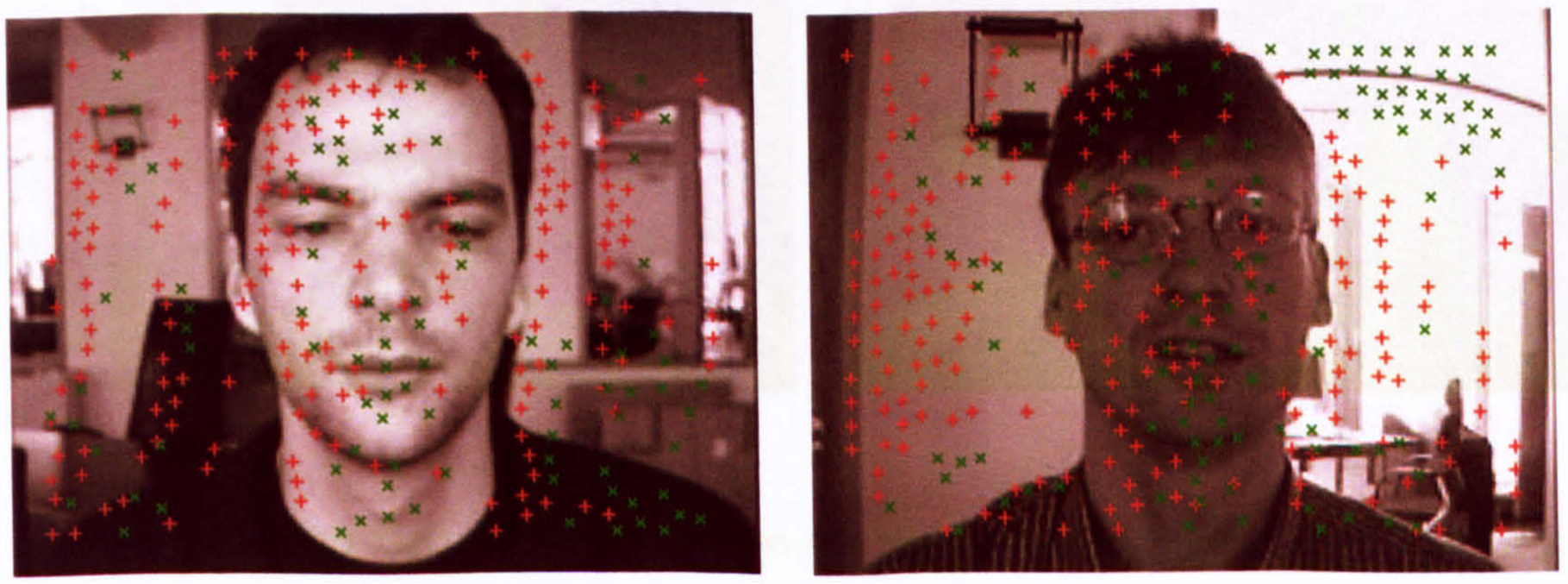

Figure 4.6: Typical result of feature detection, + denotes outer left eye corner, x denotes left eye centre

Figure 4.6: (continued) + denotes inner left eye corner, x denotes inner right eye corner

4.4 Summary

In this chapter the problem of feature detection was discussed. We introduced a novel Gabor-filter-based feature detector that attempts to model facial features in a translation, scale and orientation invariant way. Utilizing cluster-like properties of the data, two cluster-based classifiers were developed and tuned for the purpose of classification

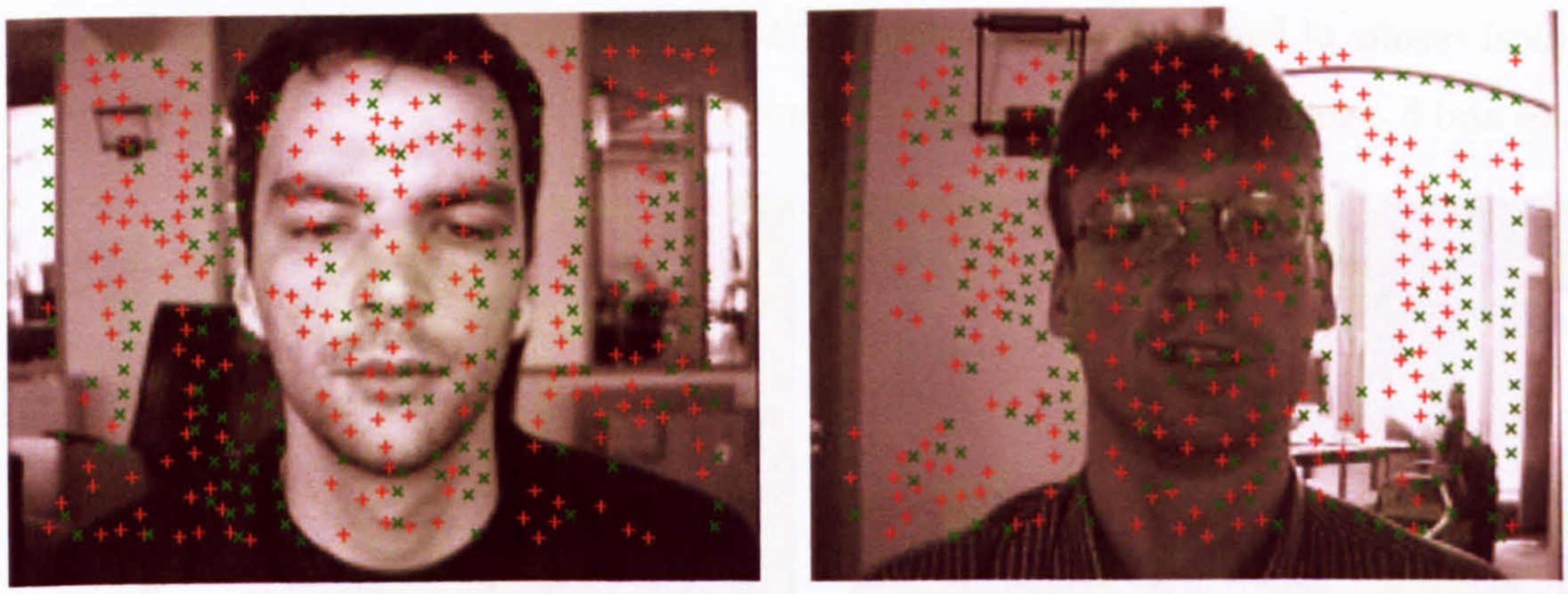

Figure 4.6: (continued) + denotes right eye centre, x denotes outer right eye corner

Figure 4.6: (continued) + denotes left nostril, x denotes right nostril

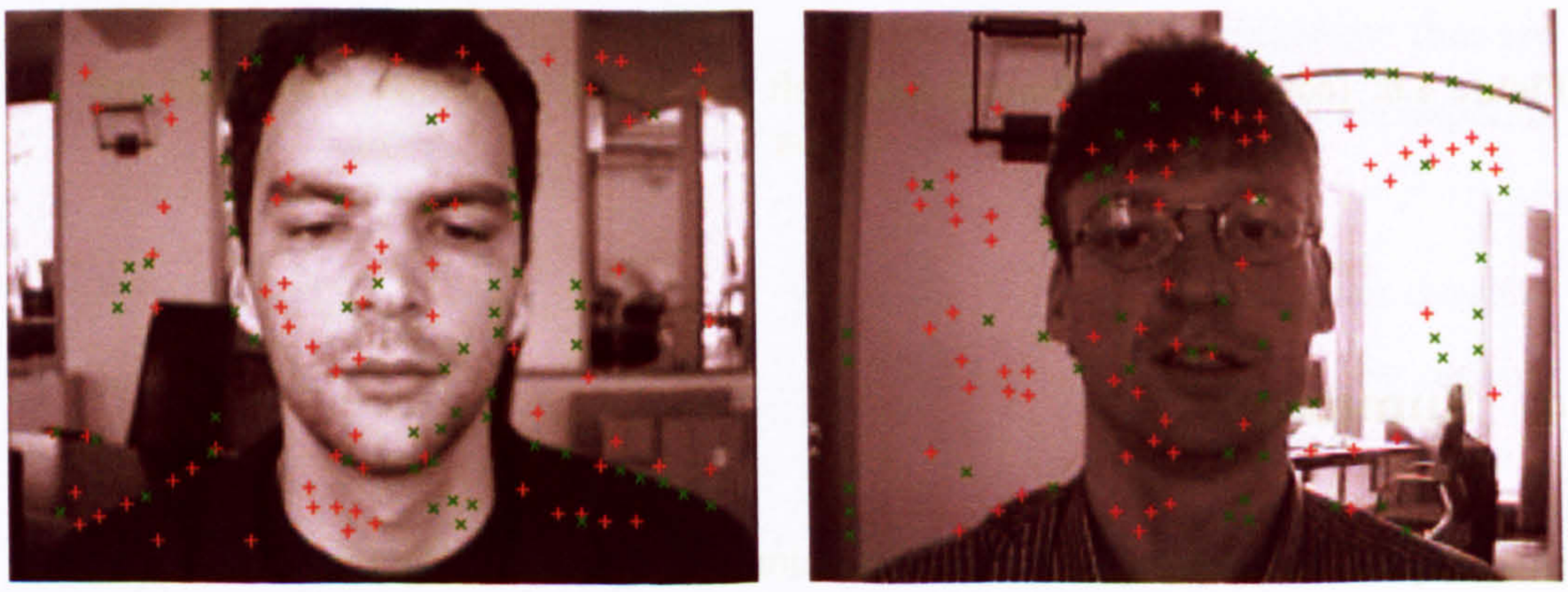

Figure 4.6: (continued) + denotes left mouth corner, x denotes right mouth corner

in the Gabor-response space. We also found, that the phase information is very important and therefore complex-valued models were used (in contrast to magnitude models). By outputting N best feature candidates, the success rate can be maintained even in the presence of the background, however false alarms have to expected as an inherent part of the output. The next chapters will introduce methods developed to exploit the detected features for finding the true face location.

Chapter 5

Transformation model

In this chapter, our approach to constellation (transformation) modelling will be described. A transformation (constellation) model is used early in the hypotheses generator (see Figure 3.1) to produce admissible-constellation triplets of detected features efficiently and to register the corresponding image patch for further stages of the algorithm.

5.1 Definition of the face space

As mentioned in section 3.3 and 4, face space was introduced in order to reduce the variability inflicted by scene capture and facial shape. In our design this space is linear and in the two dimensional image space it is represented by three landmark points positioned on the face. The particular choice of these three points was carried out by using optimised search among various linear combinations of groundtruth landmark points on the face. The XM2VTS database (see section 7.2) was used for the experiment. As a criterion, a photometric variance computed with the help of Principal Component Analysis (PCA) over the set of facial images was used. The triplet of points corresponding to the minimum photometric variance was then taken as a coordinate system. The idea for this comes from a scientific hypothesis, that if faces are photometrically correlated, it is likely, that they are properly aligned. In particular the following points were chosen and subsequently used in our experiments:

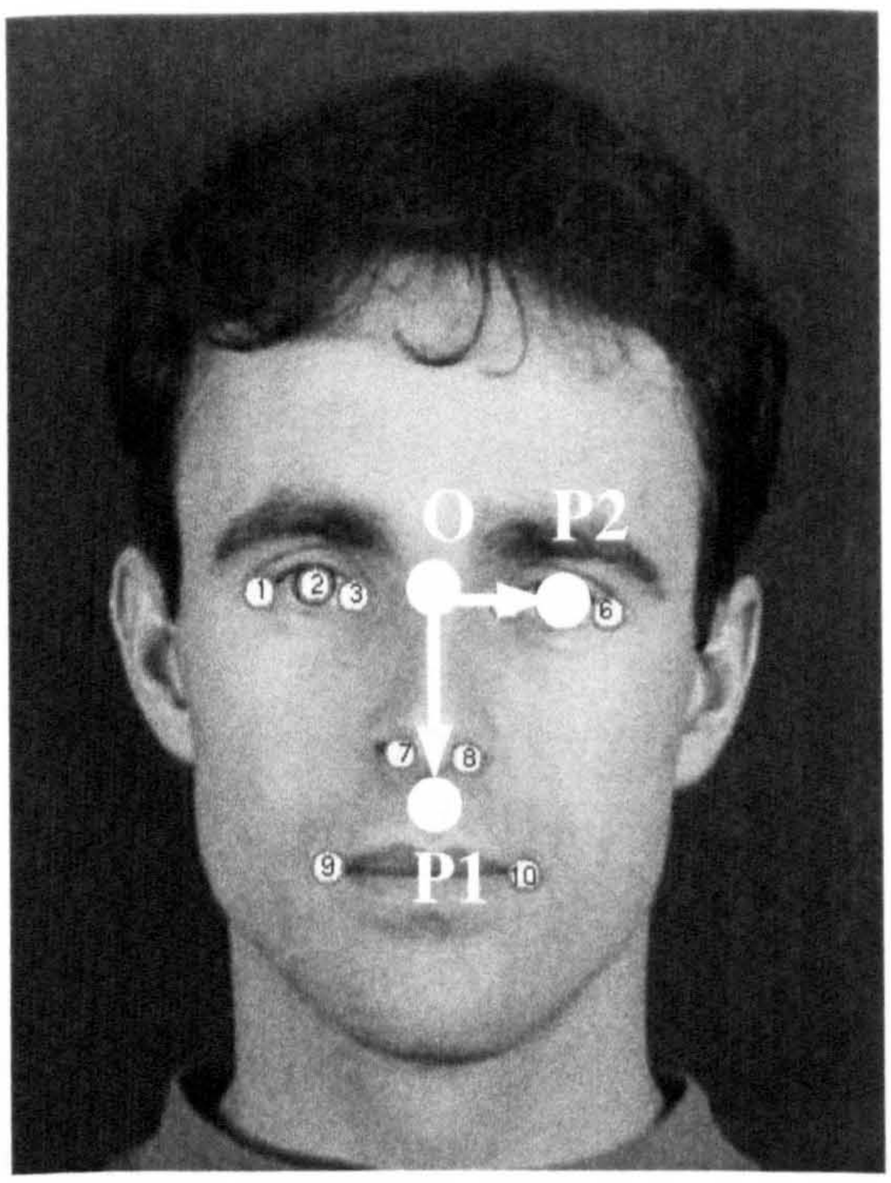

Figure 5.1: Face space definition

1. Mean value of eye centres and eye corners taken as an origin O.

2. Mean value of nostrils and mouth corners taken as P1.

3. Mean value of the coordinates of the right eye corners and right eye centre taken as P2.

It is worth mentioning that since other combinations of points were close to the optimum of the criterion function, a different choice of points could lead to similar final results. Also the availability and type of manually registered data (landmark points manually defined on the face) played a certain role in the selection. However, it is obvious that this particular choice manages to normalize the height and width of the face and this is in accordance with our intuitive feeling about the problem.

Once given three coordinates (O,P1,P2) on a face, that face can be registered (warped) into this space. The new coordinate axes are then $O \rightarrow P2$ and $O \rightarrow P1$. Affine warping of the input image is needed to transform the image (carrying the face) into the face space. In this space most of the shape variability and main scene capture effects

(like scale, orientation, translation) are removed. In such space, faces become photometrically tightly correlated and modelling of face/background appearance is therefore more effective. But what is the use of detected features in the image with regard to this space? As mentioned in the previous chapter, if we know that certain features appear repeatedly in the same place on a face they will have a small position variation of their face-space coordinates. Below we show that by using three of such spatially stable features gives us an opportunity to estimate the transformation that the whole face has to undergo in order to appear in the face space. By transforming an instance of a face in the image into the face space the scene capture effects like scale, orientation, etc. can be removed and appearance of a geometrically aligned face checked. The transformation could be established by using correspondences between features (points) detected in the image and face space feature coordinates (which can be represented by a single point for all faces, due to their small position variance). Very importantly, the transformation can also be used as a clue to support the decision whether there is a face or not, since non-facial points (false alarms) will result in hypothesised transformations that have not been encountered in the training set and this can easily be identified.

5.2 Affine transformation

The transformation defining the mapping from image space to face space which was just described above can be categorized as affine. Affine transformation from two-dimensional image space into two-dimensional face space is represented by six parameters:

$$T : \mathcal{R}^2 \mapsto \mathcal{R}^2$$

$$\begin{pmatrix} x_{IS} \\ y_{IS} \\ 1 \end{pmatrix} = \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} x_{FS} \\ y_{FS} \\ 1 \end{pmatrix} \tag{5.1}$$

The equation expresses the relationship between face space coordinates and image

space coordinates. (x_{FS}, y_{FS}) denotes face space coordinates and (x_{IS}, y_{IS}) image space coordinates. Please note that homogeneous coordinates are used in order to treat translation as a linear transformation. In the case of face registration described above the face-space coordinates are $(0,0) \mapsto O$, $(0,1) \mapsto P1$ and $(1,0) \mapsto P2$.

The registration aims to remove the geometric facial variability as well as conditions encountered during capture (distance from the camera, angle, etc.). We can also look at it in such a way that a face from the face-space can undergo a certain affine transformation in order to appear in the image. Let us explain this in a greater detail. We have a 2D coordinate system in which we map all the training set faces using three predefined landmark points on the face. In general many landmark points could be used to register the face [ETC98] and even 3D model be used. The actual mapping from image coordinates to face space coordinates is affine in our case. Then when we wish to localize an unknown face in the image using our approach, we need to register the face from the image space back into face space for the geometrically normalized appearance test. The registration is carried out using the detected features, which have to uniquely define the transformation. Every transformation has a certain number of degrees of freedom. In the case of the full affine transformation, the number is six. It means that at least three different points need to be detected on the facial image in order to achieve a successful registration through correspondences. Generally, there is a trade-off between the transformation complexity and speed, the more complex transformation the more landmark points are needed for registration.

As mentioned earlier, the choice of affine transformation sufficiently approximates the nature of the authentication scenarios scenes, since the human face is quite flat apart from the nose. It means that when 3D effects are not involved, i.e. the face is frontal or near frontal and is not extremely close to the camera, imaging effects can be to a great extent captured by affine transformations. This choice at the same time makes the detection process computationally feasible, since only three points are needed for the detection/localization of the face. Moreover, the inaccuracies caused by this approximation can be dealt with in the appearance model. Thus it can be argued that the choice of the transformation is made without any loss of generality. The schema could be extended to more general situations (which we do not explore in this thesis),

e.g. using perspective transformation or a three-dimensional model. Another possibility would be view-based modelling consisting of a set of two-dimensional affine models. Since real-life authentication scenes involve clients sitting or standing in front of a camera the choice of the affine transformation is appropriate.

Affine transformation preserves collinearity and ratios of distances. It can be decomposed into simpler representations, where primitive operations can be identified. There exist infinitely many decompositions, but only some of them provide an intuitive feel for the given mapping. We used the following decomposition:

$$\begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r & 0 & 0 \\ 0 & r & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\phi & \sin\phi & t_x \\ -\sin\phi & \cos\phi & t_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} t & 0 & 0 \\ 0 & \frac{1}{t} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & n & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}^R \tag{5.2}$$

where R is reflection (either 0 or 1), n shear, t squeeze, ϕ rotation, r scale and t_x, t_y translation.

5.3 Transformation modelling

When the transformation from the image space into the face space (denoted as T^{-1}) can be uniquely determined, it is possible to obtain a statistical distribution of this transformation (or its inverse as shown below) over a large number of facial images. What does this distribution represent? It tells us a lot about the geometrical facial variance and also what the capture conditions are. Some of the parameters of the transformation will obviously depend mainly on the camera capture setup (like scale, orientation and translation), others like shear and squeeze are to a great extent influenced by the generic facial variability itself. We can learn what the typical transformation is and use this during the detection. This is one of the main points of this approach.

5.3.1 Correspondences between detected and model features

As described in chapter 4, in our algorithm we aim to detect ten facial features. In the face space these features have indeed a small positional variance as demonstrated in figure 5.3, therefore mean positions can be taken as the feature face space coordinates, without introducing a big error. This enables us to perform a quick correspondence-based registration of the underlying photometric information into the face space. For this operation, the transformation from image space into face space has to be computed.

Since affine transformation (see section 5.2) is defined by six parameters, three detected features provide a solution. However not all triplets of features are suitable for the estimation of the transformation. Triplets which would lead to a degenerate solution or to a transformation that is highly sensitive to localization errors have to be excluded. The condition number (ratio of the biggest and the smallest singular value) of the matrix made up by putting the homogeneous face space coordinates of features as columns was used to determine the well-posedness of triplets. In short, the condition number determines how precisely a system of linear equations can be solved using a given matrix. The total number of all possible combinations of three features is $\binom{3}{10} = 120$. 58 out of these were taken as satisfactory with regard to the condition number (their condition number was below a preselected threshold). The well-posed triplets are depicted in figure 5.2.

Given a triplet of detected features, the transformation from face-space into image-space can be estimated as follows:

$$\widehat{\mathcal{T}}_{TRIPLET(f1,f2,f3)} \quad : \quad \mathcal{FS} \mapsto \mathcal{IS}$$

$$\widehat{\mathcal{T}}_{TRIPLET(f1,f2,f3)} = \begin{pmatrix} f1_{ISx} & f2_{ISx} & f3_{ISx} \\ f1_{ISy} & f2_{ISy} & f3_{ISy} \\ 1 & 1 & 1 \end{pmatrix} * \begin{pmatrix} f1_{FSx} & f2_{FSx} & f3_{FSx} \\ f1_{FSy} & f2_{FSy} & f3_{FSy} \\ 1 & 1 & 1 \end{pmatrix}^{-1} \quad (5.3)$$

Note that we denote $\widehat{T}$ as the estimate of transformation T since $\widehat{T}$ is computed using detected features, not the points (O,P1,P2) determined as the combination of the

Figure 5.2: Well-posed triplets selected according to their condition number

groundtruth (manually annotated points on the face). As $\widehat{T}^{-1}$ is determined by the inverse of the matrix representing $\widehat{T}$, it is not particularly important if we choose to model $\widehat{T}$ or $\widehat{T}^{-1}$, since these two transformations can be quickly computed from each other. The use of mean values of the groundtruth feature coordinates mapped into the face-space computed over the training set was used as face space coordinates. As discussed above, due to a small fluctuation of the positions, this introduces a certain alignment error which the appearance model has to deal with.

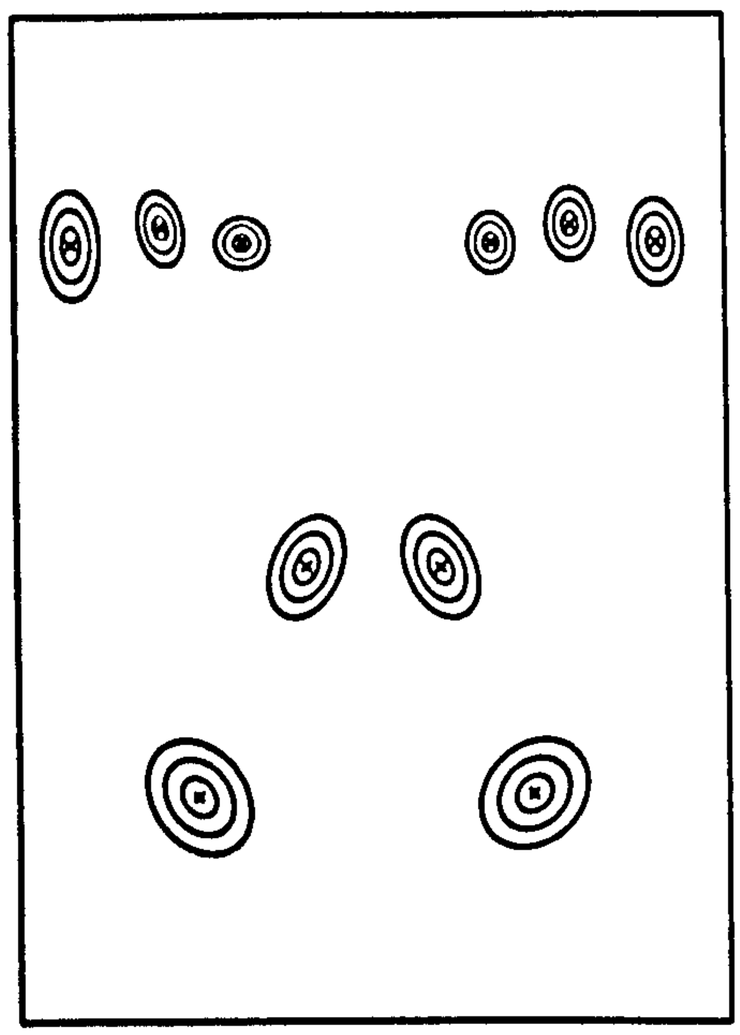

Figure 5.3: Position deviation of the ten detected features when registered in the face space. The sample means of the positions were taken as face-space coordinates of the features

5.3.2 Statistical model

In figures 5.4 the histogram of four transformation parameters of $\widehat{T}$ obtained on the BANCA database is displayed. The distribution was estimated using groundtruth positions of ten detected features and consequent computation (and decomposition) of the transformation between face-space and image-space coordinates. It is clear, that shear and squeeze are well represented by Gaussian distributions. Since scale and rotation (representing the size and rotation of face in the image space) depend mainly on imaging conditions, it would not be wise to use probabilistic models to represent

them. Otherwise imaging conditions encountered in the training set would have to be in exact accordance with imaging conditions in the test set and this would be too restrictive. In order to reflect the fact, a binary decision function with output (0,1) depending whether the value lies in a predefined interval, is used in the model. Such explicit modelling of transformation parameters may be useful in applications and is certainly useful in the authentication scenario. It is easy to make the system ignore small or extremely tilted faces by manually setting the admissible interval of values.

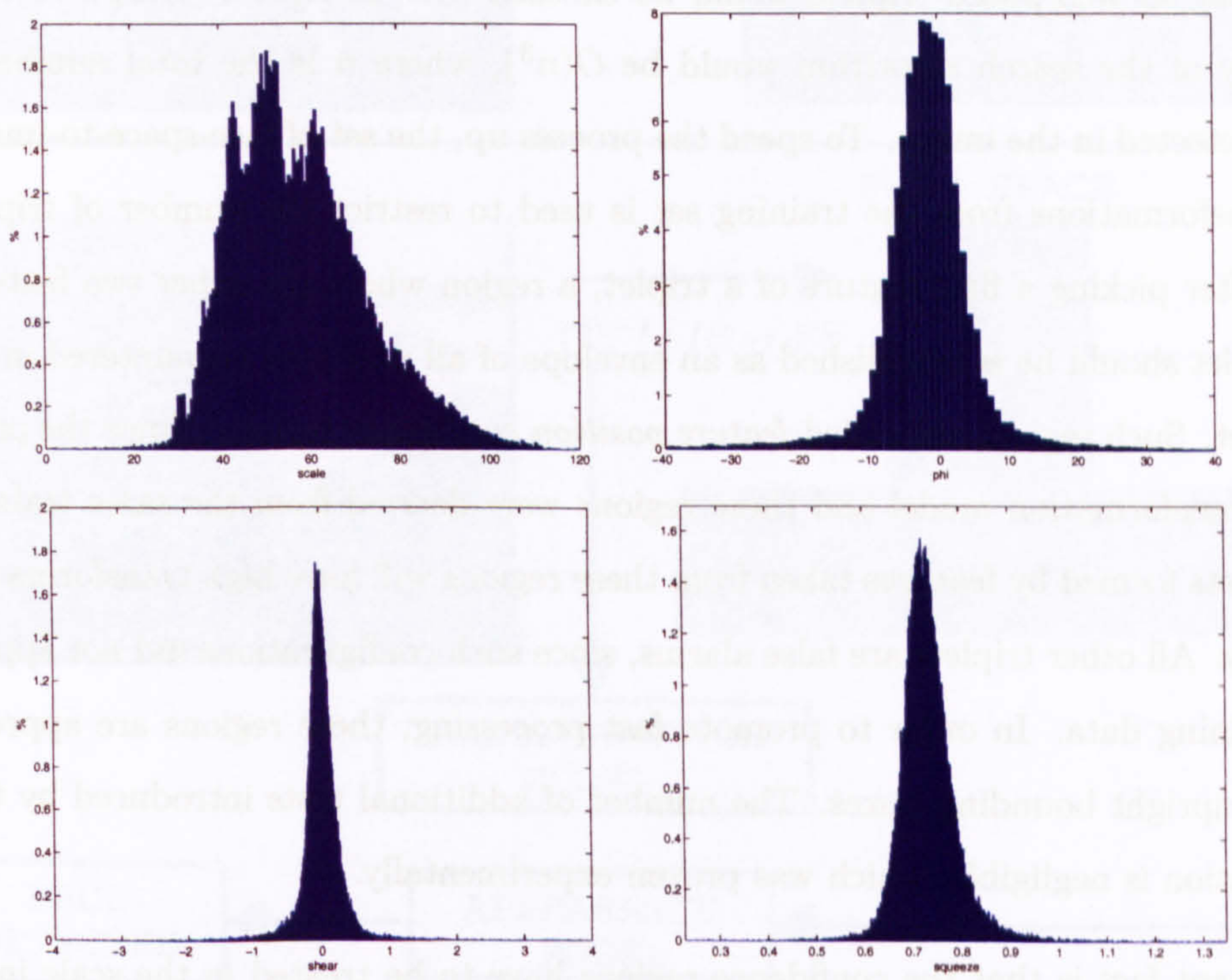

Figure 5.4: Histograms of transformation parameters over BANCA database world-model images – From left to right, top to bottom: scale, ϕ, shear and squeeze

The probability of a given transformation T can then be expressed as:

$$p(\hat{T}) \approx p(n) \cdot p(t) \cdot BOOL(r \in [r_1, r_2]) \cdot BOOL(\phi \in [\phi_1, \phi_2]) \tag{5.4}$$

where $p(n), p(t)$ are modelled by a normal distribution and $BOOL$ denotes a Boolean function the output of which is $\{0, 1\}$ depending on the truth value of its argument. When r falls in the interval $[r_1, r_2]$ the value of B is 1, otherwise 0 and the same holds

for ϕ. Parameters n, t, ϕ and r are the decomposition parameters from Eq. 5.2 and intervals $[r_1, r_2]$ and $[\phi_1, \phi_2]$ are the bounding intervals of r and ϕ.

The diagram of the whole detection process can be found in Figure 5.5.

5.4 Confidence regions

In case that all well-posed triplets would be checked (i.e. at least $\widehat{T}$ computed), the complexity of the search algorithm would be $O(n^3)$, where n is the total number of features detected in the image. To speed the process up, the set of face-space-to-image-space transformations from the training set is used to restrict the number of triplets tested. After picking a first feature of a triplet, a region where the other two features of the triplet should lie is established as an envelope of all positions encountered in the training set. Such regions are called *feature position confidence regions.* Since the probabilistic transformation model and these regions were derived from the same training data, triplets formed by features taken from these regions will have high transformation probability. All other triplets are false alarms, since such configurations did not appear in the training data. In order to promote fast processing, these regions are approximated by upright bounding boxes. The number of additional tests introduced by this approximation is negligible, which was proven experimentally.

An important fact is that the confidence regions have to be treated in the scale independent manner, otherwise the training set scales would have to exactly correspond to the test set scales. Scale-invariance can be achieved by normalizing the transformation $\widehat{T}$ by an inverse isotropic-scale matrix as shown in Eq. 5.5.

$$\widehat{T}_{norm} = \widehat{T} \cdot \begin{pmatrix} r^{-1} & 0 & 0 \\ 0 & r^{-1} & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{5.5}$$

Then during detection, interval of admissible scales (minimum and maximum scales) is applied to these scale-free regions and true regions corresponding to likely feature

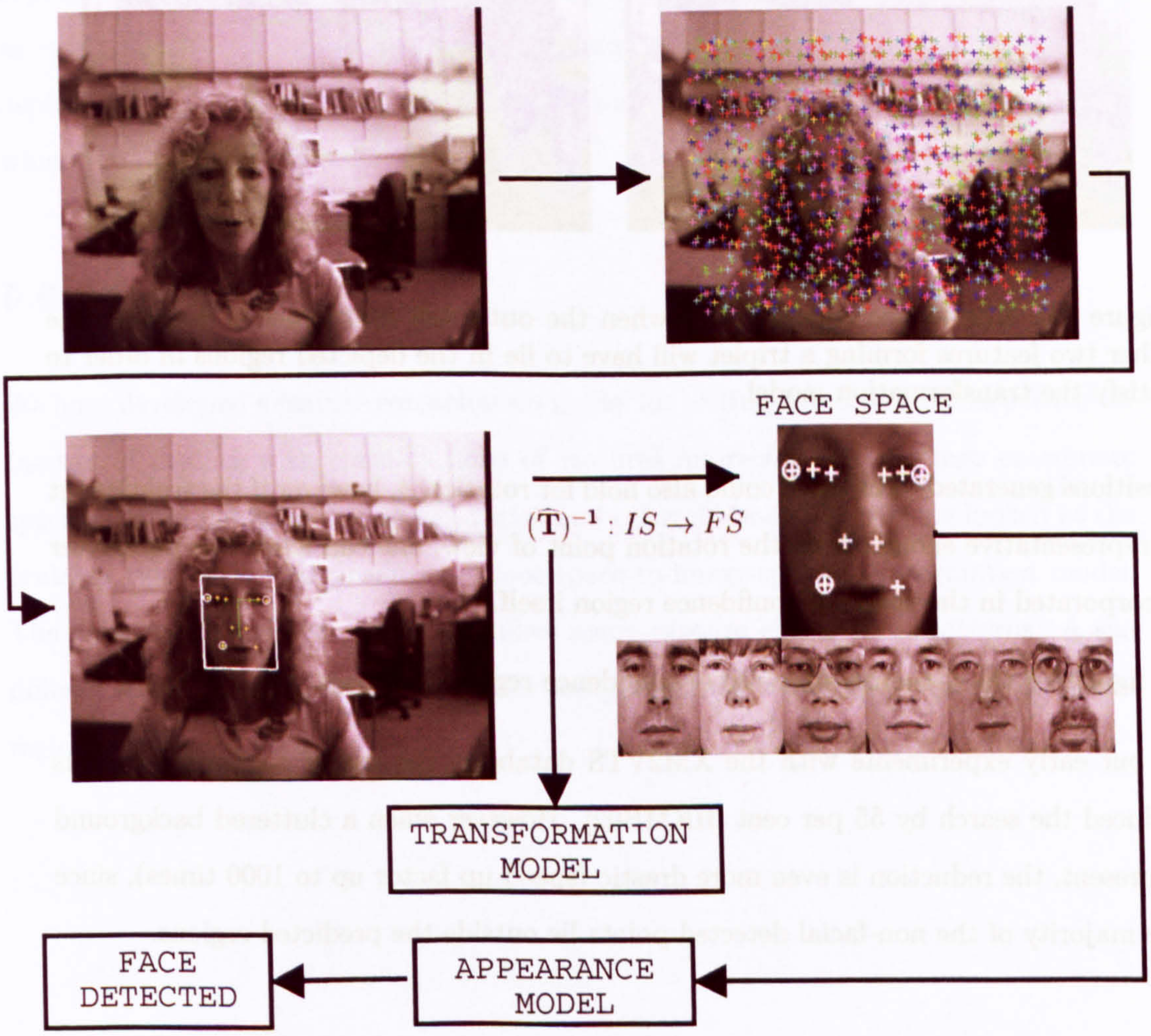

Figure 5.5: Use of the transformation and appearance model. First feature detectors are run on the image. Then the triplets of the detected features are checked for their configuration using the transformation model. The transformation from the image space into the face space ($\widehat{T}^{-1}$) is computed by using correspondences between the detected and the face space feature coordinates (the three detected features are mapped onto the face space coordinates). The detected features are depicted as circles and the face-space coordinates as crosses. Triplets yielding high probability of the transformation $\widehat{T}^{-1}$ are used to register the underlying patch into the face space using $\widehat{T}^{-1}$. The registered patch is then subjected to the appearance test, which decides whether the patch is a face or not.

Figure 5.6: Confidence regions used: when the outer left eye corner is detected, the other two features forming a triplet will have to lie in the depicted regions in order to satisfy the transformation model.

positions generated. The same could also hold for rotation ϕ, however if the training set is representative enough from the rotation point of view, one can leave this parameter incorporated in the shape of confidence region itself.

In figure 5.6 the demonstrative use of confidence regions is shown.

In our early experiments with the XM2VTS database, the use of confidence regions reduced the search by 55 per cent [HKMB02]. However when a cluttered background is present, the reduction is even more drastic (speed-up factor up to 1000 times), since the majority of the non-facial detected points lie outside the predicted regions.

5.5 Geometric registration

Once the transformation from face-space coordinates to image coordinates is established, the photometric content can be registered into the face space, where the appearance will be verified (see schema 3.1). This process involves only promising hypotheses triplets from the point of view of the aforementioned transformation model.

In fact the whole image (not only the part underlying the detected features) can be warped into the face space, however it is desirable to deal only with the face region. For this purpose we chose to cut out a box around the face. The face-space coordinates of its corners were found experimentally, so that the majority of registered faces would fit

in this box and the background would stay out. Also some form of the masking could be used here, however it turned out that this is was not necessary, since the model coped well even with bits of the background. The registration itself is implemented as resampling of the image using the computed affine transformation T. Figure 5.7 depicts the bounding box in the face space and the resulting registered image patch which was used for further processing.

5.6 Summary

We have developed a feature constellation model for sorting detected local face features. Instead of dealing with constellations of features indirectly in the image coordinate space as other methods do (angles, ratios and other ad hoc methods) we looked at the problem rigorously and produced a face-space-to-image-space transformation model. The resulting model also helps to remove scene capture effects from patterns. A significant speed up of the hypotheses selection process was achieved by using confidence regions.

Figure 5.7: Face hypotheses 1–12 based on detected features (circled) together with the corresponding patches taken for further processing (images taken from the BioID database)

Figure 5.7 (continued) : hypotheses 13–24

Figure 5.7 (continued) : hypotheses 25–36

Chapter 6

Appearance model

In this chapter our approach towards face appearance modelling will be presented. We use the appearance model as the final step in face hypothesis verification as depicted in the schema in Figure 3.1.

6.1 Face appearance modelling

The appearance of a human face varies over race, time, sex and due to expression, lighting and head pose changes. In order to construct a reliable system, a model is required which can represent all such variations and which can distinguish the face from the background.

In the previous parts of our algorithm, shape and scene capture effects were to a great extent removed by mapping the data into the face space and a finite set of face hypotheses based on good transformation score was left for further processing. The appearance test is the final verification step where it is decided if the face is present or not based on the gray-scale values. Its input is a geometrically registered image patch containing the underlying photometric information (see Section 5.5) and its output is a score which expresses the consistency of the image patch with the face class. From the technical point of view, any existing state-of-the-art face/nonface classifier could be used here. For example, Adaboost [VJ01], Support Vector Machine, kernel PCA, kernel

Fisher's discriminant, neural networks etc. This fact demonstrates a great versatility of our method, since we do not have to rely on one particular setup or model. In order to choose a suitable model we tested two mainstream models and finally chose Support Vector Machine after taking into account the computational requirements and the performance exhibited on realistic datasets. The following sections will discuss these findings in greater detail.

6.2 PCA-based appearance model

Principal Component Analysis (PCA) is one of the first methods used in face recognition and detection [TP91, MP95, MP96]. The probabilistic face model using PCA functions by projecting face images onto a feature space that spans the significant variations among typical face patterns. The basis vectors of this space are known as "eigenfaces", because they are the eigenvectors (or principal components) of the covariance matrix of the distribution. The projection operation characterizes an individual face by a weighted sum of the eigenface features. Therefore, in order to recognize a particular face, it is necessary only to compare these weights to those of known individuals.

In order to recognize the whole face class, probabilistic models using metrics connected to the aforementioned projection subspace were constructed. Two types of models in particular were used for modelling object class appearance:

1. **A multivariate unimodal Gaussian** (for unimodal distributions of detected objects)
2. **A multivariate mixture of Gaussians** (for multimodal distributions of detected objects)

These probability densities expressed as likelihoods using projection parameters can be used as a face appearance model.

Correlation was the first method that was used in computer vision for object detection. This is an optimal method for the detection of a deterministic signal corrupted by

additive white noise. Backgrounds which could possibly be expressed as white noise are unrepresentative and a face class cannot be expressed as a specific deterministic signal, therefore this simplistic model does not work in practice. Moghaddam's and Pentland's probabilistic approach [MP95, MP96] tries to model objects from real world by probability distributions - i.e. every possible pattern is assigned its probability that it belongs to the object class. The image of a face is considered to be a multi-dimensional vector, the components of which are grey level values. The dimension of the vector is usually very high (for instance for an image of 100×100 pixels, the dimension is 10000). In order to model the probability density reliably in such a high-dimensional vector space, the number of the training patterns would have to be much higher than the dimensionality. Reducing this "pixel oriented" high dimensionality by working in some subspace (linear or nonlinear) is a natural way to solve this problem. PCA methods use linear subspace of the high dimensional feature space. Projection space is constructed through the Karhunen-Loéve transform [DH73, TK99, Bis97] often referred to as PCA.

Specifically, given a set of training images $\{\mathbf{x}^t\}_{t=1}^{N_T}$ from an object class (in our case human faces), the requirement is to estimate the class membership or likelihood function for new unseen data $\mathbf{x}$, i.e. $P(\mathbf{x}|\Omega)$, where Ω denotes the object class.

Computing PCA

Given a training set of m-by-n images, the training set of vectors $\{\mathbf{x}^t\}$, where $\mathbf{x} \in \mathcal{R}^{N=mn}$ is created by concatenating the rows or columns (it depends on definition) of the image (or image window). The basis functions for Karhunen-Loève transform (KLT) are obtained by solving the eigenvalue problem:

$$\Lambda = \Phi^T \Sigma \Phi \tag{6.1}$$

where Σ is the sample covariance matrix, Φ denotes a matrix of eigenvectors of Σ arranged in columns, and Λ is the corresponding diagonal matrix of eigenvalues.

It should be mentioned here that although the order of the eigenvectors in matrix Φ is not important, the correspondence between an eigenvector and eigenvalue in diagonal

matrix Λ must be preserved, i.e. n-th eigenvector (column) corresponds to the n-th eigenvalue in matrix Λ. Since the covariance matrix is real, symmetrical and positive definite, it implies, that eigenvalues are positive. The eigenvectors are constrained to form an orthogonal basis of the vector space. From the statistical point of view this means that the coordinates (new features) in this eigenvector basis are not correlated. In PCA, the eigenvectors corresponding only to the largest eigenvalues are extracted from the matrix Φ. They form a subspace of the whole feature space. The projection into this space forms the feature vector $\mathbf{y} = \Phi_M^T \tilde{\mathbf{x}}$, where $\tilde{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$ and $\bar{\mathbf{x}}$ is the ensemble mean vector and Φ_M is a sub-matrix of Φ containing the principal eigenvectors.

PCA can be understood as a linear transformation $y = \mathcal{T}(x) : \mathcal{R}^N \rightarrow \mathcal{R}^M$. The resulting principal components of a vector (i.e. coordinates in the new basis) preserve the major linear correlations in the data and discards the minor ones. By selecting the first M principal components (i.e. first M eigenvectors corresponding to the first M largest eigenvalues) an orthogonal decomposition of the vector space $\mathcal{R}^N$ into two mutually exclusive and complementary subspaces is accomplished: the principal subspace (or feature subspace) generated by the following set of basis vectors $\mathbf{F} = \{\Phi_{i=1}^{M}\}$ containing the principal components and its orthogonal complement represented by $\bar{\mathbf{F}} = \{\Phi_{i=M+1}^{N}\}$.

The $\mathcal{L}_2$ norm of the vector $\mathbf{x}$ can be decomposed into these two subspaces. The component in the orthogonal subspace $\bar{\mathbf{F}}$ is denoted as the distance-from-feature-space (DFFS) which is the simple Euclidean distance. The component of $\mathbf{x}$ lying in the feature space $\mathbf{F}$ is named distance-in-feature-space (DIFS) and is generally not an Euclidean distance, rather Mahalanobis distance. This decomposition is depicted in Figure 6.1.

For illustration the set of 20 eigenfaces is shown in Figure 6.2.

Now this decomposition is used for an estimation of the high-dimensional Gaussian densities. Let us demonstrate it for the case of unimodal Gaussian density. After sample mean $\bar{\mathbf{x}}$ and covariance Σ have been estimated, the resulting density $P(\mathbf{x}|\Omega)$ is defined as follows:

$$P(\mathbf{x}|\Omega) = \frac{exp[-\frac{1}{2}(\mathbf{x}-\bar{\mathbf{x}})^T\Sigma^{-1}(\mathbf{x}-\bar{\mathbf{x}})]}{(2\pi)^{N/2}|\Sigma|^{1/2}} \tag{6.2}$$

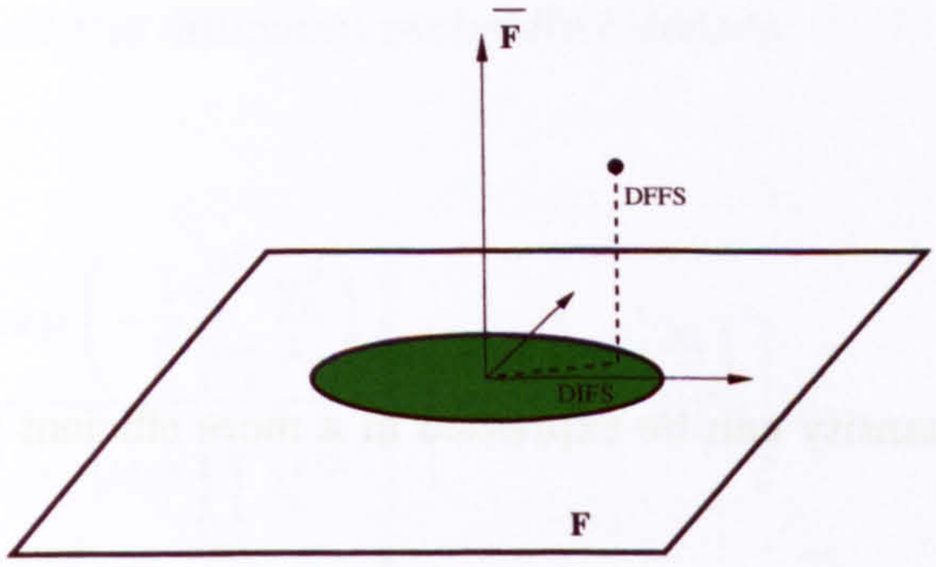

Figure 6.1: Decomposition into principal subspace $\mathbf{F}$ and its orthogonal complement $\bar{\mathbf{F}}$ for a Gaussian density

Figure 6.2: Set of eigenfaces computed with images taken from the BANCA database

The sufficient statistics for expressing this probability density is the Mahalanobis distance.

$$d(\mathbf{x}) = \tilde{\mathbf{x}}^T \Sigma^{-1} \tilde{\mathbf{x}} \tag{6.3}$$

where $\tilde{\mathbf{x}} = \mathbf{x} - \bar{\mathbf{x}}$ This quantity can be expressed in a more efficient way as follows:

$$\begin{aligned} d(\mathbf{x}) &= \tilde{\mathbf{x}}^T \Sigma^{-1} \tilde{\mathbf{x}} \\ &= \tilde{\mathbf{x}}^T [\Phi \Lambda^{-1} \Phi^T] \tilde{\mathbf{x}} \\ &= \mathbf{y}^T \Lambda^{-1} \mathbf{y} \end{aligned} \tag{6.4}$$

where $\mathbf{y} = \Phi^T \tilde{\mathbf{x}}$ are the new variables obtained by changing the coordinates into the new orthogonal basis. Moreover it holds:

$$d(\mathbf{x}) = \sum_{i=1}^{N} \frac{y_i^2}{\lambda_i} \tag{6.5}$$

When using only an M-dimensional subspace of principal components (i.e. projection into an M-dimensional feature space), estimation only of a part of this quantity can be used (since the $N - M$ coordinates are unknown). Pentland and Moghaddam used an estimator, which consists of combination of DFFS and DIFS and is defined as follows:

$$\tilde{d}(\mathbf{x}) = \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} + \frac{\epsilon^2(\mathbf{x})}{\rho^*} \tag{6.6}$$

where $\epsilon^2(\mathbf{x}) = \sum_{i=M+1}^{N} y_i^2 = \|\tilde{\mathbf{x}}\|^2 - \sum_{i=1}^{M} y_i^2$ and $\rho^* = \frac{1}{N-M} \sum_{i=M+1}^{N} \lambda_i$

The first term in the estimator defined in the equation 6.6 is the DIFS and the second term is the DFFS weighted by ρ^*, where ρ^* is computed in such a way that it maximises

the relative entropy (or Kullback-Leibler distance) between the original probability distribution $P(\mathbf{x}|\Omega)$ and the estimated probability density:

$$\begin{aligned}\hat{P}(\mathbf{x}|\Omega) &= \left[\frac{exp\left(-\frac{1}{2}\sum_{i=1}^{M}\frac{y_i^2}{\lambda_i}\right)}{(2\pi)^{M/2}\prod_{i=1}^{M}\lambda_i^{1/2}}\right]\left[\frac{exp\left(-\frac{\epsilon^2(\mathbf{x})}{2\rho}\right)}{(2\pi\rho)^{(N-M)/2}}\right] \\ &= P_F(\mathbf{x}|\Omega)\hat{P}_{\bar{F}}(\mathbf{x}|\Omega) \qquad (6.7)\end{aligned}$$

Using likelihood for classification

The density estimation $\hat{P}(\mathbf{x}|\Omega)$ is hereafter used to compute a face appearance score S for a given input image window based on vector $\mathbf{x}$. Vector $\mathbf{x}$ is obtained by concatenating the columns or rows of the extracted window as mentioned above.

$$S(WINDOW|\Omega) = \hat{P}(\mathbf{x}|\Omega) \qquad (6.8)$$

where $\mathbf{x}$ is the column-wise concatenated input image window.

Once all the scores for the hypotheses were obtained the hypotheses can be ordered in ascending order according to $S(WINDOW|\Omega)$ and top best can be taken as output or possibly a thresholding can be used to remove all non-face hypotheses. The results of two experiments carried out on the XM2VTS database using this approach will be presented in section 7.2.

6.3 Support Vector Machine based appearance model

The foundations of Support Vector Machines (SVM) have been created by Vapnik [Vap95, CV95]. SVMs became popular due to many attractive features and promising performance in real-life tasks. The crucial concept behind them is the Structural Risk Minimization principle, in contrast to the Empirical Risk Minimization approach often

used within statistical learning methods. The Structural Risk Minimization principle aims at minimizing an upper bound on the generalisation error, as opposed to the Empirical Risk Minimization which minimizes the error only on the training data. Use of this principle gives SVMs greater potential to generalise, which is the main goal in statistical learning.

Support Vector machines belong to a larger group of machine learning algorithms so called kernel methods. The use of kernels as generalized dot products allows to construct nonlinear decision surfaces in an efficient way and thus solve nonlinear problems. Currently, a lot of research effort is devoted to kernel versions of PCA, Fisher discriminant and others and huge number of various results have been reported recently.

SVMs were used successfully in face detection in the context of the sliding window approach [OFG97, HPP, LGL00]. In our localization algorithm we use SVM as the means of distinguishing well localized faces from background or misaligned face hypotheses using geometrically registered photometric data (by using face space) coming from the previous parts of our algorithm.

As mentioned above, kernel methods are currently a very popular research topic. Although we will briefly introduce SVMs below, we would like to refer an interested reader to the extensive number of existing sources on this topic [httb].

6.3.1 Introduction to SVMs

In any pattern classification problem we want to estimate a function $f : \Re^N \mapsto \pm 1$ using input-output training data $(\mathbf{x}_1, y1)$, $(\mathbf{x}_2, y2)$, $(\mathbf{x}_3, y3)$, ... $\in (\Re^N \times \pm 1)$ such that f will correctly classify unseen samples $(\mathbf{x}_i, y_i)$, i.e. $f(\mathbf{x}_i) = y_i$. Please note we constrain the discussion to a two-class classification problem where the labels are $+1, -1$. For multi-class problems slight modifications would be needed.

Consider first the case where we have a linearly separable training set $Z = (\mathbf{x}_i, y_i)_{i=1}^n$, that is, there exists a linear discriminant function of the form:

$$\mathbf{x} \mapsto \mathbf{w}^T\mathbf{x} + b, \mathbf{w} \in \Re^d, b \in \Re$$

for which the corresponding decision function $t = sgn(\mathbf{w}^T\mathbf{x} + b)$ has the property $\hat{e}_n(t) = 0$, where $\hat{e}_n(t)$ is empirical risk or training error defined as follows:

$$\hat{e}_n(t) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}_{\{t(\mathbf{x}_i)\neq y_i\}}$$

The condition above just states that all the training samples were classified correctly.

Of course, there can be infinitely many linear discriminant functions that separate the training set without errors and, consequently our task is to choose the best one. Here, one of the crucial principles in SVM comes to use. It is the margin. Let d_+, d_- be the shortest distance from the separating hyperplane to the closest positive (negative) sample. The "margin" of a separating hyperplane is defined to be $d_+ + d_-$. For the linearly separable case SVM looks for the separating hyperplane with the largest margin. This can be done by solving a Lagrangian problem. The hyperplane that maximizes the geometric margin is called the optimal separating hyperplane.

Non-separable cases require additional modifications, in particular slack variables in the classification cost function. We will not go into details and refer the reader to the literature [Vap95, Erä01].

So far we talked about SVM that solves the classification problem by using only a linear discriminant function. It was the introduction of kernels that allowed SVMs to become non-linear. The idea was first reported in the sixties by Aizerman [ABR64]. If we suppose we first map the data to some other higher-dimensional Euclidean space $\mathcal{H}$, using a mapping which we can call Φ:

$$\Phi : \Re^d \mapsto \mathcal{H}$$

then using the virtues of the SVM training algorithm we can train the SVM in this new space by just using dot products in $\mathcal{H}$, i.e. functions of the form $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$. The kernel function K performs the dot product implicitly, i.e. $K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$. If we use some kernel function K in the training , we do not need to know explicitly what Φ is. It produces SVM which "lives" in an high dimensional space but still takes almost the same amount of time to train as in the linear case. Linear separation by

an optimal hyperplane is still performed but in a different space. The big issue here is of course the choice of kernels which lead to different nonlinear decision surfaces. The choice of kernel for a given problem is still an open research issue. The most commonly used kernels are:

- Linear kernel: $K(\mathbf{x}_1, \mathbf{x}_2) = < \mathbf{x}_1, \mathbf{x}_2 >$
- Polynomial kernel: $K(\mathbf{x}_1, \mathbf{x}_2) = (< \mathbf{x}_1, \mathbf{x}_2 > +1)^p$
- Radial basis function kernel: $K(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{\gamma}\right)$
- Sigmoid-function kernel: $K(\mathbf{x}_1, \mathbf{x}_2) = \tanh\left(v < \mathbf{x}_1, \mathbf{x}_2 > +a\right)$

Training SVM presents a quadratic programming problem and efficient implementations were tackled by researchers [Pla98]. However, it should be noted that huge training set sizes still remain an implementation problem.

6.3.2 Learning face appearance with SVM

In order to use SVM to learn face appearance, face and non-face examples have to be collected. They have to be representative in order to achieve good generalization on previously unseen data. In our current setup we used the world model subset of the BANCA database (see section 7.3) as the training data. Due to their intended use in realistic authentication tests these data carry enough variability in face appearance including imaging effects. However there are two important questions which need to be addressed. The first is, which feature vectors actually to learn, second, how complex an SVM-classifier to use. Both of them are strongly related to speed. Let us discuss these matters in more detail.

Regarding the first issue, lots of approaches exist in the literature. Usually dimensionality reduction is performed on raw pixel data first and then a classifier is trained using these low dimensional projections. This approach applies to the use of neural networks [SP98, RBK98] and classical approaches like the one of Turk's or Moghaddam's [TP91, MP95]. However, our experiments with SVMs showed that prior dimension reduction is not necessary and one can work directly with the image intensities.

SVMs handle the problem of dimension well and with regard to performance they are comparable to other approaches used in face detection [HPP].

Regarding the complexity of SVMs, due to the speed requirement, we decided to use a two-stage classification. A fast linear-kernel SVM is used in the first stage on low resolution face-space patches. This stage produces a list of hypotheses with accurately localized faces being mostly at the top of the list. However the linear SVM in low resolution cannot distinguish between slightly misaligned faces and also its robustness to the background is not perfect. What is important is that it allows us to preselect N best localization hypotheses in a fast manner.

For verification purposes N output localization hypotheses (where N is a small number) does not present any problem, since most of the verification tests are not time costly. In other words it means that N verification tests would have to be performed and some selection rule used - e.g. max/min on the verification score. A description of a successful localization setup using maximum of the normalized correlation in the Fisher space as the selection rule can be found in [HKKK03, SKKM03].

Although multiple hypotheses on the output do not degrade the quality of the subsequent verification, we also tried to tackle the problem of just one localization hypothesis on the output. This also allows us to perform a direct comparison with other methods since they usually produce only one detection/localization per image.

There is another very important parameter which strongly influences the performance and that is the resolution of the photometric data. Our chosen resolution in the first stage was 20×20 pixels and it copies the choice of several other authors. If the resolution is too high, facial details start playing a significant role and a huge amount of training data is needed to capture this variability. One should not forget that face model tries to distinguish true facial patches from false ones (background, parts of faces, etc.) and this presents a very hard problem. Every path that decreases the complexity of the learned data should be followed. The choice of representative negative examples is also a hard issue. A bootstrapping technique proposed by Sung and Poggio [SP98] offers a nice way to solve this task. It works as follows:

1. Start with a small and possibly incomplete set of non-face examples in the training

database

2. Train the classifier with the current database of examples

3. Run the detector on a sequence of random images. Collect all the non-face patterns that the current system wrongly classifies as faces. Add these non-face patterns to the training database as new negative examples

4. Return to step 2

In the following sections we will go into more detail about the chosen model.

6.3.3 First Appearance Test Stage – linear coarse resolution SVM

Training data

Data which are used as an input by the appearance model are generated by the hypotheses generator. The chosen approach gives us the means to geometrically normalize the photometric data and therefore the scale and rotation error tolerance can be very low. We hoped that this should make the model more discriminative than the sliding window methods as explained earlier. Examples of the normalized patches coming from the hypotheses generator to this stage are shown in Figure 5.7.

These patches should be classified into face and non-face classes based on score. Since many triplets (up to 58 as seen in Figure 5.2) can lead to a successful localization, we need a score expressing the accuracy of the given patch being a face. SVMs offer a very suitable score and it is the discriminant function value. Without going into detail, it basically represents a distance of the sample from the hyperplane in the high dimensional kernel space.

As positive examples, 20 × 20 pixel face patches registered in the face space using manual groundtruth features were used – examples of which can be seen in Figure 6.3. The set of negative examples was obtained by applying the bootstrapping technique introduced above. It involved running the detector with a small initial set of negatives and collecting those background samples that were misclassified into the face class.

Figure 6.3: 20×20 image patches used by the first stage of appearance test (last row depicts examples of negative samples)

This guaranteed that only those samples that have a high information value will be considered. We experimentally tested several sets of background patches and chose the one yielding the best performance.

6.3.4 Second Appearance Test Stage – non-linear fine resolution SVM

As mentioned before, the first stage gives an ordered list of hypotheses, based on low resolution sampling. In our experiments it was observed that this step alone is unable to distinguish between slightly misaligned faces. This is caused by the fact

that downsampling to low resolution makes even slightly misaligned faces look almost identical. In order to increase the accuracy we employ a fine resolution classifier. Although we verified experimentally that linear SVM manages to distinguish faces from background, in the case of higher resolution the dimensionality of the problem increases significantly and therefore linear SVM did not cope well. Faces in higher resolution require more training data and therefore a more complex classifier has to be used due to the increased amount of emerging visual details. More complex classification requires also more computation time, however we should note that here we do not aim to classify all face/non-face hypotheses, but only a small fixed number of the fittest ones passing the first appearance test. We tested nonlinear-kernel SVMs, which performed well for the purpose.

Training set

Since we aim here to distinguish slightly misaligned faces from good localizations, the training set has to reflect that. With the use of the groundtruth, we used the localization error measure d_{eye}, which will be defined in Section 7.1. Using this measure in the training, the best N hypotheses produced by the linear SVM can be sorted into positives and negatives. In our experiments we used value $d_{eye} = 0.05$ as the threshold. Also, since our main focus is to accurately localize the eye centres, in this stage we redefined the face patch borders so that the patch contains mainly the eye region. Experimentally we found a suitable bounding box around eyes and the geometric normalization was also based only on eye centres (i.e. two point normalization). An example of the data used for training the classifier is shown in Figure 6.4.

It should be noted that since this stage aims to identify the precise location, quite a lot of faces which could be regarded as successful localizations from the human point of view are labelled as negative examples in order to make the system more sensitive to misalignments.

The results on several databases comparing PCA and SVM-models will be discussed in detail in chapter 7.

Figure 6.4: Training samples for the refinement stage, rows 1–2: negatives, rows 3–4: positives

6.3.5 Illumination correction

We tested several illumination correction techniques and we decided to use the zero mean and unit variance normalization. It is a very simple correction which does not actually remove any shadows from the face patches, it is more a scaling technique than the illumination correction. It removes a linear distortion from the signal, i.e. normalizes bias and contrast. All the variations caused by shadows have to be therefore incorporated in the training set in order to achieve a good generalization. The field of illumination correction is currently very popular, so in future, possibly a more sophisticated approach can be exploited by our method.

6.4 Summary

In this section we proposed an appearance model based on two stage Support-Vector-Machine classification. As our results will later show, in order to achieve high localization accuracy, a fine resolution model together with non-linear SVM has to be employed. After dealing with the practical issue of appearance modelling, we can draw a conclusion that since the majority of sliding window localization/detection methods

use small resolution for classification (similar to our linear model), the localization accuracy of such methods cannot therefore be satisfactory. Next chapter will present the results of the majority of our experiments in a concise manner.

Chapter 7

Experiments and Evaluation

7.1 Evaluation data for authentication scenarios

In contrast to face recognition and verification, there exists no common performance evaluation for face detection and localization. Many authors measure the performance of their system in terms of receiver operating curves (ROC), but the term "successful detection/localization" is either not explicitly defined at all, or at best in an ad hoc way [YKA02]. The only exception is the work in [JKF01], where a stringent localization criterion has been proposed and which we decided to adopt here. It takes into account the position of facial features, in particular eye centres. The measure is defined as follows:

$$d_{eye} = \frac{max(d_l, d_r)}{||C_l - Cr||} \tag{7.1}$$

where C_l, C_r are the groundtruth eye centre coordinates and d_l, d_r distances between the detected eye centres and the groundtruth ones.

It was established experimentally that in order to succeed in the subsequent verification step [SKKM03], the localization accuracy has to be below $d_{eye} = 0.05$. It is due to the fact that the majority of face verification algorithms are very sensitive to misregistrations. It is also true that at this value of d_{eye}, the localization error starts to be visually

noticeable as could be seen in Figure 7.1. Therefore in the following evaluations we treat localization with d_{eye} above 0.05 as unsuccessful.

To evaluate and compare the advocated method with others, we focused on the XM2VTS, BANCA and BioID face databases. These databases were specifically designed to capture realistic face authentication conditions and are currently regarded as the most important benchmarking data sets. They contain faces which are sufficiently big for a subsequent verification or recognition experiments. For XM2VTS and BANCA, even rigorous verification protocols exist which gives an opportunity to evaluate the overall performance of the whole face verification system.

Although many more face databases exist (like CMU, FERET, etc.), evaluation on them is beyond our focus, due to the different purpose of the databases (small faces, very bad capture conditions, multiple faces in the scene, no controlled access, poor resolution unsuitable for verification and recognition etc.).

In the following sections, the localization results achieved on the aforementioned datasets as well as a comparison of various modifications of our method discussed in the previous chapters will be presented. Also a performance evaluation of the feature detectors used will be presented in Section 7.6.

7.2 XM2VTS database

The database is primarily intended for research and development of personal identity verification systems where it is reasonable to assume that the client will be cooperative. In order to capture natural variability of clients caused by changes in physical condition, hair style, dress, and mood, subjects were recorded in four separate sessions uniformly distributed over a period of 5 months. The subjects were selected to include adults of both sexes and different ages. As people wearing glasses may be interested in gaining access to services with glasses on or off, both instances would have to be present to develop robust algorithms. A good quality digital camcorder was used to record the database. A protocol has been defined on the database that may be used to evaluate the performance of vision- and speech-based person authentication systems. The protocol is

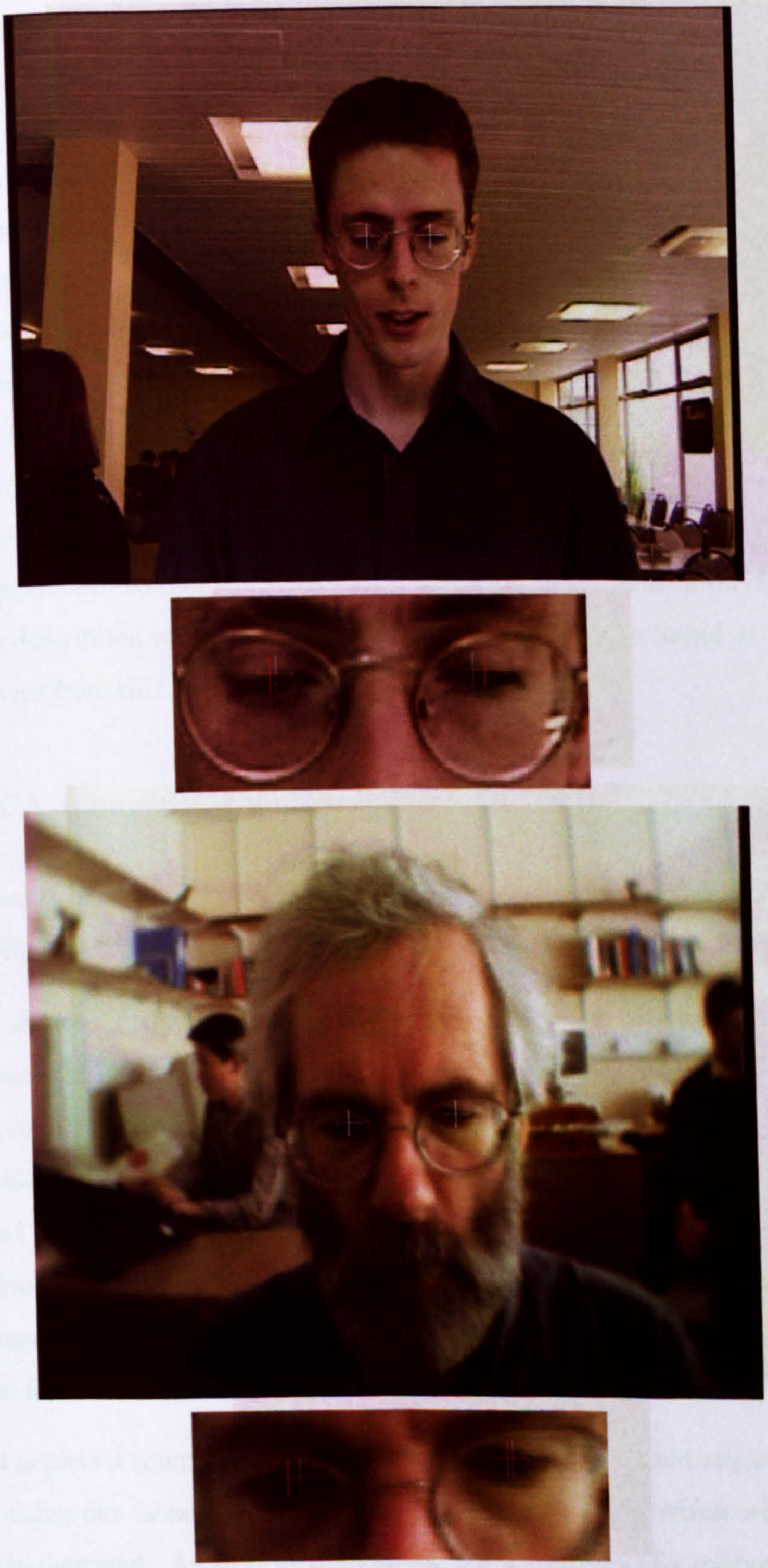

Figure 7.1: Localization with $d_{eye} \doteq 0.05$, please note that the localization error is not big but as shown in the close-up it is clearly visible that at least one of the eye centres is missed - the white cross denotes our localization, the red cross the groundtruth eye centres (Images taken from the BANCA database)

Figure 7.1 (continued)

defined for the task of person verification, where an individual asserts his or her identity. The database is divided into three sets: training set, evaluation set and test set. The training set is used to build client models. The evaluation set is selected to produce client and impostor access scores which are used to find a threshold that determines if a person is accepted or rejected. The test set is selected to simulate real authentication tests. The protocol is based on 295 subjects, 4 recording sessions, and two shots per recording sessions. The database was randomly divided into 200 clients, 25 evaluation impostors, and 70 test impostors. Two different evaluation configurations were defined. They differ in the distribution of client training and client evaluation data. It should be noted that this database is easier, compared to the following two databases. However it can be regarded as a baseline benchmarking set, since it was one of the first large face databases produced and many authors have used it for assessment of their algorithms. A complete description of the database and the protocol can be found at [MMK+99]. Sample images from this database can be seen in Figure 7.2.

7.2.1 PCA versus SVM appearance model

In our initial experiments we used the PCA-based appearance model which was described in chapter 6.

Figure 7.3 shows the performance of the PCA versus SVM-based appearance model using our early Harris-and-PCA-based feature detector [MBHK02, HKMB02]. As seen in the graphs, PCA and SVM performed comparably, however it should be noted that it is most likely due to the fact that the majority of detected features were located on the face and only few background false positive features were involved. It was observed that this feature detector performed well on scenes with a uniform background, however its performance dropped on scenes with a cluttered background. Therefore it proved inadequate for real-life situations.

Figure 7.4 depicts a comparison of the PCA versus the SVM-based appearance model, this time using our latest Gabor-filter-based feature detector, which works well in a cluttered background. As seen in the graphs, SVM dramatically outperforms in this case the PCA-based model. It should also be noted that the computation time in the

Figure 7.2: Sample images taken from the XM2VTS database

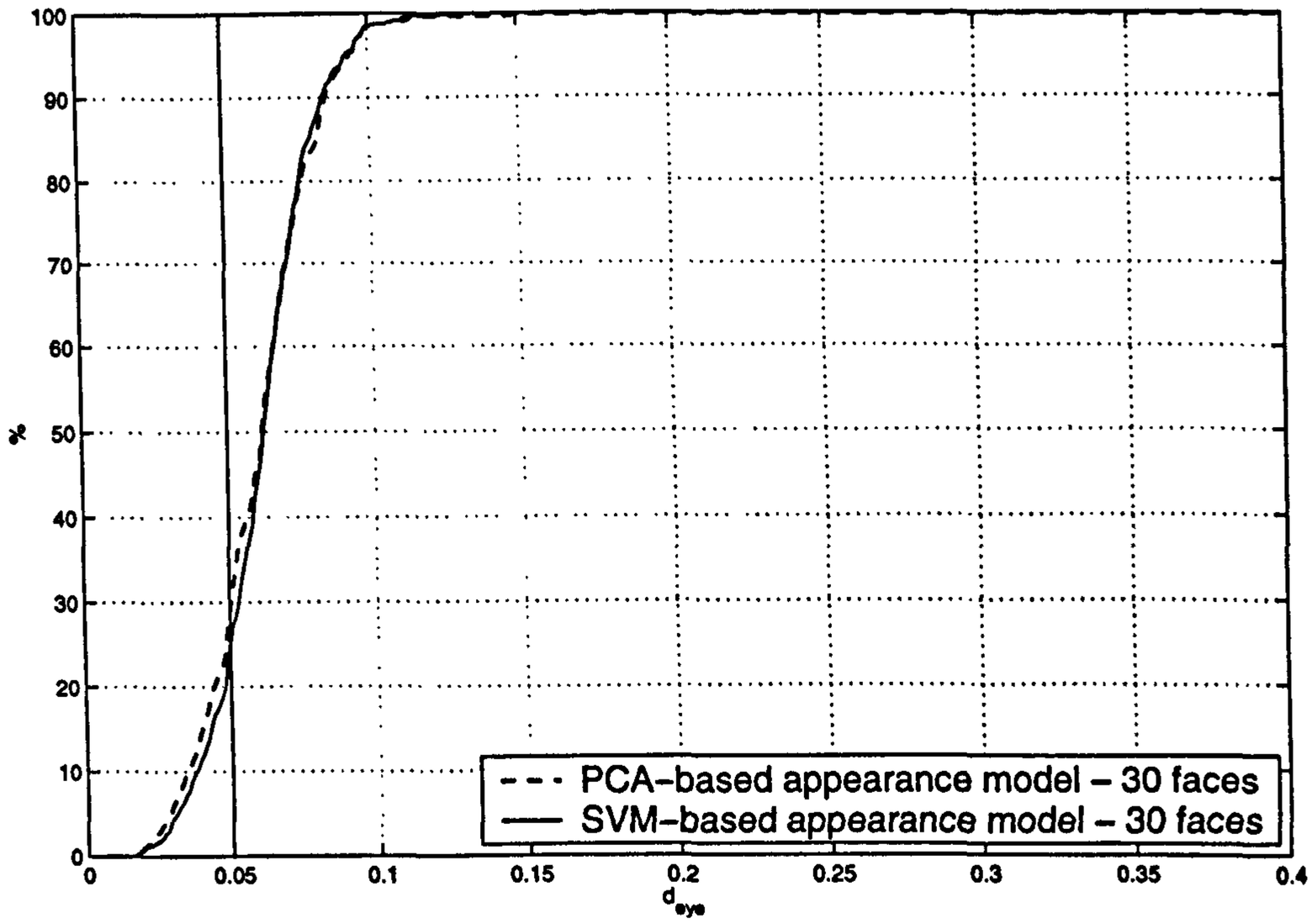

Figure 7.3: Comparison of the PCA and SVM-based appearance model on the XM2VTS database using Harris-and-PCA-based feature detector, 30 faces on the output (cumulative histograms of d_{eye})

case of PCA model is proportional to the number of eigenfaces used (in our experiments 20) and therefore there was a significant slow-down compared to the linear-kernel SVM which effectively involves only a single dot product. We can draw a conclusion from these findings, that unimodal PCA-based Gaussian cannot simply describe the facial cluster discriminatively enough in the presence of a background. In all our subsequent experiments we therefore focused only on the Gabor-based feature detector and SVM-based appearance models.

7.2.2 Localization results on the database

Figure 7.5 shows the latest localization results achieved on the XM2VTS database using Sub-cluster classifier (SCC - section 4.3.3) and GMM-based classifier (section 4.3.4) in the feature detector. A comparison with the localization method of Jesorsky et al. [JKF01] is presented. As mentioned above, we treat localizations with d_{eye} above

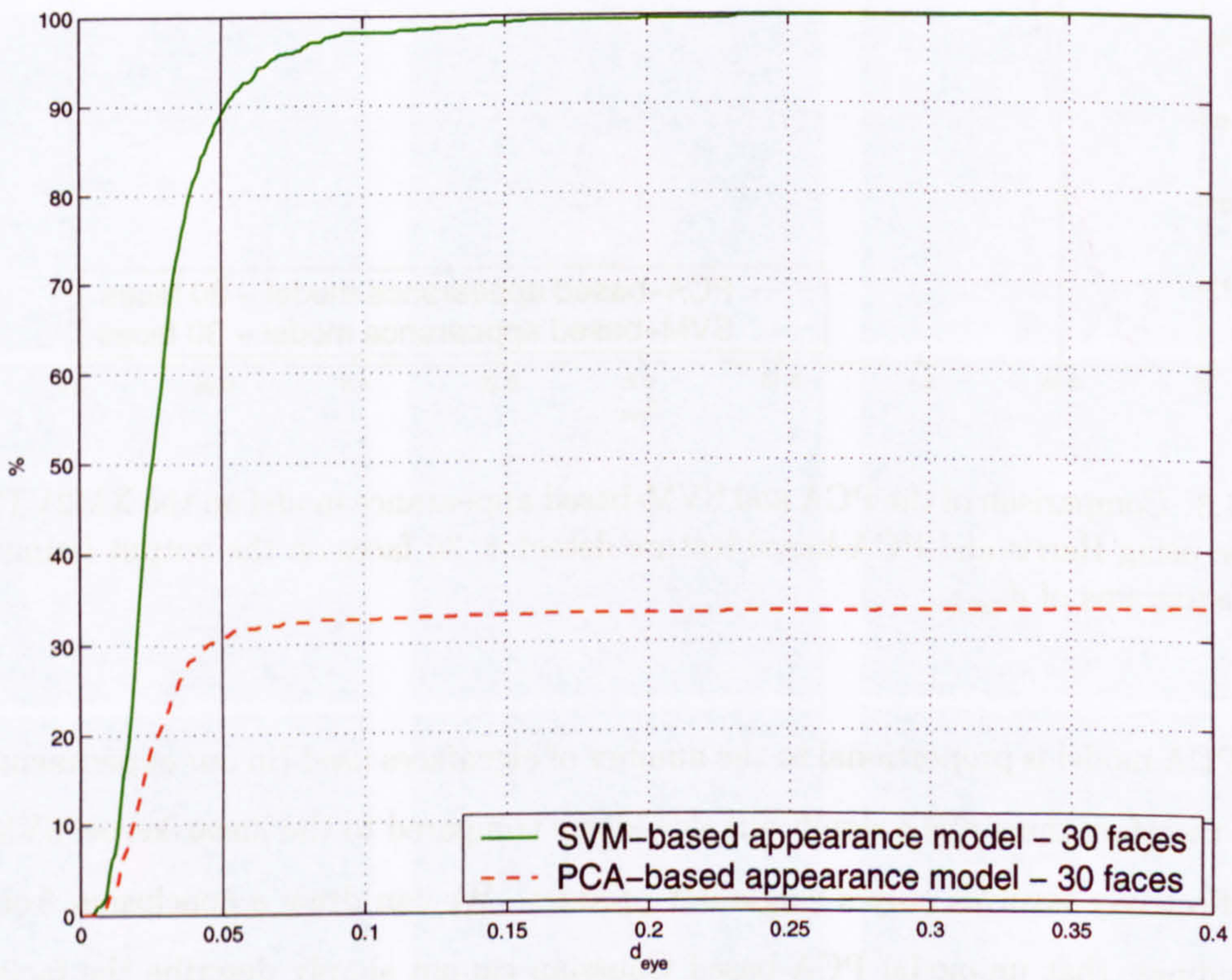

Figure 7.4: Comparison of the PCA and SVM-based appearance model on the XM2VTS database using a Gabor-filters-based feature detector, 30 faces on the output (cumulative histograms of d_{eye})

0.05 as unsuccessful. It should be also noted again that it is an experimentally proven fact that having multiple localization hypotheses on the output does not adversely influence the total performance of the subsequent verification system [HKKK03, SKKM03]. The verification test is cheap computationally and if there is a well localized face among multiple output hypotheses, the system succeeds.

As seen in the graphs, in the XM2VTS case our method gives a similar performance as the baseline method at $d_{eye} = 0.05$ taking the best hypothesis output and outperforms it by 11.6% using top 30 localizations output with GMM (d_{eye} taken as the minimum over all 30 faces). The fine resolution appearance stage improved the results by 32.2% in the GMM case and by 34.6% in the SCC case. The replacement of SCC by GMM increased the overall performance by 1.9% in the case of 30 output faces.

7.2.3 Comparison with a sliding window method

We compared the performance of our system with an SVM-based sliding window approach designed by Kostin et al. [KK02]. Their system contains a sliding window-based eye detector acting as a second stage after a bounding box face localization (typical output for the majority of sliding window methods). This eye detector was also generating several eye position hypotheses. The selection of the best eye-pair is based on a similar technique to ours - the eye pair yielding the best appearance score is produced. The results are presented in Figure 7.6.

7.3 BANCA database

The BANCA database was produced within the BANCA project [ban, htta] for which this algorithm was also developed. The BANCA database is a new large, realistic and challenging multi-modal database intended for training and testing multi-modal verification systems. The BANCA database was captured in four European languages (English, French, Italian and Spanish) and in two modalities (face and voice). For recording, both high and low quality microphones and cameras were used (2 cameras and 2 microphones). The subjects were recorded in three different scenarios, controlled,

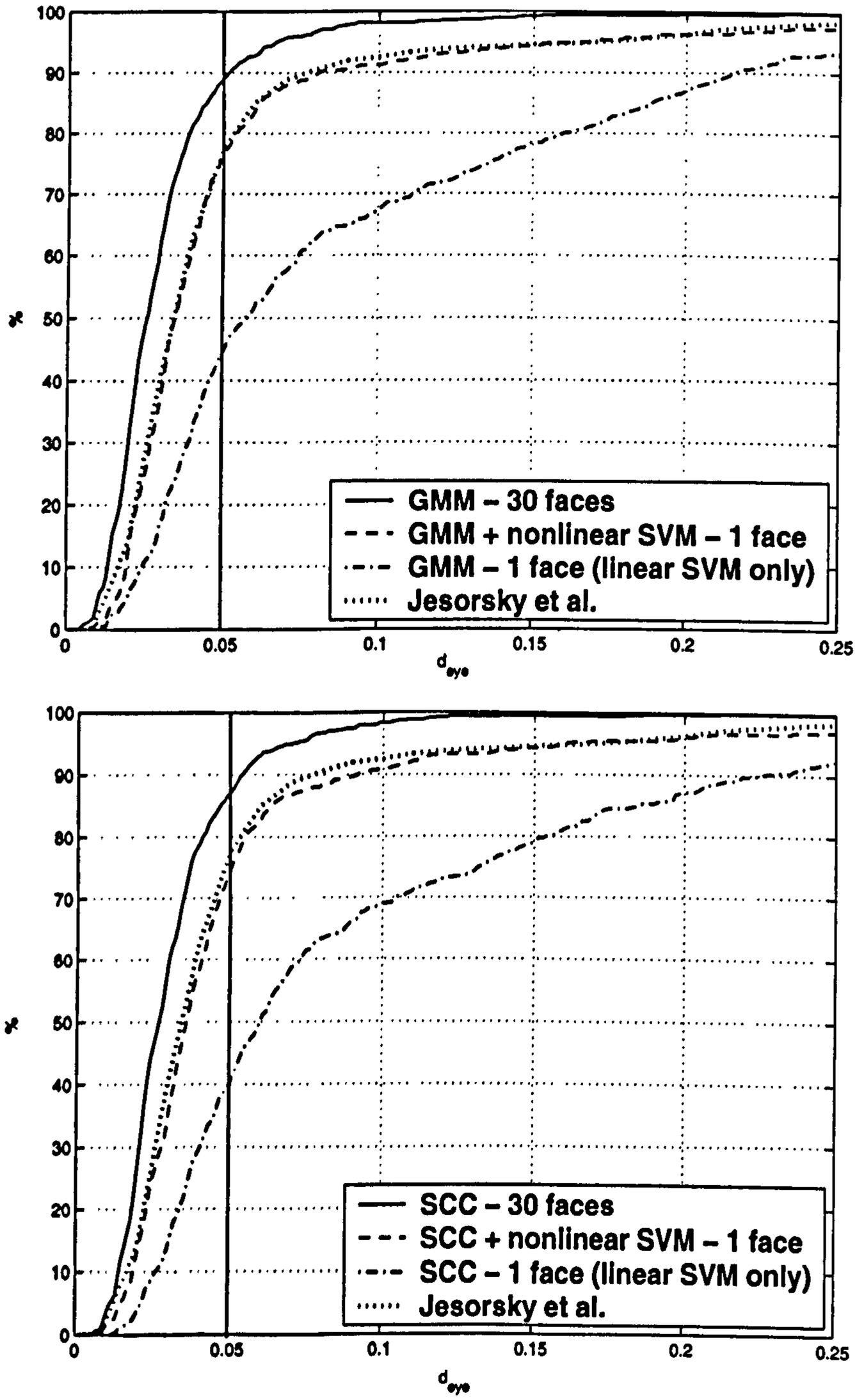

Figure 7.5: Results on the XM2VTS database (cumulative histograms of d_{eye}): GMM top, SCC bottom, the graph of Jesorky et al. taken from [JKF01]

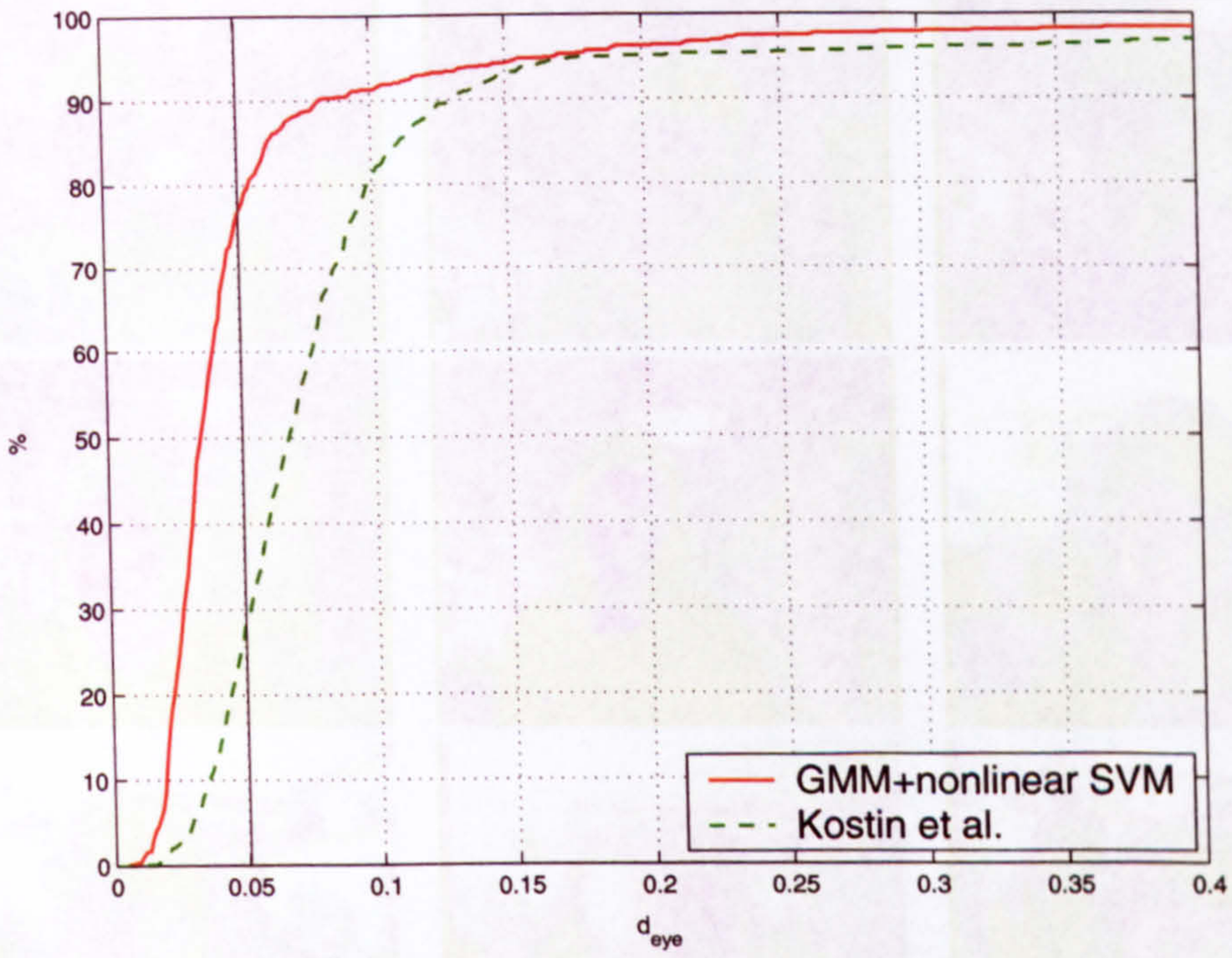

Figure 7.6: Comparison with a sliding window method on the XM2VTS database (cumulative histograms of d_{eye})

degraded and adverse over 12 different sessions spanning three months. In total 208 people were captured, half men and half women. Each subject recorded 12 sessions, each of these sessions containing 2 recordings: 1 true client access and 1 informed (the actual subject knew the text that the claimed identity subject was supposed to utter) impostor attack. The web cam was used in the degraded scenario, while the expensive camera was used in the controlled and adverse scenarios. The two microphones were used simultaneously in each of the three scenarios with each output being recorded onto a separate track of the DV tape. During each recording, the subject was prompted to say a random 12 digit number, his/her name, their address and date of birth. Each recording took an average of twenty seconds. For different sessions the impostor attack information changed to another person in their group.

Associated with the database is the BANCA protocol [BBBB+03]. The protocol defines which sets of data to use for training, evaluation and testing verification algorithms.

The BANCA database offers the research community the opportunity to test their

Figure 7.7: Sample images taken from the BANCA database - English part

multi-modal verification algorithms on a large, realistic and challenging database. It is hoped that this database and protocol will become a standard, like the XM2VTS database, which enables institutions to easily compare the performance of their own algorithms to others. Each language contains 6240 colour images in total making the results statistically significant. Sample images from this database can be seen in Figure 7.7.

7.3.1 Localization results on the database

Figures 7.8, 7.9, 7.10 and 7.11 depict the latest localization results on the English, French, Spanish and Italian parts of the database. The influence of multiple output hypotheses on the accuracy is demonstrated on the English part only (Figure 7.8).

An earlier experiment involved testing the whole verification system [SKKM03], where the selection of the best localization out of multiple output hypotheses coming from the

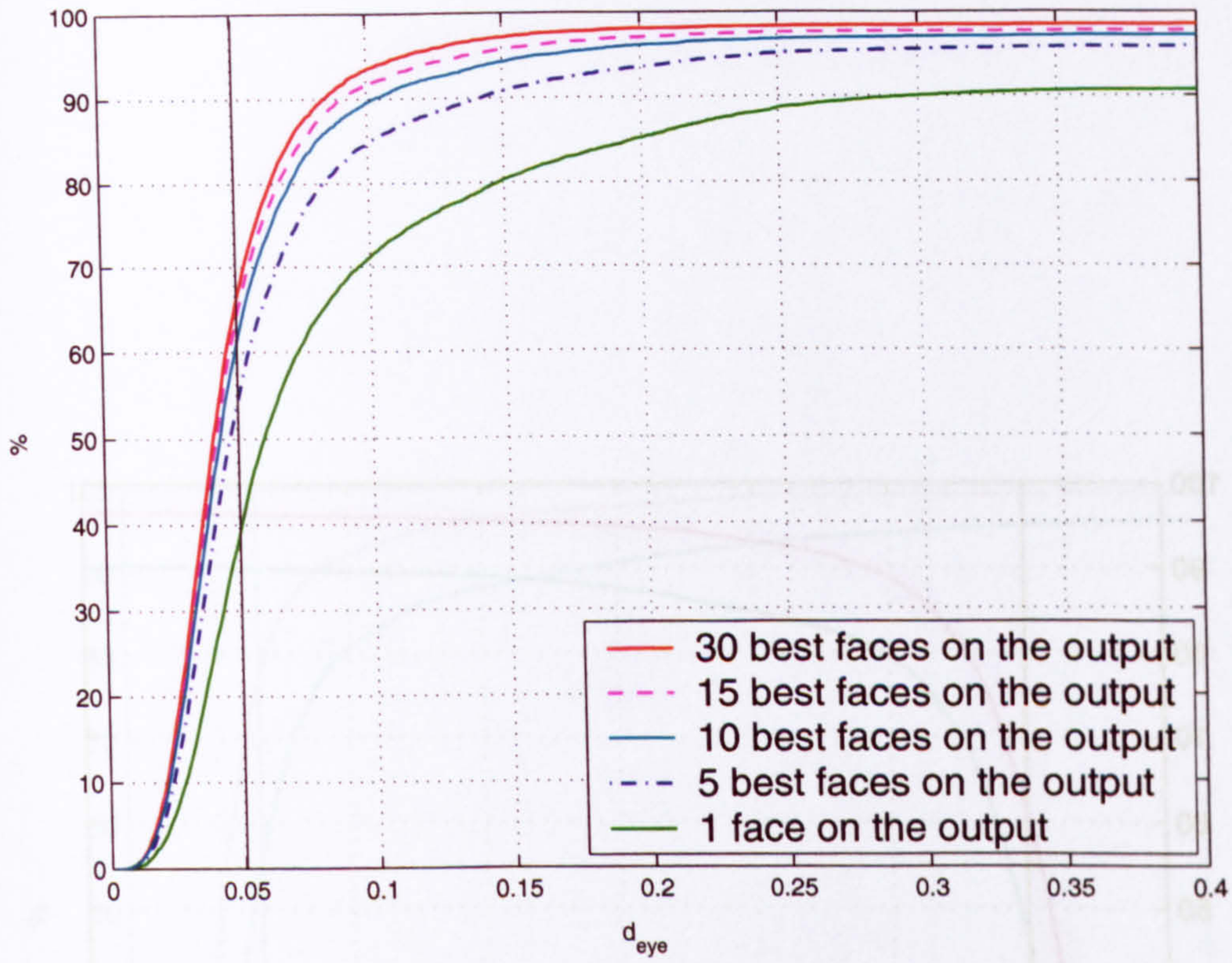

Figure 7.8: Localization results on the English part of the BANCA database (cumulative histograms of d_{eye}), GMM used in the feature detector, please note the influence of the number of output localizations on the performance

linear-SVM appearance model was performed by using client specific templates [HKKK03]. The outcome of the experiment performed on the English part is depicted in Figure 7.12. As seen in the graph, localization fails quite a lot in the case of impostor access, since a template of a different person is used for matching, however it should be noted that in fact this contributes to impostors being correctly rejected. Figure 7.8 presents our latest results on the English part using GMM in the feature detector and the two stage generic appearance model. Figure 7.13 depicts the comparison of the results with the sliding window method of Kostin et al.

7.3.2 Face verification results on the database

As mentioned earlier, the BANCA database is intended to be used in person authentication experiments. The advocated detection/localization algorithm was used as the

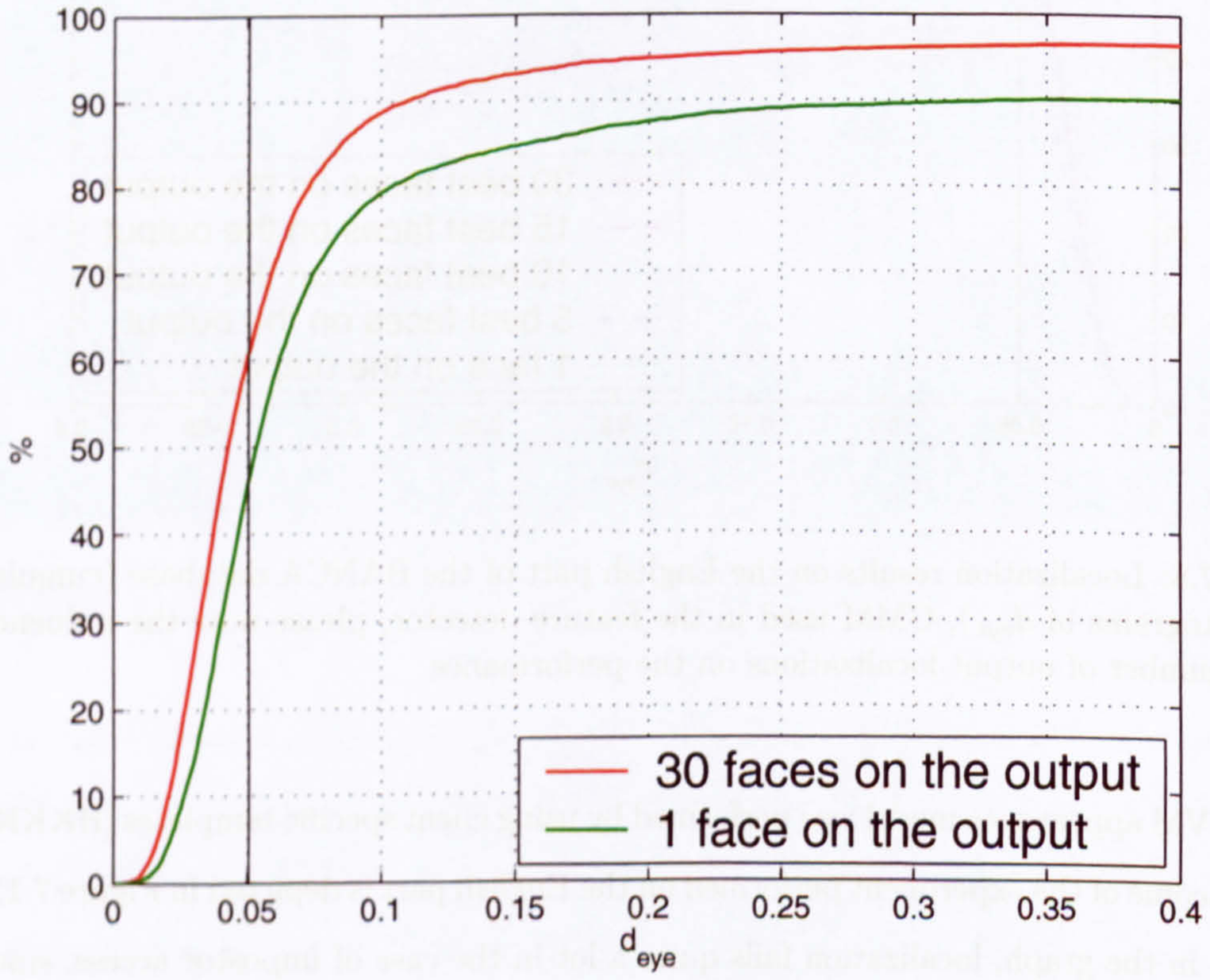

Figure 7.9: Localization results on the French part of the BANCA database (cumulative histograms of d_{eye}), GMM used in the feature detector

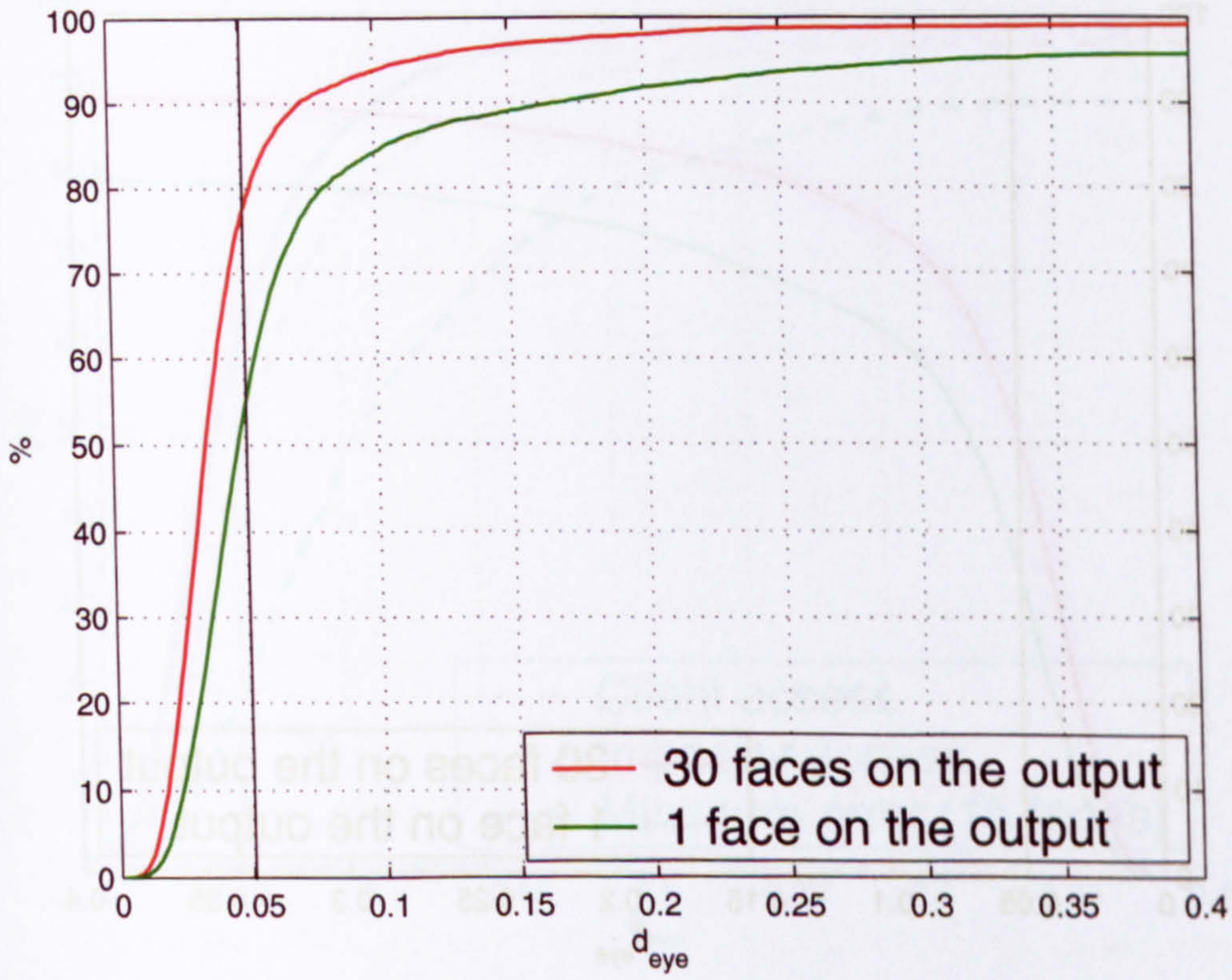

Figure 7.10: Localization results on the Spanish part of the BANCA database (cumulative histograms of d_{eye}), GMM used in the feature detector

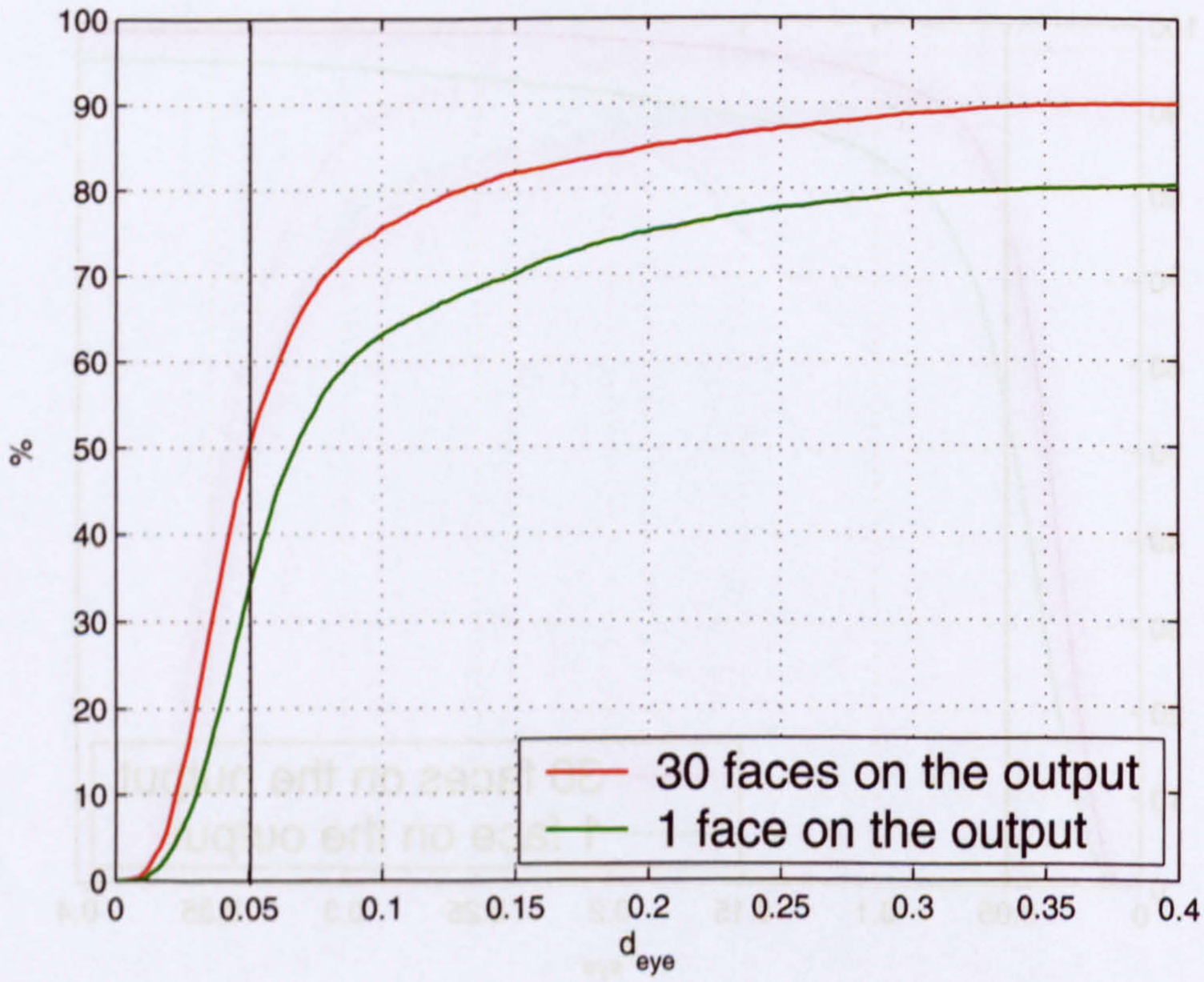

Figure 7.11: Localization results on the Italian part of the BANCA database (cumulative histograms of d_{eye}), GMM used in the feature detector

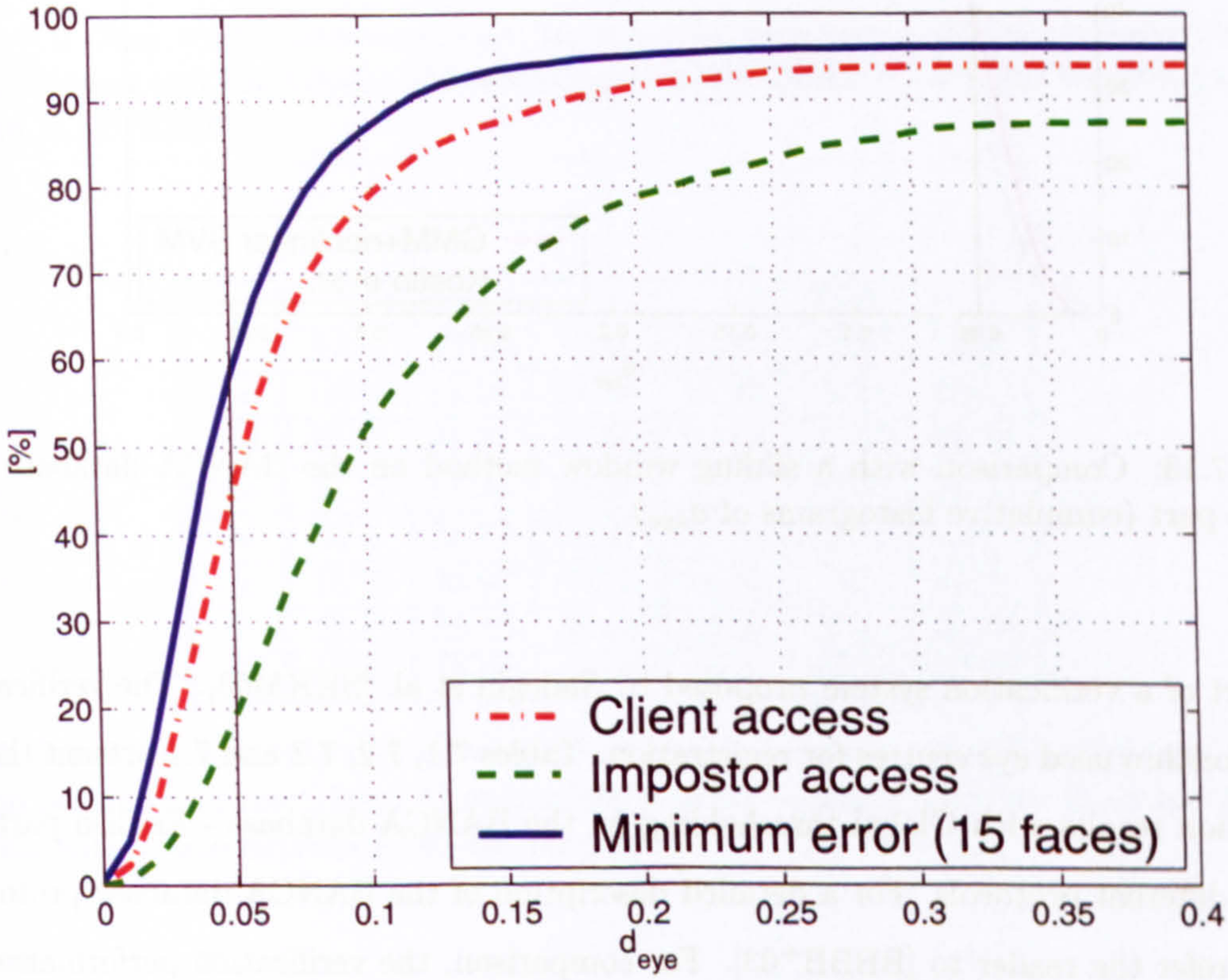

Figure 7.12: Results where best localization was chosen using client specific templates on the English part of the BANCA database [HKKK03], SCC used in the feature detector (cumulative histograms of d_{eye})

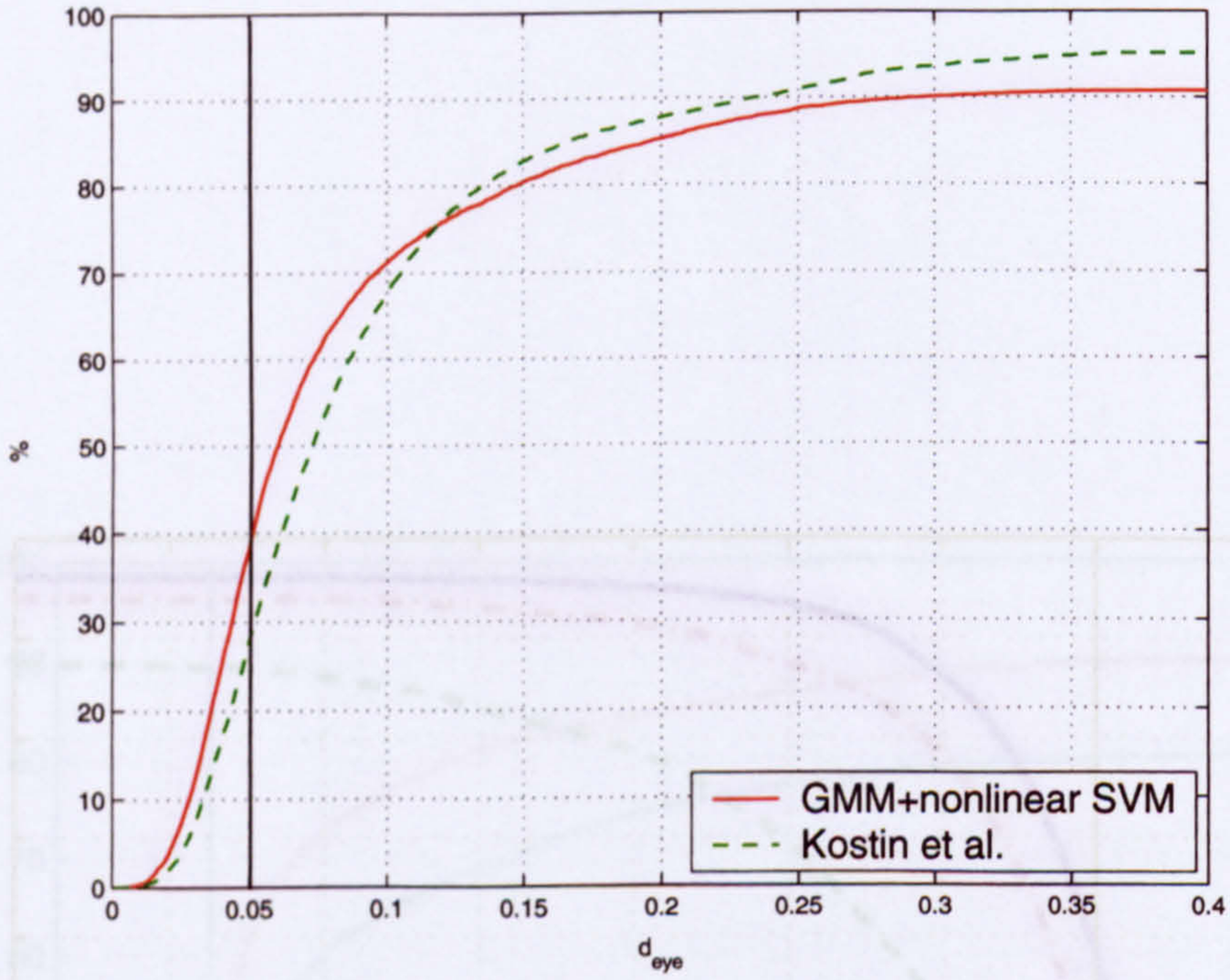

Figure 7.13: Comparison with a sliding window method on the BANCA database - English part (cumulative histograms of d_{eye})

first part of a verification system proposed by Sadeghi et al. [SKKM03]. The verification algorithm used eye centres for registration. Tables 7.1, 7.2, 7.3 and 7.4 present the verification results with Global thresholding on the BANCA database - English part, using 7 different protocols. For a detailed description of the BANCA database protocols we refer the reader to [BBBB+03]. For comparison, the verification performance using the localization method of Kostin et al. [KK02] as the baseline method is also presented. As seen from the tables by comparing Total Error (TER) values, the proposed method using 30 faces on the output significantly outperforms the system using the baseline method for localization (Table 7.1 against 7.3). In the case of one localization hypothesis on the output, our algorithm still outperforms the baseline method in six out of seven protocols with the exception being the protocol Ma (Table 7.1 against 7.2). It is also true, that apart from one protocol (Ua), there is a significant performance boost when using 30 hypotheses compared to the one hypothesis on the output (Table 7.3 against 7.2). Verification results using the groundtruth eye coordinates are shown in Table 7.4.

	Evaluation			Test		
	FAR	FRR	TER	FAR	FRR	TER
Mc	19.71	19.62	39.33	19.81	19.62	39.42
Md	28.17	27.44	55.61	28.08	26.79	54.87
Ma	15.38	15.13	30.51	15.67	15	30.67
Ud	30	29.87	59.87	29.23	29.74	58.97
Ua	24.23	25.64	49.87	23.85	27.05	50.9
P	24.68	25.43	50.11	25.19	26.45	51.65
G	18.53	19.49	38.01	18.88	19.53	38.41

Table 7.1: Face verification results on the BANCA database using Normalized Correlation Scoring and the Global Thresholding method together with the localization by Kostin et al. [KK02]

	Evaluation			Test		
	FAR	FRR	TER	FAR	FRR	TER
Mc	18.46	18.08	36.54	18.27	18.21	36.47
Md	20.67	21.67	42.34	21.54	21.03	42.56
Ma	18.17	17.44	35.61	17.6	16.92	34.52
Ud	27.5	25.77	53.27	27.5	26.41	53.91
Ua	23.75	23.97	47.72	24.33	25	49.33
P	23.81	23.16	46.98	23.24	23.12	46.36
G	16.67	17.95	34.62	16.7	18.12	34.82

Table 7.2: Face verification results on the BANCA database using Normalized Correlation Scoring and the Global Thresholding method together with the proposed localization with 1 face hypothesis on the output

	Evaluation			Test		
	FAR	FRR	TER	FAR	FRR	TER
Mc	10.29	10.51	20.8	10.77	9.359	20.13
Md	11.92	12.18	24.1	12.6	12.56	25.16
Ma	9.712	10.51	20.22	9.135	10.9	20.03
Ud	20.87	21.28	42.15	22.12	24.36	46.47
Ua	22.88	22.82	45.71	24.23	25.51	49.74
P	19.2	18.72	37.92	20.06	20.43	40.49
G	8.942	9.359	18.3	9.135	9.701	18.84

Table 7.3: Face verification results on the BANCA database using Normalized Correlation Scoring and the Global Thresholding method together with the proposed localization with 30 face hypotheses on the output

	Evaluation			Test		
	FAR	FRR	TER	FAR	FRR	TER
Mc	4.135	4.359	8.494	5.192	6.923	12.12
Md	5.962	6.667	12.63	6.346	6.154	12.5
Ma	6.635	6.667	13.3	6.635	6.667	13.3
Ud	13.27	15.38	28.65	13.27	15.38	28.65
Ua	18.27	18.72	36.99	18.94	20.9	39.84
P	12.69	15.77	28.46	12.24	15.94	28.18
G	5.128	4.231	9.359	5.096	3.974	9.071

Table 7.4: Face verification results on the BANCA database using Normalized Correlation Scoring and the Global Thresholding method together with the groundtruth eye coordinates

7.4 BioID database

This dataset consists of 1521 gray level images with a resolution of 384x286 pixel. Each one shows the frontal view of a face of one out of 23 different test persons. For evaluation purposes the set also contains manually located eye coordinates. We regard this set as very challenging, because not only the background is complex and changing, but scale variance is high and the illumination changes considerably. Sample images from this database can be seen in Figure 7.14.

7.4.1 Localization results on the database

On this database our method outperforms the baseline method in both cases (see Figure 7.15). In the case of the best localization on the output the improvement is 12% and in case of 30 hypotheses on the output, it is 20% for GMM. The refinement step in the appearance test improved the results by 30.3% when GMM was used and by 22.6% in the SCC case. The replacement of SCC by GMM increased the overall performance by 3.7% in the case of 30 output faces. Since the BioID database contains very difficult data which reflect realistic conditions, these results are more significant than with XM2VTS. It proves that our system copes well in adverse conditions.

Figure 7.16 presents a performance comparison with the sliding window method of Kostin et al.

Figure 7.14: Sample images taken from the BioID database

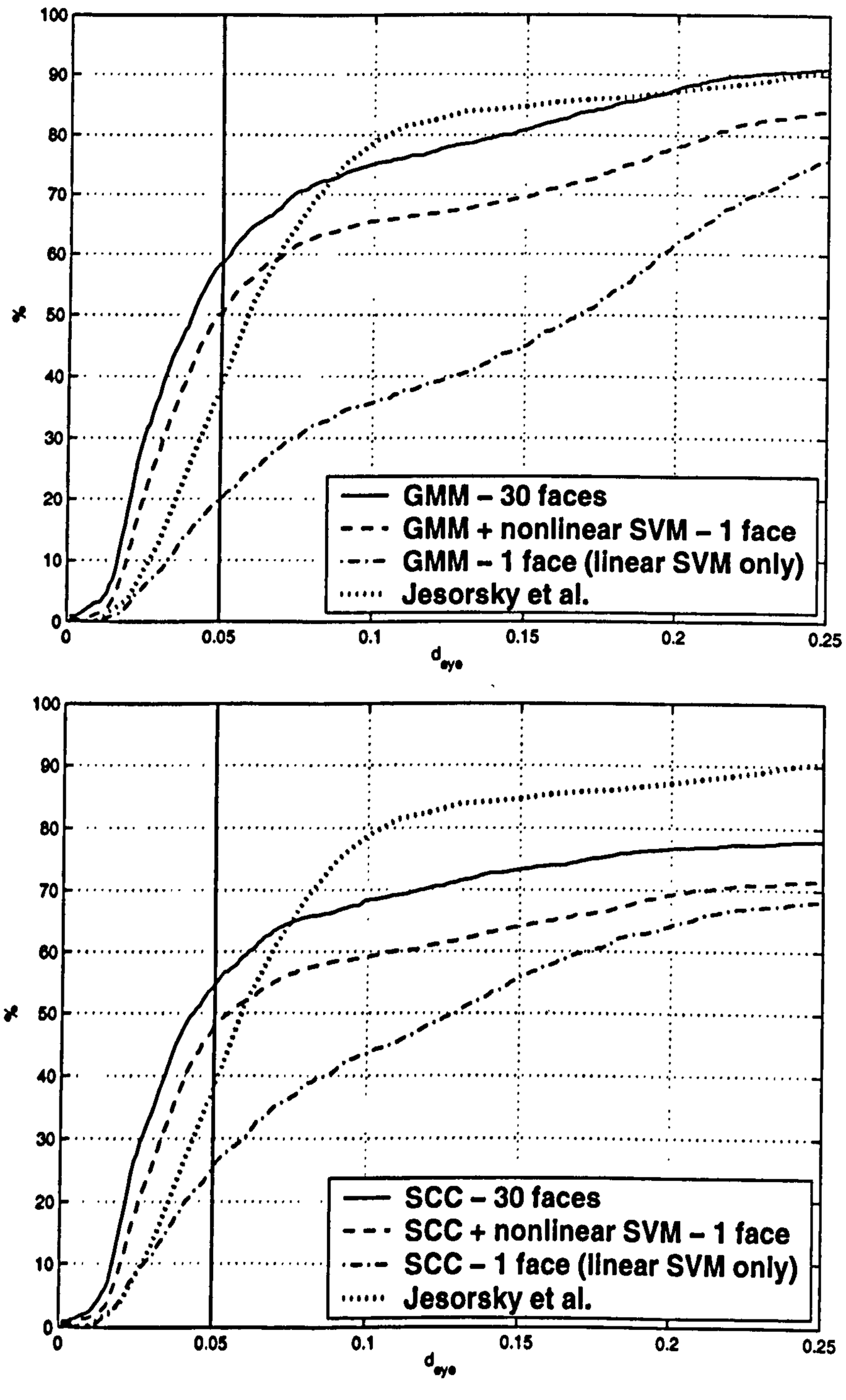

Figure 7.15: Results on the BioID database (cumulative histograms of d_{eye}): GMM top, SCC bottom, the graph of Jesorsky et al. taken from [JKF01]

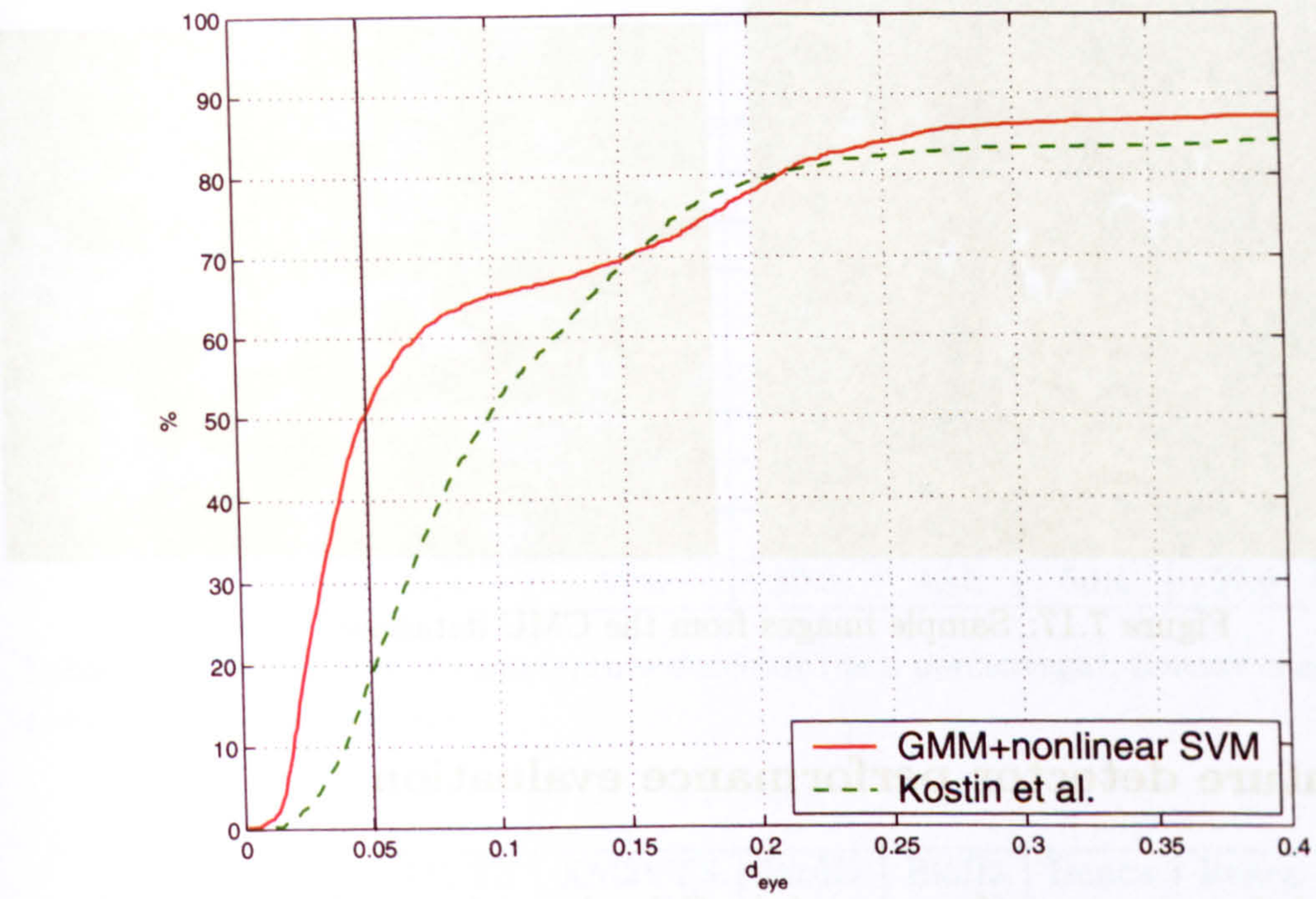

Figure 7.16: Comparison with a sliding window method on the BioID database (cumulative histograms of d_{eye})

7.5 Other face databases

As mentioned above there are many face databases, however only few are simulating real authentication scenarios. Also the groundtruth information often does not include facial features. The advocated algorithm is optimised for use in authentication scenarios, where a single frontal face of a sufficient size and resolution is present. Some databases, like e.g. CMU database obviously aim at different situations - see Figure 7.17 and are more suitable for the evaluation of face detection in surveillance or tracking. Due to the insufficient face resolution in these databases the advocated method is bound to fail on such images, since our method starts working from certain face size due to its dependence on feature detectors (approx. from 40-pixel inter-eye distance).

Figure 7.17: Sample images from the CMU database

7.6 Feature detector performance evaluation

We measured the accuracy performance of the Gabor-based feature detectors used in our experiments in order to asses their contribution to the overall performance. For training we used faces normalized to an inter-eye distance of 40 pixels (i.e. scale and orientation was removed). The choice of this value is a result of several experiments. The feature detector starts working approximately from this size of face. We define a successful feature localization as the situation when among all the detected features in the image exist at least one with $d_F \leq 0.05$, where d_F is defined is a similar manner as d_{eye}:

$$d_F = \frac{||(x_F, y_F) - (x_G, y_G)||}{||C_l - Cr||} \tag{7.2}$$

(x_F, y_F) are the coordinates of the detected feature F, (x_G, y_G) the groundtruth coordinates of the feature F and C_l, C_r the groundtruth coordinates of the left and right eye centres.

The following table depicts several performance measurements on all three databases introduced earlier.

Several interesting conclusions can be drawn from the reported tables. If we decided to detect only eye-centres (as many other methods do) we would always detect significantly fewer faces than with our method exploiting triplets of features - the difference is visible

Feature label	XM2VTS SCC	XM2VTS GMM	BioID SCC	BioID GMM	Banca SCC	Banca GMM
1	50.4	47.6	32.0	49.0	24.6	38.8
2	35.1	63.0	44.4	71.0	40.0	56.6
3	57.4	55.7	39.2	39.9	34.3	33.5
4	23.4	45.8	20.6	29.3	27.3	24.4
5	73.4	58.9	57.7	39.4	56.5	50.2
6	51.3	57.2	46.3	45.4	10.2	30.5
7	60.6	35.7	19.9	9.7	22.8	15.3
8	67.4	49.3	48.9	17.8	55.1	56.7
9	22.1	43.6	21.0	38.3	34.6	46.9
10	44.3	33.8	39.3	45.3	50.4	50.6

Table 7.5: Performance of each feature detector (as a percentage), feature matrix with 4 orientations and 3 scales

Feature label	XM2VTS SCC	XM2VTS GMM	BioID SCC	BioID GMM	Banca SCC	Banca GMM
1	51.4	56.1	50.2	55.6	36.9	41.4
2	76.3	84.2	66.1	67.6	46.9	60.3
3	71.3	70.9	43.7	51.2	38.1	44.5
4	49.8	50.9	37.1	39.1	36.6	44.0
5	76.1	84.9	68.9	61.1	63.7	67.4
6	60.0	64.2	50.5	54.8	28.2	34.3
7	72.2	70.4	48.4	29.5	41.3	54.8
8	76.8	75.5	49.4	34.5	61.2	63.6
9	34.3	54.2	40.7	40.0	45.9	49.6
10	47.2	45.8	46.4	48.7	58.5	61.8

Table 7.6: Performance of each feature detector, feature matrix with 5 orientations and 4 scales

	XM2VTS SCC	XM2VTS GMM	BioID SCC	BioID GMM	Banca SCC	Banca GMM
At least one well-posed triplet detected	75.4	72.2	56.5	56.7	59.8	65.7
Both eye centres detected	29.9	42.0	33.3	31.9	24.6	32.9

Table 7.7: Triplets and eye pair detection rates, feature matrix with 4 orientations and 3 scales

	XM2VTS SCC	XM2VTS GMM	BioID SCC	BioID GMM	Banca SCC	Banca GMM
At least one well-posed triplet detected	86.7	88.3	76.1	73.4	75.2	81.4
Both eye centres detected	62.3	74.5	51.5	48.6	32.1	44.0

Table 7.8: Triplets and eye pair detection rates, feature matrix with 5 orientations and 4 scales

by comparing values from Tables 7.7 and 7.8. For XM2VTS database we would get at best 74.5% faces detected if all eye centre pairs we checked (assuming that the appearance test would be errorless). If we assume that every localized triplet would lead to a successful localization, then by using our method, the total performance on XM2VTS would be 88.3%, i.e. the improvement achieved would be 13.8%. On realistic databases our method gains even bigger performance boost (the most on the BANCA database). Please note that we actually reached the top theoretical performance in the case of XM2VTS database with 30 faces on the output (see Figure 7.5). But it is also fair to say that possibly some of the not perfectly localized triplets could have still led to a well localized eye pair due to approximation errors caused by the assumption of the rigidness of the face and the affine transformation model. For example if someone smiles or opens their mouth too much, then, from the appearance model point of view, the mouth corner could possibly lead to a poor appearance score even if correctly detected. In this case, some mouth corner false alarm beneath the real mouth corner could perform better, since all features are currently modelled as rigid objects on the face. This is actually a desired behaviour, because the total performance is what is important. A few more tests using wrong features but producing correct localizations are definitely acceptable.

Regarding the comparison of SCC and GMM classifiers for the case of feature triplet detection, when using 12 complex features in matrix **G** (see Eq 4.15) SCC and GMM are quite comparable on XM2VTS and BioID, but on the BANCA database GMM outperforms SCC (see Table 7.7). When 20 complex features are used (Table 7.8), GMM outperforms SCC on the XM2VTS database and also on the BANCA database

and performs a bit worse on the BioID database (see Table 7.8). However the total face localization rate was always slightly better in the GMM case, and that is an important issue for us. These minor inconsistencies could possibly be attributed to a certain manual registration error (manually registered data always carry a certain error) and also the approximation error effects mentioned above could have contributed to the differences. Nevertheless the comparison figures show that SCC is a well designed and theoretically sound algorithm compared to the GMM-based model.

The reported results also show that if more features are used (represented by the feature matrix **G**) a higher feature detection rate is achieved (compare Tables 7.5 versus 7.6 and 7.7 versus 7.8). This result is expected. However one cannot forget that more features means more computational requirements, so this has to considered in the design. We have tested several different feature matrix configurations however only two of the aforementioned configurations were shown (12 complex features ($12 = 3scales \times 4orientations$) versus 20 complex features ($20 = 5orientations \times 4scales$)). With regard to scale invariance, 3 scale-invariance shifts of the feature matrix (see section 4.3.2) proved to be sufficient to cover the scale variations in the data. We have also observed that without performing rotation-invariance shifts of the feature matrix, the output of the feature detector was stable for changes of head orientation up to approximately 10 degrees. Since the test data did not contain heads with a large amount of tilt, the rotation invariance steps did not have to be used.

If we judge the performance of the feature detectors alone (Tables 7.5, 7.6), it can be seen, that one cannot rely completely on the success of one particular feature detector. Features themselves can be often occluded or shadowed in such a way that it is hard for even a person to see them. However even in such situations humans are clearly able to establish where the eye or nose is by using the information surrounding these features. It can be argued, that the advocated approach behaves in a similar manner. It does not always localize eye centres directly by a successful feature detection, very often eye centres are missed (see the results reported above), however even in this situation by using the surrounding photometric information on the face the algorithm can still successfully estimate the most likely positions of eyes.

7.7 Summary

We have assessed the localization performance of the proposed method on several benchmarking datasets. Our results show, that regarding accuracy, the advocated approach outperforms the baseline method by Jesorsky et al. [JKF01] and also it is superior to a typical representative of a sliding window method. In the case of the appearance modelling, we concluded that the PCA probabilistic model was not able to perform well in the presence of a cluttered background. The SVM-based model showed to be superior to the PCA in such a case. However a coarse resolution, linear SVM-model cannot be relied upon if only one localization hypothesis on the output is needed. For such a purpose, a 3rd degree polynomial SVM trained in finer resolution dramatically improved the results. Possibly other appearance models and classifiers can be exploited in the appearance test, leaving room for improvement.

We also showed that feature detectors alone could not lead to a satisfactory performance. Nevertheless when exploiting the proposed constellation and appearance models the performance boost achieved is dramatic.

Chapter 8

Conclusions

8.1 Summary

In this thesis we have progressed the state of the art in face localization. We have presented a bottom-up algorithm using a single grey-scale image that can successfully localize a face. Although it is designed to be used in authentication scenarios, an extension to more general situations would in principle be possible.

The chosen face representation is feature-based and exploits a simpler shape model than that used in the case of Active Appearance or Shape approaches (affine model). The proposed hypotheses generation and verification method avoids the iterative search as in the case of AAMs and ASMs, since the presence of a face can be decided in two simple steps. The choice of a simple three-point alignment model (affine) introduces a certain registration error, however it is addressed in the later stage by the Support-Vector-Machine-based appearance model. We believe that for the purpose of the face detection/localization in authentication scenarios, this representation is sufficient. Although our method produces accurate locations of eye centres, AAMs or ASMs could still be exploited as a final-processing step if higher accuracy registration was needed.

In chapter 7 we demonstrated on several realistic face databases that the advocated method performs well in the cluttered background and outperforms baseline methods regarding localization accuracy. We showed in section 7.6 that by combing detected fea-

tures in the advocated way, a significant performance improvement could be achieved, as compared to the performance of feature detectors alone. It also became apparent that simple models like probabilistic PCA are unable to capture the huge appearance variability of faces over population and capture conditions, whereas SVMs kept their reputation as a powerful pattern recognition tool.

To summarize, the following concepts contributed to the performance of our method:

- Properly designed scale and rotation invariant feature detector, modelling the appearance variability by the statistical means.
- Fast hypotheses-generation algorithm that treats feature false alarms as a naturally occurring phenomenon and facilitates the removal of scene capture effects (mainly scale and orientation) by exploiting the transformation from the predefined model face space to the image space.
- Powerful appearance model in geometrically normalized space exploiting cascaded Support-Vector-Machine classification. The use of the shape-free representation helped to increase the localization accuracy, since without scale and rotational discretization the model could be made more discriminative and sensitive towards pixel misalignments.

8.2 Future work

Although our algorithm succeeded in meeting the pre-defined requirements, some modifications could possibly still be fruitful. It was shown earlier that the algorithm was versatile regarding its structure. Although the currently used feature detectors showed a very good performance for the required purpose, other detectors could be employed and this would not mean any modification of the existing method. The choice of features to detect could also be assisted by some form of unsupervised selection, based on optimization, and an optimal set based on several criteria could be derived. Another important observation is that any of the existing "scanning window" pattern recognition methods could be used as an appearance model after the necessary modifications. Also some recent illumination correction methods tailored for faces could

bring a performance boost, since in our study we used only the classical zero mean, unit variance approach. Our current research implementation does not primarily focus on speed (approx. 13 secs/image on Pentium 4, 2.8GHz), however we believe that real-time performance is possible.

Bibliography

[ABR64] M. Aizerman, E. Braverman, and L. Rozonoer. Theoretical Foundations of the Potential Function Method in Pattern Recognition Learning. *Automation and Remote Control, 25:821–837*, 1964.

[ban] http://falbala.ibermatica.com/banca/index.html.

[BBBB+03] E. Bailly-Bailliére, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariéthoz, J. Matas, K. Messer, V. Popovici, F. Porée, B. Ruiz, and J.-P. Thiran. The BANCA Database and Evaluation Protocol. In *Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, 2003.

[Bis97] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1997.

[BLP95] M. C. Burl, T. K. Leung, and P. Perona. Face localization via shape statistics. In *Proc. of International Workshop on Automatic Face and Gesture Recognition*, pages 154–159, 1995.

[BMHK] P. Bílek, J. Matas, M. Hamouz, and J. Kittler. Detection of Human Faces from Discriminative Regions. Technical Report VSSP–TR–2/2001, University of Surrey (http://citeseer.nj.nec.com/595083.html).

[CC03] D. Cristinacce and T. Cootes. Facial feature detection using AdaBoost with shape constraints. In *Proc. of British Machine Vision Conference, 1:213–240*, 2003.

[CCTG95] T.F. Cootes, D. Cooper, C.J. Taylor, and J. Graham. Active Shape Models – Their Training and Application. *Computer Vision and Image Understanding, Vol. 61, No.1:38–59*, 1995.

[CET98] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. In *Proc. of European Conference of Computer Vision, 2:484–498*, 1998.

[CT01] T.F. Cootes and C.J. Taylor. Constrained Active Appearance Models. In *Proc. of ICCV, 1:748–754*, 2001.

[CV95] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[CWT00] T.F. Cootes, K.N. Walker, and C.J. Taylor. View-based Active Appearance Models. In *Proc. of Int. Conf. on Face and Gesture Recognition, 227–232*, 2000.

[DH73] R. E. Duda and P. E. Hart. *Pattern Classification and Scene Analysis.* John Wiley, 1973.

[DLR] A. Demster, N. Laird, and D. Rubin. Maximum Likelihood Estimation from Incomplete Data Via the EM Algorithm. *Journal of Royal Statistical Society B, 39:1–38.*

[dlTGM98] F. de la Torre, S. Gong, and S. McKenna. View-Based Adaptive Affine Tracking. In *Proc. of ECCV'98, Freiburg, Germany,1:828–842*, 1998.

[Erä01] P. Erästö. *Support Vector Machines - Backgrounds and Practice.* PhD thesis, Rolf Nevanlinna Institute, 2001.

[ETC98] G. J. Edwards, C. J. Taylor, and T. F. Cootes. Learning to Identify and Track Faces in Image Sequences. In *ICCV, 317–322*, 1998.

[FJ02] M.A.T. Figueiredo and A.K. Jain. Unsupervised learning of finite mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):381–396, 2002.

[FK00] B. Fröba and Ch. Küblbeck. Orientation Template Matching for Face Localization in Complex Visual Scenes. In *Proc. of Int. Conf. on Image Processing, 251-254*, 2000.

[GMP00] S. Gong, S. J. McKenna, and A. Psarrou. *Dynamic Vision, From Images to Face Recognition.* Imperial College Press, 2000.

[GOM98] S. Gong, E.-J. Ong, and S. McKenna. Learning to associate faces across views in vector space of similarities to prototypes. In *Proc. of BMVC, Nottingham, England*, 1998.

[Goo63] N. R. Goodman. Statistical Analysis Based on a Certain Multivariate Complex Gaussian Distribution (An Introduction). *The Annals of Mathematical Statistics, 34:152-177*, 1963.

[GPR97] S. Gong, A. Psarrou, and S. Romdhani. Corresponding dynamic appearences. *Image and Vision Computing, 20:307-318*, 1997.

[HKKK03] M. Hamouz, J. Kittler, J.K. Kämäräinen, and H. Kälviäinen. Hypotheses-driven Affine Invariant Localization of Faces in Verification Systems. In *Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, Guildford, UK, 2003.

[HKMB02] M. Hamouz, J. Kittler, J. Matas, and P. Bílek. Face Detection by Learned Affine Correspondences. In *Proceedings of Joint IAPR International Workshops SSPR02 and SPR02, 566-575*, August 2002.

[HL01] Erik Hjelmås and Boon Kee Low. Face detection: A survey. *Computer Vision and Image Understanding, 83:236-274*, 2001.

[HPP] B. Heisele, T. Poggio, and M. Pontil. Face detection in still gray images, *AI Memo 1687, Massachusetts Institute of Technology*, 2000 (http://citeseer.nj.nec.com/heisele00face.html).

[htta] http://www.ee.surrey.ac.uk/banca/.

[httb] http://www.kernel-machines.org.

[JKF01] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust Face Detection Using the Hausdorff Distance. In *AVBPA 2001, 90–95*, volume 2091 of *J. Bigun and F. Smeraldi, Lecture Notes in Computer Science*, Halmstad, Sweden, 2001. Springer.

[Käm03] J.-K. Kämäräinen. *Feature Extraction Using Gabor Filters.* PhD thesis, Lappeenranta University of Technology, 2003.

[KK02] A Kostin and J. Kittler. Fast Face Detection and Eye Localization Using Support Vector Machines. In *Proceedings of the 6th International Conference "Pattern Recognition and Image Analysis: New Information Technologies", 371–375*, 2002.

[KKK+02] J.-K. Kämäräinen, V. Kyrki, H. Kälviäinen, M. Hamouz, and J. Kittler. Invariant Gabor features for face evidence extraction. In *Proceedings of MVA2002 IAPR Workshop on Machine Vision Applications, 228–231*, 2002.

[KKK04] V. Kyrki, J.-K. Kamarainen, and H. Kälviäinen. Simple Gabor Feature Space for Invariant Object Recognition. *Pattern Recognition Letters, 25(3):311–318*, 2004.

[KP97] C. Kotropoulos and I. Pitas. Face authentication based on morphological grid matching. In *Proc. of IEEE International Conference on Image Processing, Vol. 1, 105–108*, 1997.

[LGL00] Y. Li, S. Gong, and H. Lidell. Support Vector Regression and Classification Based Multi-view Face Detection and Recognition. In *Proc. of IEEE Int. Conference on Face and Gesture Recognition, Grenoble, France*, 2000.

[LVB+93] M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. v.d. Malsburg, R. P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42(3):300–311, Mar 1993.

[MBHK02] J. Matas, P. Bílek, M. Hamouz, and J. Kittler. Discriminative Regions for Human Face Detection. In *Proceedings of Asian Conference on Computer Vision*, January 2002.

[MG98] S.J. McKenna and S. Gong. Real-Time Pose Estimation. *Journal of Real-Time Imaging, 4:333–347*, 1998.

[MMK+99] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In R. Chellapa, editor, *Second International Conference on Audio and Video-based Biometric Person Authentication*, pages 72–77, Washington, USA, March 1999. University of Maryland.

[MP95] B. Moghaddam and A. Pentland. Probabilistic visual learning for object detection. In *IEEE International Conference on Computer Vision*, pages 786–793. IEEE, Piscataway, NJ, USA, 1995.

[MP96] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. In *Early Visual Learning*, pages 99–130. Oxford University Press, 1996.

[NG02] J. Ng and S. Gong. Composite support vector machines for the detection of faces across views and pose estimation. *Image and Vision Computing, 20:359–368*, 2002.

[OFG97] E. Osuna, R. Freund, and F. Girosi. Training Support Vector Machines: an Application to Face Detection. In *Proc. of CVPR '97:130–136, 1997*, 1997.

[Pla98] J. Platt. Sequential minimal optimization: A fast algorithm for training support vector machines, *Technical Report 98-14, Microsoft Research, Redmond, Washington*, 1998.

[RBK98] Henry Rowley, Shumeet Baluja, and Takeo Kanade. Rotation invariant neural network-based face detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, June 1998.

[RGP99] S. Romdhani, S. Gong, and A. Psarruo. A Multi-View Nonlinear Active Shape Model Using Kernel PCA. In *Proc. of BMVC*, 1999.

[RKG96] M.J.T. Reinders, R.W.C. Koch, and J.J. Gerbrands. Locating Facial Features in Image Sequences using Neural Networks . In *Proc. of 2nd Int. Conf. on Automatic Face and Gesture Recognition*, 1996.

[RPG00] S. Romdhani, A. Psarrou, and S. Gong. On Utilising Template and Feature-Based Correspondence in Multi-view Appearance Models. In *Proc. of 6th European Conference on Computer Vision, 1:799-813*, 2000.

[SGO01] J. Sherrah, S. Gong, and E.-J. Ong. Face distribution in similarity space under varying head pose. *Image and Vision Computing, Vol. 19, No. 11*, 2001.

[SK00] Henry Schneiderman and Takeo Kanade. A Statistical Model for 3D Object Detection Applied to Faces and Cars. In *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, June 2000.

[SKKM03] M. Sadeghi, J. Kittler, A. Kostin, and K. Messer. A Comparative Study of Automatic Face Verification Algorithms on the BANCA Database . In *Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, Guildford, UK, 2003.

[SP98] K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–50, January 1998.

[TK99] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, 24-28 Oval Road, London NW1 7DX, UK, 1999.

[TP91] M. A. Turk and A. P. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[Vap95] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, Bell Laboratories, 1995.

[VJ01] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition, 1:511–518*, 2001.

[VL02] N. Vlassis and A. Likas. A Greedy EM Algorithm for Gaussian Mixture Learning. *Neural Processing Letters*, 15:77–87, 2002.

[VS00] V. Vogelhuber and C. Schmid. Face detection based on generic local descriptors and spatial constraints. In *Proc. of International Conference on Computer Vision, 1084–1087*, 2000.

[WEWP00] M. Weber, W. Einhauser, M. Welling, and P. Perona. Viewpoint-invariant learning and detection of human heads. In *Proc. of IEEE International Conference on Automatic Face and Gesture Recognition*, 2000.

[WFKvdM97] L. Wiskott, J.-M. Fellous, N. Krüger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. In Gerald Sommer, Kostas Daniilidis, and Josef Pauli, editors, *Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP'97, Kiel*, number 1296, pages 456–463, Heidelberg, 1997. Springer-Verlag.

[WWP00] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *Proc. 6th Europ. Conf. Comput. Vision, Dublin, Ireland, 1:18–32*, 2000.

[YC96] K. Yow and R. Cipolla. Feature-based human face detection, Technical Report, Department of Engineering, University of Cambridge, England, http:// citeseer.nj.nec.com/yow96featurebased.html, 1996.

[YKA02] M. Yang, D. J. Kriegman, and N. Ahuja. Detecting Faces in Images: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

[YLGL01] S. Y. Li, S. Gong, and H. Liddell. Modelling faces dynamically across views and over time. In *Proc. of ICCV, 554–559*, 2001.

[YRA00] M.-H. Yang, D. Roth, and N. Ahuja. A SNoW-based face detector. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 855–861. MIT Press, 2000.

List of Publications

E. Bailly-Bailliére, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariéthoz, J. Matas, K. Messer, V.Popovici, F. Porée, B. Ruiz, and J.-P. Thiran. The BANCA Database and Evaluation Protocol. *In Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, 2003.

M. Hamouz, J. Kittler, J.K. Kämäräinen, and H. Kälviäinen. Hypotheses-driven Affine Invariant Localization of Faces in Verification Systems. *In Proc. 4th Int. Conf. on Audio- and Video-based Biometric Person Authentication*, 2003.

J.-K. Kämäräinen, V. Kyrki, H. Kälviäinen, M. Hamouz, and J. Kittler. Invariant Gabor Features for Face Evidence Extraction. *In Proceedings of MVA2002 IAPR Workshop on Machine Vision Applications, 228–231*, 2002.

M. Hamouz, J. Kittler, J. Matas, and P. Bílek. Face Detection by Learned Affine Correspondences. *In Proceedings of Joint IAPR International Workshops SSPR02 and SPR02, 566–575*, 2002.

J. Matas, P. Bílek, M. Hamouz, and J. Kittler. Discriminative Regions for Human Face Detection. *In Proceedings of Asian Conference on Computer Vision*, 2002.

J. Matas, M. Hamouz, K. Jonsson, J. Kittler, Y. Li, C. Kotropoulos, A. Tefas, I. Pitas, T. Tan, H Yan, F. Smeraldi, J. Bigun, N. Capdevielle, W. Gerstner, S. Ben-Yacoub, Y. Abdeljaoued, and E. Mayoraz. Comparison of Face Verification Results on the XM2VTS Database. *In Proc. Int. Conf. on Patt. Rec. ICPR*, 2000.